/mlst

One Step Closer to the Star Trek Voice AI Assistant!

March 24, 2026



Machine Learning Street Talk

Podcast Transcript

Revision: 197f779b

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

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

Contents

Overview 2

ASR and diarization fundamentals 3

Real-time conversational AI architecture 8

Neural network streaming and multi-modal integration 10

Enterprise voice AI and real-time translation 12

Production deployment and testing infrastructure 16

Model architecture and latency-accuracy tradeoffs 20

Lattice-based decoding and language model integration 24

ASR performance metrics and real-world evaluation 28

Ethics and privacy in speech technology 33

Self-supervised learning and low-resource languages 39

Feature engineering to automated ML 45

Infrastructure scaling and PyTorch critique 51

Future of conversational AI and Ursa 2 59

References 66

Overview

Will Williams is CTO of Speechmatics, the Cambridge-based speech recognition company. Tim has used their API for years to caption MLST episodes, and this conversation happened in their offices — which explains the live demo at the start where an AI moderates a political debate clip in real time.

The technical meat covers how Speechmatics builds production ASR systems. Their approach is hybrid: self-supervised pre-training on unlabeled audio gets them comparable accuracy to fully supervised systems like Whisper, but with roughly 100x less labeled data. Williams explains why this matters for scaling to low-resource languages where you simply don't have thousands of hours of human-transcribed speech.

The architecture discussion is detailed. Their system runs multiple operating points with different latency-accuracy tradeoffs. They pad latency up to 1.8 seconds to keep the user experience consistent rather than optimizing for raw speed. Decoding uses lattices with language model integration, which lets them rescore hypotheses and handle things like proper nouns and domain-specific vocabulary without retraining the acoustic model.

Diarization — figuring out who said what — comes up repeatedly. Williams calls it harder than ASR itself, partly because speaker embeddings get corrupted by acoustic environments and partly because cross-talk creates genuinely ambiguous boundaries. They're pushing hard on implicit source separation but the problem remains open.

The conversation also covers their testing infrastructure (mirrored production traffic catches edge cases that unit tests miss), why they resist customer-specific fine-tuning (it fragments the model and makes global improvements harder), and Williams' critique of PyTorch memory management in production settings. He argues for more direct memory allocation rather than letting the framework handle it, which is a practical concern when you're serving models at scale.

This is a sponsored episode. Speechmatics' API is at speechmatics.com, and their Ursa 2 model covers 52 languages.

Featuring: Will Williams (CTO, Speechmatics) and Tim Scarfe.

ASR and diarization fundamentals •

Tim Scarfe 00:00:00

I have to say, I am bloody impressed with that. That is remarkable. Is there a way that we could give it a deliberately difficult problem?[1]

Will Williams 00:00:07

If you wanna to renegotiate more

Demo Clip 00:00:09

more American troops Hillary Clinton Hillary Clinton Hillary Clinton failed to renegotiate status of forces agreed. No. That is incorrect. And so we removed we removed all of our troops from Iraq, and ISIS was able to be conjured up in that vacuum But but I'd like to and overrun vast areas of Iraq.

Will Williams 00:00:25

It's done okay

Tim Scarfe 00:00:26

there. I'm very impressed with that.

Speaker 3 00:00:28

Oh, yeah. It's quite impressive. Isn't it, Tim?

Will Williams 00:00:32

Alright. Thanks, Hans. That'll do. That'll do. It's really cool. Wow. Demos never work first time on It's it's how the conversation flows. It's what it's able to do for you. That's the bottleneck. Latency is is I mean, yeah, it has to be low enough, but I think it's better to have a much more consistent experience where you know there's a bit of a pause. Otherwise, it can become quite chaotic. You know, 1 of things that we want to do is understand every voice. You know, this is our vision, this is our mission, We wanna execute against that. We take it seriously. And part of that means that we want to recognize underrepresented people, be as inclusive as we can, whilst recognizing that the training distribution is not always gonna cover everyone with love it too.

[1]Speechmatics PDF shownotes — paper

Tim Scarfe 00:01:12

So what which LLM are you using Flow? Will is from Speechmatics, and I'm very excited about this because I've been a customer myself of Speechmatics for many years. All of the captions that you read on MLST have been transcribed by Speechmatics for the last 2 years. I built a startup called X-ray, which is a speech transcription startup, which is using Speechmatics behind the scenes. And here I am in your offices speaking with you in person. It's an honor, Will. So thank you so much for coming on.[2][3]

Will Williams 00:01:43

Thanks for having me. I'm looking forward

Tim Scarfe 00:01:44

to our discussion. So what do you

Will Williams 00:01:46

do at Speechmatics? I'm CTO. I've had various different well, almost all different roles in the engineering team since I joined as an intern. But today I'm heading up our engineering team. Speechmatics is a speech to text API. Now no longer just that, we do real time translation and offer some kind of LLM powered APIs on top of our script speech from transcription output, But that's our bread and butter is to do transcription both batch and real time.

Tim Scarfe 00:02:14

So diarization is when algorithm can kind of figure out the boundaries between different speakers and that that's something that that your that your service supports. What's your current, you know, thinking around diarization? Diarization is

[2]Speechmatics — company
[3]Will Williams — person

Will Williams 00:02:30

a tough problem. It's probably tougher than ASR and I think it's slightly maybe underrepresented as a problem in general. I think it's something that we should be able to solve with current tech fairly well. We can embed speakers, we should be able to differentiate them, but there's there's lots of issues with acoustic environments sort of affecting those embeddings. But yeah, we're really pushing pushing hard for big advances in in diarization internally. Ultimately, there's extra information in the ASR that you want to exploit. You've got the words And and so that that there's a lot of signal there and so whatever system you have ultimately should be exploiting that information.

Tim Scarfe 00:03:12

It is a cool idea because why shouldn't it be able to do it? It should be able to automatically ascertain that that's David Chalmers.

Will Williams 00:03:18

Yep.

Tim Scarfe 00:03:19

But the reality is it's brittle.

Will Williams 00:03:21

Yeah. Well, I've got a demo actually which shows that working. Oh, have you? Yeah. Do you want to show Yeah.

Tim Scarfe 00:03:26

So so so you do automated diarization This and

Will Williams 00:03:30

is our new conversational assistant. Oh, okay. It should be held to infer. Who's who? Oh, that's amazing. And automatically kind of pick it up and start using your name and so on. Oh, wow. This is our new conversational assistant called Flow. Got a chap called Humphrey.

Speaker 3 00:03:46

Hello. I'm Humphrey. Nice to meet you.

Will Williams 00:03:49

Hi. My name's Will. How's it going?

Speaker 3 00:03:53

Will. Nice 1. I'm doing alright. Thanks. How about you then?

Will Williams 00:03:57

Yeah. Alright. Thanks. I'm I'm here with Tim from Machine Learning Street Talk. Perhaps he can introduce himself.

Tim Scarfe 00:04:03

Oh, hey. How how's it going, Humphrey? It's lovely to meet you.

Speaker 3 00:04:07

Nice to meet you too. Tim, thanks for introducing yourself. So what brings you 2 here today then?

Will Williams 00:04:13

Well, we wanted to talk about some of the the finer points of machine learning.

Speaker 3 00:04:18

Oh, fascinating. I'm not exactly an expert, but I'm happy to learn. What specific aspects of machine learning were you looking to discuss?

Will Williams 00:04:25

Dive into that, could you just tell me who's with me in the room?

Speaker 3 00:04:30

There's you, Will and Tim from Machine Learning Street Talk, and then there's me, Humphrey. That's the 3 of us, I think.

Tim Scarfe 00:04:37

Oh, that that's brilliant. And who who am I?

Speaker 3 00:04:40

You're Tim from Machine Learning Street Talk, if I'm not mistaken.

Tim Scarfe 00:04:45

No. No. You're you're not mistaken that this is Tim.

Speaker 3 00:04:49

Oh, good. Good. Just making sure, Tim.

Will Williams 00:04:54

Brilliant. This might not this might not go down as the world's best demo, but it does work.

Tim Scarfe 00:04:59

Yeah. That's amazing. So so it actually knows on the, you know, the embedding that it's me. Yeah. Wow.

Speaker 3 00:05:07

Oh, yeah. It's quite impressive. Isn't it, Tim?

Will Williams 00:05:11

Alright. That's it. Is Humphrey. That'll do. That'll do. It's really cool. Wow. Demos never work first time on

Demo Clip 00:05:17

the game.

Will Williams 00:05:17

That's really good. That's good.

Tim Scarfe 00:05:20

So what which LLM are you using today?

Will Williams 00:05:22

You know, I don't know. It's either Llama or

Real-time conversational AI architecture

Will Williams 00:05:25

no, it's Lama or GPT-four.

Tim Scarfe 00:05:27

Lama 3 or GPT-four. Yeah. Yeah. Yeah. Are you doing like an intermediate transcription and then sticking it to the LLM? Yeah. Latency wasn't too bad.

Will Williams 00:05:36

Yeah, latency is fine. So this is the thing about Moshi, as they're saying, latency is a bottleneck. We needed a 300 mil things or whatever. Yeah. It's not the bottleneck at all. It's it's the actual intelligence of the conversation. It's it's how the conversation flows, it's what it's able to do for you. That's bottleneck. Latency is I mean, yeah, it has to be low enough but I think it's better to have a much more consistent experience where you know there's a bit of a pause. Otherwise, it can become quite chaotic.

Tim Scarfe 00:06:03

The thing is the problem you've solved and I know this problem because I've implemented the same system myself is during TTS you have to turn the mic off because otherwise the TTS would feed back into the mic. We have a few heuristics in there as well. And and that's why I think the only way to really solve that problem and maybe you're proving me wrong I assumed us to have a bidirectional model.

Will Williams 00:06:20

Yeah. I think we need bidirectional models ultimately, but like this is good enough. And this gets you the performance of the LLM. Yeah. Because you're not affecting the weights of the LLM fine tuning them away. So, you know, I'm quite bullish on this approach going a long way.

Tim Scarfe 00:06:34

Yeah. That's amazing. Well, it was it was flawless. So my congratulations. Yeah. Amazing.

Will Williams 00:06:41

So like diarisation is kind of something, it's not well, yeah, it's not solved, but it does work.

Tim Scarfe 00:06:45

Yeah. What what do you think about neuro symbolic?

Will Williams 00:06:48

It's a term that I think used to make me a little uncomfortable. Very much just pure connectionist by heart, but I think I've come to appreciate that there's some applications where this is the correct way to do things. So if you have, I don't know, a particular application you wanna solve the game of Go, you want to search through a particular search base, English sentences, actually combining a neural approach with a more symbolic approach is very, very helpful. And, you know, the way that I think about, say the decode that happens in ASR, WFST decode. What what does that mean? It's a weighted finite state transducer. So these are these are these kind of data structures over which we can you can operate and you can do a search through and they define the costs of certain certain paths. Yeah. We can do a search over these things, and they can constrain what is possible in English for you. This is great. Like, this helps us to find an API as a company. So, you know, from, like, a product point of view, this is, you know, this is brilliant. If you're if you're doing multi colored research, this is also a very useful operation if if you have a particular setup. And so, you know, I'm not gonna I'm not gonna denigrate these approaches because because they're obviously useful when you're very, very applied. Then I think you get to the question of, like, reasoning and, you know, the AGI question and whether they're gonna be required there. And I think this may be where we slightly diverge on how how we think about things because I think I'd continue to be quite connectionist there. I think we want, you know, open ended reasoning. We want flexible kind of building of these programs and and and and planning abilities that are very situational, but I believe that's going to emerge from certain prediction tasks. I believe that if we're modeling at the right levels of representation that we can we can kind of we can do planning implicitly. And so the more primitive operations can approximate these discrete programs that that we write more explicitly for for things like cutting Monte Carlo research, for example.

Tim Scarfe 00:08:55

I I kind of agree. I I think that the statistical approach will get you to a place where the correct program is is in the space. But don't you still need to have because the beauty of having like an actual evaluation function is you can actually run the program and you can verify it and you can select the provably correct path. So we could get to a point where we like probabilistically select the program that we think works, but

Neural network streaming and multi-modal integration

Tim Scarfe 00:09:21

we don't know

Will Williams 00:09:21

that it works. Well, I'd argue humans are in a position where we have a state in our brains. And we don't know that certain actions are gonna provably get us the results that we want, but we have a certain level of experience in our environments that refine those representations, those predictions, those actions, such that we achieve the the goals that we want. It it might be a quicker way there. There might be you might be able to put more guarantees around things, but I'm not sure it's it's completely necessary. And I think, like, going back to what I'm saying before around representations being, I think, 1 of the key lenses to it through us to see things. I think we have the right rep representations clocked at the right rate. They can almost have your plan encoded, you know, in a very, you know, very dense kind of vector almost that can then

Tim Scarfe 00:10:08

just be unrolled. Yeah. I agree. I mean, there was that great work by Bengio, GFlowNets, that that did something very, very similar. And also I should say that even when we do system 2 reasoning we're not actually verifying completely. So on an IQ test or on an ARC challenge test we can verify the program against the test instances, and we only have 3 or 4. And actually there are an infinite number of possible programs that could generalize, and we anthropomorphically select the 1 that is what, you know, the the least complexity, the most anthropomorphically aligned or whatever. So it's not a perfect process, but it feels different, feels like there's some kind of meta process of evaluation, you know, and and and that's our system too.[4]

[4] GFlowNets — paper

Will Williams 00:10:51

Yeah. But there's lots there that we can't get our hands on like aesthetics. Yeah. That we'll never get our hands on almost. And if we have that happening inside the system, we'll also not get really get our hands on it, I don't think. And that should be okay.

Tim Scarfe 00:11:04

But I also love this deliberative sort of program that humans have. So we we ponder. So we just kind of iterate, and we just sort of, and then at some point we halt. And neural networks are always that instant intuition. They're always like, this is the answer. And I can imagine some kind of, you know, it's not necessarily neuro symbolic, maybe it is, some iterative approach of reflection, bringing in more contextual information, and then and then somehow arriving at the correct answer.

Will Williams 00:11:34

Yeah. I mean, this is exactly kind of how I see conversational AI of the future working is, you take in a frame from your environment every single step, and that should cascade through the system and update various things. And maybe you decide not to output any speech, you know, you just got zeros on the output. But then you go again, you go again, and that whole time you've had time to ponder, maybe your your high level representations are clock forward, maybe they haven't, maybe you haven't updated your high level beliefs. There's an opportunity to. So actually, I think we can, like, construct these systems in such a way that that's kind of possible. How would that work? Because I wanted to talk to you

Tim Scarfe 00:12:08

in general about the architecture of conversational AI. So at the moment, I talk to Speechmatics, and let's say I'm doing async, I have a stream coming up of audio data, and it's like PCM 16, you know, little endian. And then I have a stream coming down of JSON packets, which tell me like the time indexes and the words and so on. And this is just a constant stream. And it would be great if there was some kind of meta thinking process that could kind of like ponder and and update. Because right now, if I want to build a conversational interface, so I guess I have to do a lot of the work. You know, I I have to build the UI and I have to kind of like think about the meta model and the application structure. Would you ever design a system where

Enterprise voice AI and real-time translation

Tim Scarfe 00:12:49

more of that meta model kind of

Will Williams 00:12:50

happens on on your end? Yeah. So we're thinking about this. I would like an end to end kind of approach that that can that can do that. I think for the time being, cascaded approaches are gonna go a long way because you don't have to and by that I mean, you have an ASR system, you hook it up to an LLM, you have a speech synthesis system, you compose them, you try and reduce latency between them, And this works fairly well, actually, by the way. So you can get latency down to, 1.2 seconds, and then this is what we do is, like, we'll pad up to 1.2 seconds to just give a very consistent user experience. And that's and that's fine, but there's there's not much kind of meta kind of cognition going on in the LLM in this kind of scenario. So so to get to that you have to kind of flip paradigms to, yeah, a much more end to end approach and there's, yeah, some interesting research questions around how you how we get that that system doing what we want. I think a lot of the energy at the moment is trying to set up multitask problems such that you can do well on processing the audio, being able to exploit some of the paring linguistic stuff that happens in the speech and make use of it in some way that you wouldn't be able to make use of it. It was just converted to text and then put and sort of mirroring that on the output and producing sort of, yeah, speech with appropriate kind of emphasis. I think that will take us so far but there's still this it still won't have this kind of like meta reasoning that we you were just talking about. So there's there's a bit of a paradigm change that's needed there. And and I I have like a very very strong belief that until we can arrive at subword like level representations directly unsupervised from audio that we will struggle to do that. I just had another thought as well that at the moment there's an async service, you

Tim Scarfe 00:14:39

know, that there's like basically a streaming service from Speechmatics and there's a batch service. Yeah. And the batch service is much better because it has more more context or their streaming still really really good. And 1 thing I've been thinking about building conversational applications is having a kind of hybrid where because we want streaming low latency, we have the raw information from your stream, but then we do some kind of like retrospective multi scale post processing with an LLM. So after an utterance has completed you know maybe a couple of utterances ago we have another meta process which is running it through the LLM but the thing is the LLM might be a multimodal thing so it's actually looking at a screenshot of the UI I'm sending it the event log of all of the buttons people are pressing and maybe some, you know, external information, contextual information, and it's using all of that to kind of like improve the transcription retrospectively.

Will Williams 00:15:30

Yeah, sounds this sounds good. And people have done things like they'll take a snapshot of your mouth and they'll use that to improve the accuracy of the ASR. Yeah. How your lips are moving. And so like any of this additional metadata that can come in to improve the transcript, there's absolutely nothing stopping anybody going and doing that and it's probably a good idea, you know, there's a particular application in mind. Yeah, I've

Tim Scarfe 00:15:53

thought about that a lot actually. In the lip reading literature, I think the accuracy still quite low. It's about like 56% or something. But it's really cool having this multimodal approach. So I can see your lips, I've got your audio stream, and also you can use, you know, face recognition for better diarization. So I can I can enroll you, you know, so Will I talk with Will and I can tell when Will's talking and when Will's talking I use that to refine the diarization boundaries and so on? And it feels like the more modalities of data we have, the more we can kind of smooth off the the edges.

Will Williams 00:16:26

Oh, definitely. Definitely. I mean, you know, I'm a big proponent of, you know, having data streams come in the same as what we have as humans, know, a really rich source of audio and vision. And it'd be great if that was there by default for a lot of these applications, plus more probably.

Tim Scarfe 00:16:46

Yeah. Amazing. So so like in a future version of Speechmatics I'll be able to just give you like different multimodal streams.

Will Williams 00:16:52

I, you know, that would be great. I mean, if we can profitably make use of them. Absolutely.

Tim Scarfe 00:16:57

Yeah. Even stuff like the accelerometer on AR glasses. I mean, like, you know, so it's almost like spatial information, visual information, audio information, and maybe even social information. So maybe in the future we'll be kind of connected spatially to different people in the vicinity and all of this could just kind

Will Williams 00:17:13

of go up. Yeah, definitely. I think like some of these some of these modalities will just drop in there very, very easily. I mean, I'm, you know, I'm envisaging an unsupervised training process that can that can make sense of these different modalities and their interactions between them. I think a lot of these additional ones we can just bang them in and we'll be able to make use of them.

Tim Scarfe 00:17:30

What's your take on the whole LLM space at the moment?

Will Williams 00:17:33

There's obviously a lot of activity, a lot of really big models being trained, a lot of interesting economic activity, I think, around it. I I suppose the angle that I'm most interested in is how these models leaning us towards future more generally capable intelligent systems. And I do I do have some question marks around the amount of energy and almost fanaticism that's that's surrounding the scaling hypothesis. We just need to continue scaling this current paradigm. Next token prediction, transformers, Internet text. I I I do think that these systems will start to slow down in terms of the rapid progress that we've seen currently. And Yeah. Maybe g c 5 being pushed back is a bit of an early sign for that, I'm not sure. We'll have to see when that model drops. But yeah, these models are clearly very very good for certain activities, text processing, coding as being an alternative to Google in some cases, very, very powerful. But it's clear to me that there's a level of generalization that maybe isn't happening in them that that we're seeking ultimately and that we need to build bigger and better systems. Yeah. I would I put a question mark on the text Internet data piece in particular. I think there's a limited amount of statistical significance that you can get from adding more text tokens versus what we've got today. And maybe it has to be an exponential increase in data, I think the scaling laws will bear that out. There's a bit of an open question on how fast synthetic data can take us. I think it will iron out a lot of, you know, particular use cases where you want format your data with a certain table of contents or Mhmm. You know, a mathematical kind of application where we want to prove theorems in a certain way. Maybe synthetic data will take us a bit further, but it doesn't seem to me that the training on next the next token on Internet scale data, text tokens, is gonna unlock the kind of reasoning capabilities that, yeah, that lots of people want to see emerge. But but if they're honest, they haven't really been happening. There's lots of low low level reasoning that's once like, certainly is happening. But this this aspect of creativity, inventing new things, it just doesn't seem to be happening. And I don't know. I feel like there's a there's a voice in a community kind of polarization over this issue. 1 half is saying, of course, these things, you know, aren't doing that. The other other

Production deployment and testing infrastructure

Will Williams 00:20:00

half is saying, well, there is evidence of it and let's just scale it further. I think there's there's maybe a bit of truth on both sides there, but yeah, don't know. What do you think?

Tim Scarfe 00:20:09

Well, I agree. I think the problem is that in a way both communities are right. So the models are clearly just system 1 systems, they're not system 2 systems. And what's surprising everyone, you know, myself included, is how far you can get with system 1. So just this interpretive reasoning does actually give you the, you know, probably gives you the right answer for the wrong reasons, but it doesn't matter, it gives you the right answer a lot of the time. And as we memorise more of the long tail, know, it increases to, you know, its performance increases still further. And then there's the thing, well, if we wanted to get system 2 reasoning, this kind of creative reasoning that happens in the moment, how often do we actually need to do that creative reasoning? I think that's the question. Think they're still arguing that we can push system 1 a little bit further and it will just improve.

Will Williams 00:20:58

I suspect you can compile things down from system 2 to system 1 with sufficient data and the right setup. But that almost assumes that you've got the system 2 piece, and I think it will become exponentially more difficult to learn these behaviors.

Tim Scarfe 00:21:12

It's weird. Ryan Greenblatt said something interesting the other day. He said that you can always get a dumb system to approximate a smart system if you just throw lots of data at it. And that's kind of what these things are now. Mean, know, these LLMs are idea generators. And quite often the right answer is in the possible set of trajectories, so you just need to keep asking it, and then you can hack together a solution with tool use and evaluation and selection and so on, and you can often get the right answer out of it. So because these systems are so flexible and hackable, it's possible to just make them work a lot of the time. And in a way, that's something that's quite tragic for me because no one's looking at any other forms of AI now.

Will Williams 00:21:50

There there is a certain level of tragedy around it. But I I I do think we need to push the current paradigm to its limits. I guess you get into the finer questions of how much do you wanna then spend on a a big training run. But I think we need to just realize by almost by construction that that once we scale these to a certain limit that we're not able to extract maybe more

Tim Scarfe 00:22:13

out of them. Yeah. And I think we're gonna see that soon. Mean, when I spoke with Aiden Gomez, he actually said to me that more than half of their data at this point is synthetic data. And I don't know exactly what he meant by that because it's not it's not easy to generate synthetic data for language. I mean, maybe they've got a whole bunch of programs that do stuff like that, but it's already like a massive amount of synthetic data. And I'm guessing that we'll see some kind of saturation with with the language models. When I speak with people like Sarah Hooker, she's already saying clearly that, you know, in multilingual models, they're not really generalizing very well between the languages, you know, it is just a case of like learning these distributions. But because they do work so well when there's this Mechanical Turk thing going on at OpenAI, there's the perception that, you know, they're better than they are perhaps. But are you worried about this? So when you see these multi-modal GPT 4 point type models, you know, lots of people might naively think, oh, you know, we don't need Speechmatics anymore. Mean, what what would you say to that?

Will Williams 00:23:12

I'd say run the word error rates, to be honest. Do the eval. Yeah. Because I think if we want to have a platform, like a voice platform, which with which to interact with machines, we want to have some level of guarantees around accuracy and whether it's gonna get my words correct. If if all my frames of my sort of speech are being bunged into a giant attention mechanism and some of them are being dropped or some of them are being misrecognized, I'm not sure how comfortable I am with that. Certainly, I can see that the speech problem isn't solved, and there's a lot more there's a lot more to do.

Tim Scarfe 00:23:46

Can you see a future where you I was playing with Moshe the other day, which is like this new, you know, bidirectional low latency model, and it's the first time I've seen a bidirectional model for, you know, that does ASR and language modeling and so on. Super low latency. It wasn't very good. It hallucinated and did lots of crazy things actually, but, you know, but as a kind of, you know, template for conversational interface, was quite cool. But, you know, could you imagine a future where OpenAI just completely changed the architecture? They had a bidirectional model, you know, that had like real time input and output and so on. Like, at that point, would you be worried?

Will Williams 00:24:19

I think I'm only worried when I see the numbers. Yeah. To be honest, I need to need to see the hard facts. I'm not sure those changes would necessarily get them to a system that does it for example, better live captioning. Mhmm. You know, doing ASR in real time. This is 1 of our differentiators like accuracy real time, low latency. We've seen very good numbers there and, you know, I'd wanna I'd wanna see a big shift there before I I really got worried, to be honest. Yeah. I I think these kind of, like, full duplex conversational assistants are very, very interesting. I do think conversational systems should be modeling their own voice, for example. So if they're speaking out and they can't hear themselves talking in the room because it's too quiet, they should automatically modulate their voice up. This is the type of thing that we should see happening in systems. We just don't. I mean, no 1 thinks about So, yeah, being able to sort of model your own voice and have this kind of bidirectional interruption possibility is, like, is very interesting, but there needs to be heck of a lot more smarts, like, sitting behind it than currently. Yeah. I think in particular demo that you mentioned, there were lots of times where you couldn't really finish your sentence because it was trying to interrupt you and talk over you. And so that experience is like very unnatural. And then there's a problem of having these systems kind of maintain their LLM knowledge and and how well it does how well it does that. So I think there are challenges there, but these are challenges that we're interested in working on. Are you interested in augmenting your models with LLMs in any way? Yeah. So we we know that LLMs have very good sort of long term consistency, you know, so if you have a very long transcript, the subject matter can be ingested by the LLM, it can understand what topic you're talking about, it can understand the finer details, it can go fix up, you know, Selim's stakes. So this is, like, this is really good. And there's other workflows that where LLMs can get involved on transcripts, which make a lot of sense. We already support some of these summarization chapters. We're we're looking at ways to integrate LLMs more into our core pipeline because they clearly have a good level of world knowledge that

it'd be very useful for SL to have.

Tim Scarfe 00:26:19

And would you train your own LLMs or would you use another provider? We would probably use open source models

Will Williams 00:26:27

for this because there's there's a certain threshold beyond which they're they're useful for this kind of like world knowledge. And so we need to they need to be large models and therefore we're not we're not gonna be training these ourselves from scratch. We may want to fine tune them.

Tim Scarfe 00:26:41

When when you're developing these models, there must be a massive trade off between, you know, how much it costs to build the model and and its performance. How do you think about that?

Will Williams 00:26:51

In our case, we're training models that are, let's say, under 10,000,000,000 parameters. So so the train the train costs are not too prohibitive, particularly given, you know, we've got we have some costs in terms of an internal cluster. That sort of things is fine, but the economics of inference is interesting. So what kind of performance do we wanna offer? What's the word error that you can get per dollar? These types of questions, density, latency, we'll we'll look at

Speaker 3 00:27:19

all of

Will Williams 00:27:19

these. And so 1 of the things that we've decided to do is have multiple operating points that sort of sits at different points on this on this kind of curve. So we'll have a smaller model that's higher throughput, it's more efficient to run, slightly cheaper, And then we got our kind of our premium enhanced model where we want to just optimize for accuracy almost at any cost. We wanna just make sure that is as accurate as possible. And so, you know, we think there are a couple of interesting points to sit on that curve And that's very much guided by our product team.

Tim Scarfe 00:27:55

What do you say to people who argue that speech tech is just becoming commoditized? You know, like with the the ImageNet thing, it was kind of seen as

Will Williams 00:28:03

a solved problem at some point. Do you think we're anywhere near that with ASR? I think for batch English, it's getting closer. I definitely know the ways in which it fails because I'm obviously a big time user of this tech and I'm testing it all the time. So I know that for conversational AI, we're we're still not quite there. But definitely more more than anything, batch ASR has has been just big advances in the field, which has been great to see. I think non English languages are lagging behind by like 3 or 4 years, roughly speaking. And then maybe real time is is like got a 3 year lag on a 2 year or something like that. And yes, so people are struggling to do to maintain this kind of accuracy in real time, which obviously becomes relevant when we're starting to build conversational assistance, which you know we're starting

Tim Scarfe 00:28:51

to get into. I'm interested in your articulation of where, because you said you know where the failure points are. So for example, when I was building x-ray, we noticed that it was quite bad in noisy environments. But we also noticed that, you know, Speechmatics does an incredible job of like lifting low speech. So we didn't need to do any audio processing. We're not like, you know, normalizing the audio. We're not pre filtering. Suspect that if we did something like source separation, you know, because like the cocktail party problem is is a big 1 where you have multiple people. And what we noticed is that some of the models seem to implicitly do something which is akin to active source separation. So for example, when when I start speaking, it seems to lock on to my voice and ignore the voices in the background. And this is presumably something that you haven't intentionally

Model architecture and latency-accuracy tradeoffs

Tim Scarfe 00:29:38

done.

Will Williams 00:29:38

Yeah. This is an artifact of the training data, almost certainly. So, yeah, it will continue with the the dominant speaker. Yeah. But this ability to do source separation and deal with crosstalk is is a big area where the the current generation systems are just like architecturally not there. Obviously, papers that deal with this in particular ways. It's a big kind of failure mode, you know, as humans we can decide I want to attend to this particular voice and do a really good job of recognizing it and continuing the conversation. These models are very very brittle on that right now.

Tim Scarfe 00:30:12

It's actually interesting because there's a whole cottage industry of startups and their sole purpose is to pre process audio data pre ASR. And I'm assuming over time that, you know, the benefit of that is getting smaller and smaller because the the ASR providers will just kind of do it implicitly. You know, I

Will Williams 00:30:32

was poo pooing supervised learning earlier, but here's here's where it kind of comes to save you a little bit. If you don't pre process your audio and you have, you know, really big distribution, you're going to cover all of that.

Tim Scarfe 00:30:42

Yeah. So I'm thinking as well, like a future scenario might be with active inference, you know, customers might be wearing, let's say, AR headsets and those headsets might do adaptive beamforming and they might do a whole bunch of stuff and then that could be you know almost an active signal which is modulated by the ASR algorithm. So we're getting into much more kind of situated physical intelligence.

Will Williams 00:31:04

That would

Tim Scarfe 00:31:04

be very exciting. Beamforming is another big 1, there's like, you know, TDOA and we looked at that and also saw separation.

Will Williams 00:31:12

Yeah, that's really cool. Yeah, I'd be interested to know how far you got with that actually.

Tim Scarfe 00:31:16

Yeah, yeah, not very far, but but the the phones do it. So for example on on Android, it's all built into the audio APIs and they actually have a feature called mic microphone zoom. So that means that when you when you set the zoom to 2, it's already on your camera by the way, so on your camera when you zoom into someone over there, it's doing beamform on the audio, and it will actually, you know, ignore all of the people in your vicinity and plug into that person, and you can do that programmatically through the API. The problem is it's useless from an audio app point of view, because you're not going to be like, you know, pointing your phone in a certain direction. Zooming in, yeah. But the thing is, you know, people have their phones, they don't have powerful directional microphones. Mhmm. So, you know, like a lot of like deaf folks, for example, they want audio transcription. They probably have to buy specialist directional microphones and it's not very practical. But hopefully these AR glasses will help them with that.

Will Williams 00:32:06

Yeah, that's cool. How often do you release your models and how stable are they? We do weekly releases, they may or may not include a major update to our most significant models. Those those models will probably be updated on a sort of cadence of 3 months, maybe slightly longer depending on, you know, on projects that are backing it I suppose. As for stability, we have we have a lot of internal testing. We have soak tests in real time, have unit tests, functional tests, lots of performance tests, that's a big focus right now. So we can make sure that we're operating at the accuracy that we claim at low latency and we have lots of word error rate testing. So by the time we've got a release candidate that's been sort of approved and ready to go, it will have it will have had a real battery of tests and something we've been building out over the over the years on the machine learning side is getting a really wide set of test sets to make sure that we're not aggressing on our sort of core metrics which in ASR is word error rate. Do you have like a rollback strategy?

Tim Scarfe 00:33:09

Mean what happens if something goes goes wrong?

Will Williams 00:33:12

Oh yeah, so I forgot to mention 1 of the things we do is we'll run-in run-in a mirrored environment first. So so we'll deploy our new new models directly to environment which takes mirrored live traffic for a period of time and there's lots of weird and wonderful ways in which people can send audio files in which just bring unexpected results. There's lots of configs that are interesting. So that has been a surprisingly effective tool for us is is having mirroring and, yeah, I I would love us to bring that testing closer to closer to the build and not necessarily have to have it so we can release quicker, but it just catches so much that it's been really powerful for us. Yeah. As for rollbacks, yeah, we can we can roll back to our previous version. So we've got quite a lot of DevOps maturity on that side. So we're not afraid to go and go and do a release and we can canary it as well with certain parameters should we need to.

Tim Scarfe 00:34:12

Yeah. Having having these models they're huge right and if you wanted to keep a couple of older versions up and running because let's say 1 customer or a few customers had a big problem and you might want just them to go back to the old version but everyone else uses the new version would you be able to do

Will Williams 00:34:26

Yes we can and we do. So sometimes there are just circumstances where yeah we'll want to just be a bit more careful with a certain customer maybe because they're waiting for a certain feature that's they're about to release and give them early access and then give them their own sort of upgrade path.

Tim Scarfe 00:34:46

So yeah, it's really good having flexibility there. Another interesting thing is that some of your customers choose to fine tune the models on their own data. Is that correct?

Will Williams 00:34:56

That's not correct.

Tim Scarfe 00:34:57

That's not correct. So I'll let it that out.

Will Williams 00:34:59

Well no, it's really interesting discussion. So it's been an ongoing strategic discussion whether we should allow customers to bring in their own data, point in the model, get a ostensibly better final result. And, you know, we've gone the route of building global models where we'll we'll try and get as much data as we can that's as representative as possible and make it so that fine tuned model is just as good as our original global model. And on the whole that strategy has has worked. I guess in machine learning, if you do have like a prescribed domain, you can fine tune and get a get a better better model.

Lattice-based decoding and language model integration

Will Williams 00:35:40

But looking at the maturity of ASR systems, it's not a massive gain and and it's not and the complexity that would come with that is not a brilliant trade off Mhmm. In my book. We've put all our energy into to getting that 5% for our customers by instead of building that whole system where you can go train it and fine tune it yourself, putting it into general gains. I'm just pushing accuracy really hard across the board and I think that's worked well for us.

Tim Scarfe 00:36:07

Yeah, it must massively increase the the architectural complexity, especially if the models are big, because you need to load them into memory, and then they might not be used all the time. Because if you have like 1 big shared model, you know, you can have elastic auto scale and pooling, and you know, you've got all the people using the same model. Now all of a sudden you've got this weird esoteric model that 1 customer uses every Friday afternoon and you don't know whether to load it into memory and you've presumably got to replicate it to different data centers in the world and so on, makes it really difficult.

Will Williams 00:36:35

Yeah, it'd be expensive to host. I think it makes more sense on prem, for sure. So if, you know, customers are to run it themselves and they're going to run it with just their traffic, you know, you can get good utilization there for them, which is nice. But yeah, if we're running in our SaaS alongside of all other models, that becomes painful at that point.

Tim Scarfe 00:36:52

What is your sort of high availability and globally distributed setup look like? So I guess you've got servers in different countries and in your release schedule they'll automatically be kind of distributed to different countries and ASR especially the, you know, the streaming version, it's very latency sensitive.

Will Williams 00:37:08

So people want to use their their local servers. Yeah, that's right. Yeah. So we'll we have 1 in EU, we have couple in The US, we've got 1 in Australia as well, and we can send things up on demand for certain customers as well if we need to. So yeah, we're we're sensitive to that and as our customers, so we will deploy to a number of different environments.

Tim Scarfe 00:37:26

Interesting. Tell me about latency.

Will Williams 00:37:28

Yeah. So latency is a really interesting topic in ASR because nobody sees low latency numbers when they look at word error rates. There's only the sort of full batch, full fat system that, you know, if you're running captioning, you're never actually gonna use. Something we've been pushing internally quite hard recently is looking at latency accuracy curves and, you know, we want to be sort of free to optimal there. We want to sort of dominate the competition. And so yeah. So we've been we've been looking at our accuracy at various different latency points and and doing some things to really push that curve down into the left. Yeah, that's been a very fruitful exercise and it's something that I think the industry and academia probably need to, you know, do a little bit better at reporting is, okay, what is the accuracy of the system when we're running it in real time at a reasonable latency? And very,

Tim Scarfe 00:38:23

very few people do actually quote on that. Yeah. You said something interesting as well because you're talking about latency curves so a lot of people might not appreciate this that you can actually set the latency so you can set it to be let's say 300 milliseconds or 500 milliseconds and the longer the latency is presumably like the more information you have. So how does that work? I mean on Speechmatics at the moment is it fixed or can

Will Williams 00:38:46

the customer set it? Yeah, customer can set it. Alright. I think at the moment you can go down to I don't know point 0.7, 0.5 seconds and still get reasonable performance before it all falls off a cliff. And then you recover your full batch performance by moving out to like 4 or 5, 6 seconds. Something like this. And so there's a graceful sort of like transition between those 2 sort of modes of operation. But yeah, the idea is that if you set a lower latency sort of target that the system has to sort of output its current best guess, it's also probabilistic. And so, you know, you do have a current best guess. But if you give more time, you can bring in things like a a language model that can update all your hypotheses, give you hopefully a more refined score on on what you've been saying. And so you do typically feel the difference of waiting a little bit longer. And 1 of the things I think is interesting to think about is, well, as a human being, how would I transcribe this optimally? What would my latency be? If if I could immediately just choose a word and write it down, what would my latency be? And actually sometimes it pays to just wait a little bit to try and, yeah, disentangle what somebody has said and that's no bad thing. And I think the same thing goes for the ASR system. We don't really want systems at 200 millisecond latency. That's just not useful. You're gonna get get mistakes that if you set

Tim Scarfe 00:40:10

it that low even with like a perfect system. You said something interesting about having like a language model do some post processing. How does this work? I mean, are your models just like, you know, you have the waveform domain and you might do some feature extraction and you have this model and eventually you have words coming out the end and then you're alluding to you know a secondary model almost improving the results of

Will Williams 00:40:31

the first 1. The big picture is that we have so we have a lot of text and everyone who's building LLMs out there is going to obviously be aware of this. But there's information that you want to exploit if you're building a speech system and actually not to do so just to put yourself at a disadvantage. So we need some way for this kind of knowledge of how words follow each other Mhmm. To be integrated into the system and the different ways of doing that sort of distinguish the different ASR systems, so people have tried lots of different ways of getting that knowledge in there. Yet 1 of the ways that we do it is we we build a lattice that has like a directed graph that has all of your hypotheses contained within there and we march over it with a language model arc by arc and we will update the scores. This is like the full fat version of ASR. It's it's maybe it adds a bit more to your latency to do it that way but it means that we can, you know, you can you can inspect these, you can sort of you can do lots of good things with that, there's actually lots of interesting FSD operations on them. But yeah, it means that you can bring a big language model in there or an external language model as it gets called and update those scores. And you know that language model can be as big as you like because, you know, there's no sort of restriction. And so yeah, we really should be seeing the power of these LLMs coming to the ASR world. Yeah. Can you spell out

Tim Scarfe 00:41:47

to the audience what what a lattice is? So you know, like a grid is is a type of lattice. This might be quite strange for the audience because you know they think of language models as just almost being over a flat sequence. Yeah. But you can actually have some structure in that. So you

Will Williams 00:42:00

have some some nodes and arcs and you can imagine transitioning across an arc if you sort of consume a a particular word. What ASR kind of does is it it it considers the infinite number of paths that could exist if you progress from word to word. So so it's time based. So at any given moment in time, maybe I've spoken a few different words and you're not sure. In fact, it could be an infinite set of words, but the recognizer will then pair it down and hopefully just output the ones that are probable. And so so you can follow your way through this, like, this lattice to an end state, so you have a start and end state. And all the different pass through that should be fairly legitimate sentences. Mhmm. So it's not just a fixed grid. Maybe things converge at 1 node halfway through because, you know, that is like, it's dead certain that that's what's been said and then it goes on again.

Tim Scarfe 00:42:51

Does this happen in in a stateless way? So we're doing transcription, we have this lattice, we do some kind of optimal traversal, we pick the next word based on, you know, some algorithm and then we have another load of audio data come in, and we go through the whole process, we do the feature extraction, you know, we do some predictions, we do the lattice, we do the traversal again, so it's almost like an autoregressive language model, and every single step is stateless, so it almost it's not in any way, you know, affected by the step before other than the result of the step before. Yeah, I

Will Williams 00:43:23

mean I guess it depends how you how you wire things up, so you could technically make it stateless, Your acoustic model is going to have some kind of horizon, some context length, so is your language model, you know, it could be attending to something you said like half an hour ago, that's technically possible. And and same for the decode as well. So the way that I think about ASR is kind of you have these 3 components that have to come together in some way, which is like the acoustic model, language model, and the decode which is like a search process, which will produce those lattice. And they each can have their own state. So they can be incrementally sort of updated.

ASR performance metrics and real-world evaluation

Will Williams 00:44:00

Oh, that's interesting. Yes. So all of

Tim Scarfe 00:44:02

them are parameterized, and those parameters kind of drift over time. But but the interesting thing is that you are actually doing, I guess, some kind of a beam search or some kind of a future search. So you're you're kind of predicting what could the future look like many steps ahead, and then you choose 1, and then the next step, you're kind of starting again, and now I'm looking at all of these future trajectories. Yeah, exactly. Yeah. Yeah. That's pretty fascinating. And then and then it just works.

Will Williams 00:44:28

And then it just works.

Tim Scarfe 00:44:30

Just like magic.

Will Williams 00:44:31

Yeah. Yeah. It definitely didn't feel like magic, I don't know, 10 years ago. It was the word error rates were very, very high. Know, like 35 percent and now we're down like 5 percent or something. It's just a rough sort of picture. So it does feel more like magic now

Tim Scarfe 00:44:47

for sure. And these are probabilistic systems and I was talking to Murray Shanahan and he was talking about this thing called Loom which is where you can look at all of the conversation trajectories. So you could ask an LLM 20 questions, and you can say, you know, think of a thing, and then you can kind of go through more of the questions. And you think that the language model has internal consistency, and the thing that it actually thought earlier on will be what it said later on. But any of the subsequent steps can actually make it diverge and say something completely different. But we look at it as if it had some self consistency, which is kind of weird.

Will Williams 00:45:17

Yeah. Yeah. And you have those exact same branching kind of process happening in the search phase in ASR. You just prune away the stuff that wasn't likely under your language model and wasn't likely under your acoustic model. So these things are just probabilistic models? Yeah.

Tim Scarfe 00:45:32

They don't reason.

Will Williams 00:45:36

Wow. There's a big discussion. I don't think they do. Good. I think we're just I mean, this is the thing with supervised learning. We are just learning a mapping. And maybe there's some very low level forms of reasoning that you could say exist there, but this is very much a dumb mapping. And I think this can be seen by the the levels of generalization that happen. So, you know, 1 of things that we've been talking about previously is out of distribution generalization, how it's not really happening in ASR and the way that people believe it is. I think that's a bit of a clue that there's not the type of reasoning that we'd recognize happening in these systems at all. When they generalize in ways that we generalize, I think we can maybe revisit that question. But when and until they do, I've I'm very very skeptical that there's much reasoning happening at all. So the question is why do they generalize?

Tim Scarfe 00:46:27

So I mean I speaking with Sarah Hooker from Cohere last week, and she's really grumpy about the EU AI Act and the executive order in The States because they've just set a threshold of the number of flops above which models are dangerous, and below which they are apparently not dangerous. And there's this idea that there's some commensurate relationship between risk and generalization and the number of flops, the amount of compute power that goes in. And she argues that the reason for that is, well yes, of course, there's a commensurate relationship between compute and capabilities because the models are just memorizing more of the long tail. And people look at that and they confuse it with generalization ability, or OOD. But what's your take?

Will Williams 00:47:09

On the AI safety question and what how we should be regulating it, think, I mean, my view is that we should be a little bit overcautious to begin with in general. So I wouldn't be completely poo pooing, you know, these new moves, but it it does feel quite over the top in general. But we're having some of these conversations because we don't really understand the types of generalization we're shooting for and when it's emerging and to what extent and how do you measure that. I think tying it to flops is is questionable, you know, I can it's a it's a bit like saying in ASR, I've got a great turnaround time, I can turnaround time, I can turnaround things in naught point, naught, 1 real time factor RTF. So, well, great. Did you actually return me the correct words? Were were they actually in there? And it's the same thing with flops. Well, great. You can run a huge amount of matrix multiplication, but what's actually happening behind that? And we're very long away from consensus on that. So it it like I say, when there's bit of uncertainty, I'm actually okay with us being a little bit of like over the top with regulation. But it has to be paired back quickly and that that's as we learn more. And I don't think that's going to happen. So that's where my my grumble would be.

Tim Scarfe 00:48:23

Yeah. Because she was saying, I mean, she trains multilingual models. So she has to do more compute because she's actually learning completely different distributions of data in different languages. And it is a simple case of I've just got more stuff to do, I've got more stuff to And the problem seems to be whether or not they actually do some abstract generalization between those distributions. What's your take? Mean, because obviously, it's interesting, right? Because you actually work at Speechmatics, you know, you're building these systems, you've seen behind the curtain, you know how they really work. Yep. Are they are they doing magic generalization or are they just like memorizing more of the distribution? There's some

Will Williams 00:49:02

level of generalization happening. I'm 1 of the biggest testers of our system, so I can see, you know, some of that impressive generalization that does happen. We do we get the right words on on distributions that we've never trained on, which is good but I think people are worried about a level of generalization that that goes way beyond that which I haven't seen. I mean, I think 1 of the 1 of the simple ways to evaluate this stuff is to do it qualitatively. I think you're a big proponent of this as well is I'm gonna take the system, I'm just gonna interact with it very simply and you can get a good feel for the level of generalization that that exists and I'm not sure we're on a trajectory into dangerous territory there. And so yeah, and also just having it as a kind of straight flops calculation like Sarah I guess was saying just makes just makes no sense.

Tim Scarfe 00:49:52

Yeah. I mean, with Murray Shanahan moved me a bit on this because, know, you you can make technical arguments about, you know, the limitations of reasoning. You know, they're clearly doing a colloquial, interpretive reasoning. But that actually works really well. And when the model gives you the correct answer, I don't think it's necessarily correct to say it's not reasoning. You know, it's arrived at the correct answer. And I really liked your point about the qualitative evaluation of models. Because, you know, we've got many ways of evaluating language models, for example, like LLM Cicerino, and so on and so forth. But it's the same thing with Speechmatics, right? So I built a, you know, a speech transcription company which was like, you know, using various different APIs, and in my opinion, many of the companies basically lie about their benchmarks. And I would just use the thing. And the thing is as well, benchmarks, they generalise over everything, like real world usage. I'm actually using the thing in my noisy cafe, or in my bedroom, or talking over the phone with someone, and the results are plain to see. It either works or it doesn't, and it's usually obvious which is best. Even when I use language models, I use like, you know, Sonnet 3.5, love it, you know, GPT 4 row. You can just use these things, and, you know, maybe maybe like I'm deleting myself a little bit, but I I can feel it. I can feel the difference. Can you feel the difference?

Will Williams 00:51:09

Can you feel the AGR? Yeah. Think I can. I I think I think I think you can roughly tell how a system is doing just by using it. Yeah. We are very good evaluators of these systems and that's underappreciated. I definitely think benchmarks have that place. It's almost like they're sort of, you know, sort of

Ethics and privacy in speech technology

Will Williams 00:51:30

necessary but not sufficient. Right? We want to be passing our benchmarks that we care about, that we've well on previously Mhmm. Particularly at Speechmatics. But it's not it doesn't tell you everything. We've seen this actually when we we made this sort of a change to use self supervised models in our system. And we did the training, we did the evaluation, we had really happy results. And then it was only like a few months later that we saw someone put some children's voice through and it started to recognize it correctly. We didn't we didn't have it in our test sets at the time. And so it was generalizing in surprising ways that was quite exciting. And so, yeah, benchmarks are definitely not gonna catch everything.

Tim Scarfe 00:52:08

Yeah. And there's this kind of Goodhart's Law thing as well, is when, you know, when a target becomes a measure, it ceases to be a good measure. And the great thing about, you know, subjectivity and evaluation is that everyone has access to a a source of randomness and novelty, which means, you know, you're actually getting very useful signal there. But then the good thing about benchmarks is standardization. You know, which is that you have this kind of this 1 golden mark that, you know, that rules them all. So it's quite difficult to interpret, you know, the subjective reports of of many people. Mhmm.

Will Williams 00:52:39

Yeah. And and I think benchmarks drive progress really well when you're in a certain regime. When you're in this sort of like ImageNet and LibriSpeech kind of space where they've been like flogged to death in academia, there's just less signal. So it's like, yeah, you go through its early phase of lots of signal and then less thing, you have to go replace it with the next thing. That's really important to recognize when that's happening and move on. And, yeah, I think as a speech community, I don't think we've been very good at moving on from certain test sets, for example. In fact, things are no longer test sets as soon as you've used them for a certain period of time. And this is just like, this is fundamental to appreciate that that people are ultimately going to be optimizing against that test set result by the sort of meta decisions that they make, the architectural changes that they make and so on.

Tim Scarfe 00:53:28

Yeah. I know that I think this is something that people don't realize is the amount of information that leaks from the test set, you know, sometimes indirectly. And Francois Chelet knows this. So this is the reason why he has fastidiously guarded the private, you know, ARC test set. And even then though, it's still kind of leaking, you know, because people are starting to make inferences about what's in that set based on generalization patterns in in the public set. And eventually, 1 day, there'll be a spark because someone will figure out a thing that works on the private set, and then it's basically it's gone because test sets rot over time. Yeah. Yeah. Yeah. So so what do

Will Williams 00:54:05

you do? Do you guys have like a private test set or? We have a whole bunch of private test sets, you know, that we we trust. And, you know, we'll run them against our stuff. We'll run them against competitor stuff. So we really know what's going on in in the market and how well we're doing internally as well. And it's important to keep adding to that because you got to be real with yourself that you're not, you know, you haven't suffered this kind of leakage problem even if it's very indirect.

Tim Scarfe 00:54:29

How do you mitigate that? Because I guess even in your internal testing, how do you not inadvertently kind of overfit on your private sets?

Will Williams 00:54:37

Well, we have to have checks for the leakage and that we're, you know, there's there's no overlap with our training and test data, so that's really important. But beyond that we want to look at the distributions that we're training on, making sure that we're not overfitting to 1 1 kind of distribution. So yeah, that kind of goes into planning but it kind of goes along with our overall approach that we want to continue to widen that distribution because as long as you're sitting with the same distribution for a long time, you are susceptible I believe.

Tim Scarfe 00:55:05

So Will, are you going to give us a demo?

Will Williams 00:55:08

Yeah. Well, I've got a few demos lined up. 1 of them is on the laptop here. Should just be transcribing our voice and doing some live captioning which is, you know, 1 of the the big use cases that we have at Speechmatics.

Tim Scarfe 00:55:21

Amazing. I didn't tell you the story but when we last interviewed Noam Chomsky, we used Speechmatics in in x-ray and we were actually sharing our screen with the transcription and it was only because of that transcription he was able to understand us because he's profoundly deaf. Oh, wow. And we're going to release that video soon. But it was thanks to Speechmatics.

Will Williams 00:55:41

Wow. That's really cool to hear.

Tim Scarfe 00:55:43

Yeah. So we've we've I guess we've built a similar app to this. But yeah. So let's look at the accuracy then. What what's it doing? The latency is incredibly good. Yeah.

Will Williams 00:55:52

This isn't released yet. Maybe it'll be released by the time podcast goes goes out but the latency I think here is at 0.8 max delay.

Tim Scarfe 00:56:00

No. 2.8?

Will Williams 00:56:01

Yeah. I think so. Yeah. Which means some of it will be coming back faster than that. We've also got the partials turned on so you can see how fast the engine is actually operating behind the scenes.

Tim Scarfe 00:56:09

Yeah.

Will Williams 00:56:10

Some of our customers won't wanna use that at all actually if they're doing live captioning, but it's it's useful just to get a feel for how the engine works.

Tim Scarfe 00:56:18

Yeah. I love having the partials on because, know, from a UX perspective it makes it feel much more, you know, responsive.

Will Williams 00:56:24

Yeah. I'm quite used to reading them, so I quite like them, but I know there are some people who it just kind

Tim Scarfe 00:56:29

of messes with their brain a little bit, so. Oh, yeah. Well, we we render them on X-ray. I'm a big fan.

Will Williams 00:56:34

You know, we want to have a really consistent API. So this is 1 of the issues actually with Whisper, is we we iterated through numbers from like 1 to 10,000. Mhmm. We synthesized each number. We pushed it through Whisper and it just hallucinates on a whole bunch of numbers. So there's like some numbers that you cannot get correct with Whisper. Really? Yeah. And maybe it's been maybe it's been all fixed since we ran this on, like, in v 2. But but yes, that's that's like, yeah, a good example of the hallucination problem. But we so what we'll do is we'll we'll recognize the spoken form and then apply like a deterministic inverse text normalization function on top of on top of the words so you so you can get things like 3.141 rendered correctly.

Tim Scarfe 00:57:19

Wow. That's interesting. Do you think they now do, you know, synthetic data for audio? Because they presumably didn't on Whisper 1.

Will Williams 00:57:28

Yeah, I think you would probably need synthetic data to cover some of these edge cases.

Tim Scarfe 00:57:34

Yeah. Yeah, that's so fascinating that because professor Christopher Summerfield called it the Swiss cheese problem, which is that, you know, sometimes you're in a hole in the Swiss cheese, which is to say you're you're in part of the distribution where there's no data and then all of a sudden the models just don't work.

Will Williams 00:57:50

Yeah. I mean, this is this is where end to end models do actually have a disadvantage is that you have to make sure you're covering that distribution really like with the audio itself. Yeah. Which can be quite costly and you know, you just have to go collect that data if you need it or synthesize it.

Tim Scarfe 00:58:06

Very cool. So this new model, will it work with your existing API so people can just switch over to the new model?

Will Williams 00:58:11

Yeah. This should just be a an immediate change that appears on on our cloud.

Tim Scarfe 00:58:16

Wow. Very cool. I want to use it. Can you give us sneaky squirrels access to it?

Will Williams 00:58:23

Yeah. Almost certainly.

Tim Scarfe 00:58:24

Yeah. Yeah. Amazing. Yeah. We almost need some esoteric words like super fragile what was it? Supercalifragilisticexpialidocious.

Will Williams 00:58:35

I think he's got it but it's split it into 2 words. I'll take I'll take that.

Tim Scarfe 00:58:40

Yeah, self attention transformer theory, bidirectional transformers, neurosymbolic architectures. It's not bad. I mean, I must say we are using incredible microphones here. That's gonna help. You know, I think the microphones almost transcribe themselves.

Speaker 3 00:58:56

They transcribe themselves. Yeah. Yeah.

Will Williams 00:58:58

I'll take that.

Tim Scarfe 00:58:59

I wonder if is there a way that we could give it a deliberately difficult problem? So this is what speech sounds like to a hearing aid user in a busy environment.

Will Williams 00:59:31

Providing fair, high quality service to this large group is becoming a high priority for any service provider organization. That's pretty much perfect, isn't it?

Tim Scarfe 00:59:42

Honest. I'm very impressed with that. Yeah. Because, you know, I think this is the most impactful application of the technologies, you know, hearing impaired. 1 100. You know, that was totally what we are focused on.

Will Williams 00:59:52

Yeah. Yeah. Yeah. That's really, yeah, it's really exciting to see it being used in those kind of important ways. That's amazing. I wonder if

Tim Scarfe 00:59:58

I can find an example of crosstalk. We're doing exactly what we talked about

Will Williams 01:00:01

earlier, which is qualitatively testing a system.

Tim Scarfe 01:00:03

Yeah. I know. Isn't it great? And the other thing, by the way, is you know I'm playing it out from the speaker of my iPad, and I've noticed experimentally, perhaps you have as well, that it's dramatically different from a voice in the room. So I don't know whether it's to do with some Moire effect.

Will Williams 01:00:20

Something like that. Yeah. Yeah, there's definitely effect that hasn't been very well accounted for.

Tim Scarfe 01:00:24

Yeah, because 1 thing we were trying to do for months is just come up with ways of doing automated testing. And so we got 1 of our staff members to read out a whole bunch of, you know, things. And we found out that the the placement of the microphone, whether or not it was being played out from a speaker, what kind of speaker it was, just completely transformed the Oh, wow. The performance downstream. So it it's almost impossible to test this

Self-supervised learning and low-resource languages

Tim Scarfe 01:00:50

stuff, basically.

Will Williams 01:00:51

I didn't realize how deep you've gone into this. Yeah.

Demo Clip 01:00:56

Paid few taxes or lost 1000000000 dollars a year. You you are Donald Trump Trump's apprentice. Let let me talk about this.

Will Williams 01:01:04

Senator, I think I think I'm still on my time.

Demo Clip 01:01:06

Well, I think our isn't this a discussion? This is our Yep. Discussion. Let's talk about this. Well, is let me let me interrupt let me interrupt you and finish my sentence if I can. While she was secretary of Now I can wait. Had a private server I get to wait and was discovered I did raise senator keep that pay to play process governor Pence

Will Williams 01:01:24

Because Hillary Clinton failed to renegotiate.

Demo Clip 01:01:27

More American troops. Hillary Clinton Hillary Clinton Hillary Clinton failed to renegotiate. The status of forces agreed. No. That is incorrect. And so we remove we remove all of our troops from Iraq, and ISIS was able to be injured up in that vacuum But but I'd like to correct over on vast areas of I

Tim Scarfe 01:01:44

have to say, I am bloody impressed with that. That is remarkable.

Will Williams 01:01:49

It's done okay there.

Tim Scarfe 01:01:51

My congratulations to you.

Will Williams 01:01:52

Yeah. Thanks.

Tim Scarfe 01:01:53

Yeah. I mean, as as a veteran user of ASR tech myself, I am bloody impressed with that. I I wanna I wanna use it. Please use it. Give me access

Demo Clip 01:02:03

to it.

Tim Scarfe 01:02:05

But it's

Will Williams 01:02:05

the Damn. The crosstalk problem is still there though.

Tim Scarfe 01:02:08

I mean Oh, yeah. Yeah.

Will Williams 01:02:09

But the thing is it's it's hard for humans. Right? So Yeah. You know, at what point do you say, look, this part of the problem is AGI complete and we're done. You know, let's just wait for that final system.

Tim Scarfe 01:02:20

Yeah. Because like the human brain has this remarkable ability to kind of compartmentalize. It's almost like we factorize the audio signal into these different components and we process them separately. Yeah. But it's amazing like just how well you can do without doing that. Yeah. Can get a long way. Yeah. Well done. Yeah. Thanks. I love

Will Williams 01:02:40

it. Ethics in in speech recognition. What are your thoughts about that? Very wide ranging kind of topic, I guess. But think ASR has not been in the limelight here. I think we've gotten away with not having too many practical difficult decisions to make. You know, we don't have to have a group of people red teaming our models, for example.

Tim Scarfe 01:03:06

Isn't that ironic though, because speech data, you know, from a biometric point of view is very sensitive. Yes. It's arguably way more sensitive than the stuff people are putting up to chat GPT.

Will Williams 01:03:17

Oh, yeah. So if we were doing voice identification and storing that, we'd have big questions to answer. We're not doing that. Yeah. So we've we've somewhat sidestepped that that sort of whole area.

Tim Scarfe 01:03:27

Yeah. And I and I guess the difference is as well that you you throw away the customer data. Yeah. Whereas OpenAI, you know, they seem to have almost re-defined the boundaries of what's acceptable by boiling the frog. So at first, everyone was like, oh my god, you're storing the data. This this is terrible. They didn't do anything about it. And now everyone's just slowly acclimatizing to it. But voice data does seem quite sensitive. And I guess this is why we're seeing the move to the client, you know, the edge. Yeah. You know, where voice data is processed locally and then might not be sent up to the cloud. But we went through all of this with X-ray. We checked and, you know, the providers just throw the data away. Yeah. And that's why it's okay.

Will Williams 01:04:07

Yeah. Yeah. I like running stateless services. So, you know, it's great from a computer science point of view, but, yeah, from an ethical point of view, it's very simple kind of story. 1 of our selling points is that we wanna be like, you know, the trusted secure enterprise provider. And so, yeah, going off and training on people's data is even if even if we have consent, it sends a particular signal. And my question would be, what do we actually need that to progress our systems forward? And historically, we haven't we haven't needed to do it. And so it makes it an easy decision because I really would like customers to not have to worry about this.

Tim Scarfe 01:04:45

Yeah. It's a tough 1 because the reason why Google is so good is for this reason, they actually use the users as a feedback signal and and Bing doesn't work very well. And I used to make the same argument about OpenAI and Anthropic. So OpenAI are the incumbent, and they're doing the mechanical checks, still the customer data update strategy. Anthropic don't do that. They say that even though in their terms of service they can use customer data, they don't, and they haven't. And they're now the best. So maybe it's not that important after all. I don't know.

Will Williams 01:05:14

Yeah. Well, they've clearly made it work. And I think, yeah, we similarly haven't felt too much pain from I mean, already going down our route, so.

Tim Scarfe 01:05:25

Yeah, interesting. Has anything else come up on on that?

Will Williams 01:05:28

Well, we have a question, you know, 1 of the things that we want do is understand every voice, you know, this is our vision, it's our mission, we want to execute against that, we take it seriously And part of that means that we want to recognize underrepresented people, be as inclusive as we can, whilst recognizing that the training distribution is not always gonna cover everyone would love it to. And so, you know, it's 1 of the reasons why we've pushed hard on self supervised learning is if we feel that we get a wider, yeah, wider generalization on that front. And, yeah, we started to measure that somewhat internally. But that that's that's yeah. That's an interesting ethical question. It's like, to what extent are you just gonna optimize purely for revenue? To what extent are you gonna build a product that could be used by a really wide range of people? So there's there's a tightrope to walk there.

Tim Scarfe 01:06:20

Yeah. Know. I wonder what the trade off is. Remember there was that famous gender shades paper where Timnik Gebrew, I think, showed that facial recognition didn't work very well for people of color. And even now with multilingual LLMs, they typically don't work as well, you know, for for the primary language. So the mainstay LLMs that people use, they work really well for English Yeah. And then they don't work, you know, there's this kind of long tail. And there's always the question of, are you in any way kind of making a trade off? So by diversifying and oversampling, you know, low attribute classes, are you kind of reducing your headline performance on on the main class? And is that a trade off that you're thinking about?

Will Williams 01:07:00

Well, I I don't quite have that same mental model. So, you know, if you need more parameters to soak up a wider distribution, go add those parameters. It is more an issue of having that data in the first place really.

Tim Scarfe 01:07:15

Is it so so having the increased ambiguity of like more distributions doesn't affect the accuracy of of the primary 1?

Will Williams 01:07:22

Not not massively. Not massively. There are there are obviously going to be trade offs there. But if you have the if you have the data, you you we can train larger models to to soak up and and learn how to

Tim Scarfe 01:07:35

handle it. And in some cases, like a lot of languages are quite low resource and also like with low frequency classes, there might just not be much data there. So would you oversample or would you like synthetically generate it? We haven't

Will Williams 01:07:48

gone too far into synthetically generating stuff. I think that's something that we could consider more. But yeah, you would have to oversample and learn learn as much as you can from the the paucity of data that you do have.

Tim Scarfe 01:08:01

So why do all of the ASR companies say that their model is the best? Well,

Will Williams 01:08:08

first answer that might be that they truly believe that's the case because they're in their own world of their own training and testing data. So it may well be that on their own data that that is true and really the only way out of that is to get third party evaluation done and and hopefully that will be a bit more representative of a general, a broadest, broader distribution over which to test. I think marketing maybe can take over as well and, you know, do its thing with these numbers. In in this field, in SI and in others, I'd encourage people to look at third party evaluation because the way that you evaluate and the data that you evaluate on makes a dramatic difference. Mhmm. You know, if you if you consider collapsing words that have a hyphen or down down to 1 entity, this can, like, this can produce differences in your word error rate that are equivalent to, like, fairly large architectural gains that get published. So this is like very very significant and as you get get more into the sort of the test set floors, I would call it, these evaluation method type methods actually dominate the final result. So it's not just the data. You want that to be independent third party ideally, but the eval as well can just, like, really swing things. You know, a language like English and ASR is very mature now, and so that that method of of evaluation really will dominate. And so, yeah, you know, if you've got your own use case and you care about certain things and don't care about others, then go build your own sort of metrics and, you know, that that should be the way to go. But, yeah, it is a comp it's not a it's

not a straightforward space. I mean, we've been doing this for 10 years and we've constantly been updating our ability to evaluate this well and I still don't think we're there yet. So, you know, I think about some of our customers, how on earth are they gonna do a really good job of this? I'm not sure.

Tim Scarfe 01:09:59

Yeah. I mean, we used to use a company called 3Play Media, I think, and and they they do a whole bunch of, you know, benchmarks across the ASR providers. What what do you think about those those third parties? And and even even with that, as you were saying, just the the methodology they use is actually problematic for reasons that some of the customers won't understand, right? There's a few

Will Williams 01:10:21

things to think about. On the evaluation metric you may be verbatim or not in ASR. So maybe your test set has things rendered in spoken form or written form. Maybe it's dropped all the filler words. Maybe it hasn't. So maybe it's not cleaned up form or not. And, you know, I think this is 1 of the the the aspects that we hit up against is we produce a verbatim transcript. We want to be more verbatim. We want to have filler words. We want to have hesitations. We we want to give users the option of whether it's in spoken or written form. So we we want to just do things as correctly as possible, but sometimes that that hits us on the on the benchmarks. But if you do a pure

Feature engineering to automated ML

Will Williams 01:11:00

verbatim transcript, know, I

Tim Scarfe 01:11:01

think we back ourselves. When when I read the I think there was like a state of ASR paper from 3Play Media and they were comparing all of the transcription providers and Whisperer was on there, and you know I think Whisperer's pretty good but out of all of the providers it really suffered with hallucinations. So why is Whisperer different? I think there's

Will Williams 01:11:22

a few reasons why. Number 1 it's a supervised model that's been trained on data that contains artifacts that maybe we don't want to train on. The other thing is that it's end to end which which means that you're we're modeling the acoustics in the language in the same kind of space and we sample and sometimes that sampling can go wrong. Mhmm. And so we do have a little less control over what it's gonna produce. So there's a big data problem there but I think there's also an architecture, something that's built in and this is something that's also built into language models by design, by language models. So, yeah, I think there's been some work to mitigate this, but but, yeah, we definitely see a a different kind of picture on hallucinations with with our setup, which we, you know, we're quite happy with. But yeah, I I think that the data has played a big part there.

Tim Scarfe 01:12:10

So I I read the Whisper paper out in great interest when when it came out, and they were quite bullish because they were basically saying, you know, we're a bunch of guys and we train transformers, and we've trained a big transformer on, you know, basically all of the, you know, like the the audio transcription paired data on the Internet. They had to do a whole bunch of work to get transcriptions and to clean them up a little bit, because as you said, a lot of them are are pretty noisy. But they were saying, this is an end to end model, and because we trained it on was it 600000 hours of audio that the first version it does, know, really good out of domain generalization and and you know everything in the world is good. What could possibly go wrong?

Will Williams 01:12:48

Yeah, I had some issues with that. So the OAD generalization wasn't really as good as claimed. So we ran against our test sets, which definitely are not out there anywhere and it wasn't generalizing in the ways that I think they claimed. So I was already very skeptical about this. I think it sort of speaks to a wider issue with supervised learning. You know, I'm very much interested in the intelligence question, I think, as OpenAI are. But in my book, supervised learning is is is not even close to being on that path to to AGI, for example. You have to have that data to start with. If there's a problem with your model, you have to go and get somebody speaking it and the correct transcript. If you want to add a new word, you have to go and do that. And so there's a level of generalization that simply does not exist in supervised models and some level of out of distribution generalization happening, but it's it's not as wide as as is claimed. And and certainly, it's not sample efficient. And so, you know, yes, I'm interested in building like really good products, high quality products, but also I am I'm interested in how can we build smarter machines and you've to think about the sample efficiency question there. And and yeah, the sample efficiency is terrible for Whisper. I mean, this is like, oh, it's great in many many ways, don't let me sort of like knock it too much. But it's not sample efficient. In order to train that English system, it has to use a 100 x more data than if you trained a self supervised unsupervised model and then trained a small supervised system on top of that. So, you know, I thought it was a little bit ironic at the time because OpenAI should be pushing the big AI questions here And actually, this is just a somewhat dumb supervised model that's very, very good at 1 thing, but doesn't possess that kind of like generality and generalization that I think we're shooting for.

Tim Scarfe 01:14:47

Yeah. Just help me understand, sir. I I my understanding is the whisper model is self supervised. Have I

Will Williams 01:14:52

got that wrong? Well, it will train on transcripts, so it needs transcripts. Yeah. So so hang on. How does the whisper model work? So it's an so it's like an encoder encoder sort decoder model. Okay. And so waveform in, encode it, and then you'll be predict then they set it like a multitask kind of decoding process where, you know, you you kind of it outputs a token that says what the kind of language is and then outputs, optionally output timestamps in the word and so on and just do a full kind of decoding process. So it's sort of encoder decoder setup, but like the key for me is there's no self supervised piece in there and that's fine. You can get away with it to some extent with that but you have to sort of make up for that with a large amount of data and for some languages that large amount of data simply doesn't exist And to scale and deal with a long tail of issues, you're not going to get the data to go do that. And so like so it's like it's okay, but it's like it's it's it's it's not got the kind of level of generality that that I would like sample efficiency that lets you do, you know, better on other languages. Yeah. I mean, I've been a huge advocate for building unsupervised models and learning directly from data and then using those models to give you the generalization power. And I believe that approach is scalable, you know, with respect to the intelligence. So you're a

Tim Scarfe 01:16:11

huge advocate of of this kind of unsupervised approach, know, sketch that out for me.

Will Williams 01:16:16

Yeah. So, you know, self supervised learning in general is you want to set up some task where the data itself in some way defines the task for you. We've seen this across, you know, lots of different fields now, you know, kind of infilling, masking, predicting the future, corrupting the data, predicting the the uncorrupted version from the corrupted version and so You know, these are all these are all ways of learning from just directly from data, learning useful representations. I do see ASR as actually sort of the wider AGI question as a representation learning problem primarily. I think if you've the right representation to operate over, you can you can do things very very quickly and so, you know, this was 1 of the most amazing results I've seen in my time the company about 4 or 5 years ago, you know, we started introducing self supervised models and we got to the same loss on our ASR task with a 100 x less data after we inserted a self supervised model that was trained on some prediction task. And this is this is like a wow moment. This is like so so significant. You've just done a 100 x on that on that journey towards a more intelligent system. I think that's like that's really significant. We haven't really gone beyond that since that point. And so I think

we need more of those kind of moments to be honest.

Tim Scarfe 01:17:30

Whisper is is self supervised or it's not? Well, you could think of

Will Williams 01:17:35

it as self supervised I suppose on the language modeling part. So it's decoding to words and, you know, you can do kind of a word level prediction. So it's, you know, in some sense that's in some sense it's self supervised but really the core of the training is is classic supervised, you have audio in, you predict you're trying to produce a particular transcript and so you need the paired data. Mean that's the key. Interesting. So so what's the setup with Whisper then?

Tim Scarfe 01:17:59

So so you you have you have like a whole bunch of pairs of audio and text and then you chunk it up and then you put the audio in and then it does a decode and then based on some loss function you update the weights in the decoder and the encoder. Is that not self supervised? No. I mean self supervised would only have the audio data. So you're saying the difference with your approach is that you have like a completely separate, you know, self supervised process just for the audio, which is not paired with the text?

Will Williams 01:18:30

Correct.

Tim Scarfe 01:18:30

Okay. And on their 1, their process kind of like conditions both of the encoder and the decoder on a single prediction task. Yeah. That's the difference. Okay, that makes sense. Yeah. Because it reminds me a little bit, and Jan Lecun used to advocate for this approach. So there was a paper called Dino, where, you know, he would first train a self supervised model, you know, just on like visual representations, and even like showed you the represent representation space had far more information in it. And then when you go to kind of do a supervised, you know, downstream task, the representation space will kind of specify because it's like it only needs to know what it needs to know for that particular task, so you're throwing away all the other information.

Will Williams 01:19:07

Yep. So you're kind of like keeping around more information in that audio model. Yeah, that's right. That's right. And then, you know, the idea is, the hope is you go train with just like 1 hour data, really tiny supervised dataset that can just pull out the right representations and just maybe like learn that final clustering if you like.

Tim Scarfe 01:19:28

But you're applying a lot more kind of human designed architecture into this process, you know. There are people who say the benefit of deep learning is that you could just go end to end, and it's just magic. But you're saying actually, in many cases, it's better to kind of like decompose the problem into different modalities and subtasks, and and then put the whole thing together later?

Will Williams 01:19:47

Yeah. So the way that I would approach it is we want to do as much unsupervised learning as possible. We want to learn as rich representations of the world as possible and then take a smaller labeled dataset as you can and and then go learn an API for your product. And in our case that will be training, you know, a small supervised ASR system based on a fairly meaty unsupervised model. This gets you a level of generalization that you won't get with with just supervised learning. Or you you can get it but you have to cover it. You have to cover it with a lot of data in supervised case. Yeah. And that's the trade off.

Tim Scarfe 01:20:28

Yeah. Because, you know, there was that Anthropic paper recently that on the the Claude sonnet model, it learned this representation space and you could get it to talk about the Golden Gate Bridge. Mhmm. And and stuff like that. It was really interesting that it seemed to find all of the the categories that that we talk about and we know. Yeah. But don't you have to put some priors in? I mean, it's almost like if it's completely unsupervised, if it's a complete blank slate, how does the model know, you know, because it's like a clustering algorithm in a way, isn't it? It's kind of saying this is really important, so I'm going to keep this around and that's not important. How does the

Infrastructure scaling and PyTorch critique

Tim Scarfe 01:21:00

model know like what to hold on to and what to throw away?

Will Williams 01:21:02

Well this is what gradient descent teaches it. So if you if you set up a test that's sufficiently difficult and has sufficient depth, so as it's learning it can be learning increasingly more interesting things. This is what we have with LLMs. That next open token prediction, you know, it starts quite easy, but then as you've trained your model to just get a tiny improvement in your loss, you're gonna have to you have to learn some world knowledge. I mean, in order to improve it. I mean, this is really good. So these are like tasks that in some sense have infinite depth and and it's like that with audio. You can you can go and set up a prediction task, and it's hard. And in order to improve about that, you have to learn increasingly subtle things about the real world. And because you've trapped those representations, they're then available for you to use when you're training a supervised model on top. Interesting.

Tim Scarfe 01:21:46

So are you talking about like an auto encoder?

Will Williams 01:21:49

Yeah. That that that's an example. I'm big on using time as a supervisory signal. I think time has a lot lot to offer us.

Tim Scarfe 01:21:57

I'm I'm a huge fan of time as well. Yeah. Yeah.

Will Williams 01:22:01

Yeah. I've got a lot of time for time. I think 1 of the big things that I want to advocate for is learning on multiple scales of granularity. And time, having a system that's in a certain state and is sort of faced with the world, it will get a a signal on multiple sort of resolutions. So we'll get like a pixel level kind of supervisory signal. But then things are happening at the kind of the meta level, I guess, our environment that that happening and happen in sequence. And so there's a signal, there's like a very rich signal. There's almost like hierarchical that's coming from the environment that we we is there for us to to us to exploit. And I don't think we've gotten going exploiting that yet

Tim Scarfe 01:22:43

properly. So the the features evolve with respect to time at different resolutions. Yeah. So you could almost kind of like create a bunch of like time differences. Yeah. Turn those into features and use that as part of let's say your auto encoder model?

Will Williams 01:23:00

Yes you could. I think this is almost like the next frontier for me is removing the engineer from constructing the task. So you can you can model at a certain rate, I don't know, a 100 hertz and you can learn interesting features. You can get Hubert style tokens, the the semantic tokens, representations of what's going on. They're like 25 hertz. You can predict over those and actually that's what's what people are using to build speech sensitive systems at the moment. So it's interesting. We we've got these different components that are clocked at different rates. They're all useful to come together because they're learning different things that are important. But the problem is that you have to you have to have an ML engineer, ML researcher go and figure out that these resolutions are important and this is exactly how we should set up the task. And so, you know, that's that's kind of my next frontier for that I see for self supervised learning is discovering these these levels automatically and constructing prediction tasks or having them emerge automatically, and so that we can predict over them and learn really efficiently.

Tim Scarfe 01:23:59

Someone from OpenAI might say, don't do feature engineering, why don't we just do a 3 d convolutional neural network? So we take a spectrogram every 300 milliseconds, and then we just stack spectrograms over time, and we do a 3 d CNN over it, and the system will learn all of these different time dependencies at different points in the spectrogram and so on. The alternative is to impute some domain knowledge into that, so you're actually discretising that 3 d spectrogram, and you know you might have some log curve on there, so as the time horizon goes wider you increase the coarseness, maybe you increase the coarseness at sort of like lower time horizon. So what's the trade off between doing that yourself versus letting the neural network figure out?

Will Williams 01:24:45

Well, if you're doing it yourself, you've probably got some application in mind and you might be able to tune your way to victory there and maybe there are certain applications where that makes sense. I think if we're interested in building increasingly general machines, we don't have time to do that, we don't have probably the knowledge to do that, and so ideally this would this would emerge naturally. Yes.

Tim Scarfe 01:25:06

So it's almost like creating a system that can meta learn its parameterization. Or even meta learn it's kind

Will Williams 01:25:11

of the task on which it needs to it needs to predict. Feature engineering, you know, there was like there's this progression that's happened over the last 10 years, so you know, people used to say, okay, feature engineering that was something we did by hand. But now, okay, we have these stacked auto encoders that can go learn that for us. And then it's okay, can train this like end end to end and we have this like deep learning architecture. And oh, we can automate that. We can have these kind of discrete program search to go look over all the different architectures that can produce a new 1 for us. And so that part of the ML engineer job has also disappeared. And and now kind of we're scaling these architectures. So so this is like a a scaling piece that's kind of been solved. And, you know, I see the bottleneck now as the task itself, is is the big bottleneck. We need to remove the ML engineer from defining the task in the first place. I mean, you have to put in some price, obviously. It has to be some price, but they can be quite weak, I think.

Tim Scarfe 01:26:04

Yeah. I mean, it's just like what Francois Cholet says about developer aware generalization. So, know, the systems are generalized to a task that the developer of the system didn't didn't know about. That's what we need

Will Williams 01:26:15

to get to. That's not much of what's been happening, I think.

Tim Scarfe 01:26:18

So you're saying I have this ridiculously high dimensional space, the space is sparse. Actually we don't need most of the information in there, so we're essentially designing a domain specific compression algorithm that could generalize to any, you know, future downstream tasks that we don't yet know about. Exactly.

Will Williams 01:26:34

I mean, and this is kind of the idea of compression in general is you need to decide what you're going to throw away and forget about, what abstractions you're going to hold on to. Isn't there

Tim Scarfe 01:26:43

like an infinite set of abstractions? So so as you say, it's really easy if you know the downstream tasks, you can design the abstractions. That that that's a trivial problem. And then there's a whole spectrum of, okay, I don't know the downstream tasks. So is there some natural structure, you know, to the compression algorithm that could generalize really well?

Will Williams 01:27:01

I I think there always has to be some application in mind. And, you know, you think about how humans do this, I think we have a lot of contact with our environment, we take actions, we get signal. And so there is some signal at some level here. So it's not completely, you know, devoid of that. But there's also some, you know, redundancy removal that can just be done in general, you know, classic compression almost. And then the rest of it has to be, I think, you know, those abstractions need to be targeted towards a certain application at some point.

Tim Scarfe 01:27:35

I mean, this is a spoke with Schmidt Hoover the other the other week, and he said intelligence is compression. But there's always a thing that, you know, even from a Schrolei point of view, intelligence is about something that happens in the moment. So so in the moment, I create a skill program that generalizes really well for the particular task. So in the moment, you know what the task is, but we're talking about before the moment where you

Will Williams 01:27:55

know what the task is. I I think, yeah, these are the foundational representations that we need to support those that kind of like dynamic on the fly program generation. And I see a big web of those representations that are missing from current systems that we need to go build.

Tim Scarfe 01:28:09

Would you ever envision a system because, you know, right now we're in the paradigm of, you know, frozen models. So we train a big model, we freeze it, and then we expect it to work in a multitude of different situations. And as we've just described, that in a way that that's kind of broken, because, like, we're we're memorizing a distribution, and the system as a whole isn't intelligent because it's not dealing with novelty. So would there be future ASR systems that could kind of like adapt themselves, you know, like maybe test time inference or something like that?

Will Williams 01:28:34

Yeah. Be a big big advocate of that as well. This is what kind of I think a lot of customers believe that AI solutions would do is, you know, adapt and not be frozen and maybe people outside of, you know, the bubble that we're in don't realize that at all. But yeah, this this is like feasible and we should get there at some point. I'm not quite sure of the journey there but, yeah, having systems that can adapt to what you're saying and how you're saying it, the acoustics and become much much more dynamic is clearly beneficial.

Tim Scarfe 01:29:09

Active inference, as Carl Friston calls it, I'm huge fan of this idea, but then from a software engineering point of view it becomes really challenging because it's the ultimate manifestation of what you said before, which is like, you know, doing fine tuning models, letting the customer fine tune the models. But it's also really cool because, you know, in a federated learning point of view, you can actually be feeding back some of these learnings into the Nexus. So you've got this, like, dynamic, agential system. But how would you even test such a thing, and how would you how would you roll such a thing out?

Will Williams 01:29:36

Yeah, I don't know, the thing is goes beyond beyond some of the things I've been thinking about but federated learning is really interesting. Yeah, so maybe you have some core set of metrics that you're you're running against your your central model to make sure that your the updates from all of your local clients are not sort of disturbing the system performance. But as we were saying, if there's a level of qualitative eval that's required once these systems are so mature, that becomes a challenge.

Tim Scarfe 01:30:08

I'm really interested in federated learning as well, especially because now increasingly we're getting phones and computers that have, you know, incredibly powerful GPUs on and we're getting very good performance with quite small models And even Apple did a thing where, you know, they have a big model in the cloud and they have a small model on your phone and and there's some kind of handoff between them. And figuring out the architecture and topology of that I think is quite an exciting future. I mean, think there'll

Will Williams 01:30:31

be some level of distribution as we build more and more capable models. There'll be more interesting updates coming from from different directions to help you to help us improve and move forward. What's the, you know, the origin story of the company

Tim Scarfe 01:30:45

and and what's your background?

Will Williams 01:30:46

Yeah. So my background's in computer science. I wanted to get into machine learning around 2013 when there was beginnings of talk around deep learning and that became a phrase and I was very curious about what was going on there, so I wanted to get into that field. And so, yeah, I was due to do an MPhil at Cambridge and I posted on Google plus of all places, is there any anyone sort of offering any internships in Cambridge? And Tony Robinson, who's our founder, is, he responded. I was like, great. Let's go. Let's just go build some neural nets. And his name had just come up on a slide on 1 of Jeff Hinton's Coursera courses on on neural nets. So I knew the name. So okay, this looks great. And then, yeah, we went straight into doing some work on scaling recurrent neural network language models, which, yeah, was having a really big impact on the speech recognition. And so, yeah, in those early days we were doing some really exciting research, but it was translating straight into what was looking like quite an exciting product. And I really enjoyed that balance. Never did that master's and have just kind of stayed here since then and have helped scale our tech and the company as well. Tony Robinson is an absolute legend. He was 1

Tim Scarfe 01:31:54

of the early pioneers of, you know, using deep learning in speech recognition. And of course, 1 of the founders of Speechmatics. And completely unrelated to this, he actually gave me some, you know, some advice actually. So when we were building X-ray, I think I met him on Facebook, and he reached out and he said, you know, I'll spend an hour with you and I'll give you lots of advice about what you're doing and so on. Because he said that 20 years ago he tried to do something similar and the technology wasn't quite there yet, but obviously now now it is. But some, yeah, he's he's a living legend. Yeah, very much so. Tell me more about it. Mean, how did you at at the very early days, know, when when when he was building Speechmatics, I mean, what was it like?

Will Williams 01:32:34

Originally a distributed company, so I was working remotely and I didn't enjoy that so much. So I moved out to Cambridge and basically started working in Tony's attic with a couple of computers under the desk, which was a lot of fun. And, yeah, I remember we were writing some of our own ML frameworks at the time and we were running some long runs on it was a language model build, and I think it'd been running for 5 days. I hadn't implemented the checkpointing code in c plus plus which is a big error, a big sort of noob mistake. And 1 day, Tony comes up with a tray, tray of coffees, and and then I don't know what no. He plugs in the kettle. He plugs in the kettle and the machine just goes down and and I just, yeah, head in hands, lose the experiment, have to restart again. So it was all quite chaotic and sort of building our own server farms and things like this. So yeah, it felt like the real kind of startup experience in Cambridge which was a lot of fun. We modeled 3, we built some good models and that's kind of how Speechmatics came about. It wasn't originally Speechmatics, was more of just a research outfit, but it naturally evolved into that pretty quickly.

Tim Scarfe 01:33:47

Yeah. And you said you're working in C plus plus I mean obviously now we are spoiled with PyTorch and all of these high level abstract frameworks for building deep learning models and we could do checkpoints and we got weights and biases and we got all this kind of stuff. What was it like back then?

Will Williams 01:34:00

It's a lot more difficult to get performance out of the GPUs. So to be honest, when I started here, we started using CUDA and that felt that felt like a revolution from using OpenCL which I was using previously. So I thought I had it easy but the flexibility that you have with this modern frameworks is amazing. I mean it really has been 1 of the big drivers behind these sort of exponential gains and progress that we've seen in the field. I think a lot of maybe newcomers will take that for granted, but it wasn't always there. There were good frameworks, so there was TorchLure and Ciano and others that were very good and very clever, but you still had to be careful if you wanted to really push the GPUs hard. So, know, we ended up writing some of our own custom CUDA C code and we, yeah, optimized that by hand. And that was really what was needed to get the best systems at the time. Whereas you can get you can get close to that now with with a lot of stuff that's off the shelf, which is which is amazing. I know there's lot of work still on compilers to to make that story better, but yeah, it's very very nice now.

Future of conversational AI and Ursa 2 ●

Tim Scarfe 01:35:00

Yeah, because there's an interesting journey from at the beginning you're just building a single model on a single GPU and were you ahead of the curve on doing things like, you know, distributed model training and stuff like that?

Will Williams 01:35:11

Yeah. Yeah. We're doing multi GPU training. Yeah. Yeah. We I mean, people talk about like fusing operators at the moment. That wasn't a thing because they were already pre fused. I mean, you you didn't defuse them or anything like that. You just wrote the kernels. And so, yeah, some of the things that was I'm seeing us doing today, was like, what why why why exactly are we back to I mean, because we have lots of flexibility with things like PyTorch, which is great. But in terms of writing a really efficient application, you could just go straight and do it, particularly memory management as well. I'm not a massive fan of the sort of dynamic memory management and the the caching allocator that you have in PyTorch. For a lot of the applications we have, you can just if you know what you're doing, just go define your memory upfront, do a little bit of memory management yourself and be a lot more efficient. So in some ways, I've seen this kind of regress a little bit or at least it feels like it. But but yeah, I mean the flexibility is just amazing these days, so I can't complain about that.

Tim Scarfe 01:36:04

Yeah. It must be interesting because obviously from from your perspective, you're hiring a bunch of, you know, the best experts in the world at this. And it's actually good to have that flexibility because you have very bespoke requirements and you want to do it exactly as you want to do it. And even something like PyTorch, you know, you might still grumble about because it doesn't quite do what you want it to do. Yeah. No, exactly.

Will Williams 01:36:22

We can make it do what we needed to do, which is the nice thing. So it can be extended in ways that are interesting.

Tim Scarfe 01:36:28

So at the very beginning of your company history, you just had a handful of people and and, you know, you you kind of had had access to a beefy machine and so on. What's it like now? How many people do you have working here and what was the journey from there to now? So the company is now

Will Williams 01:36:40

120 people, engineering is about 55, machine learning teams about half of that again at 22. So that's a big, big change for us. We were even bigger at 1 point. So we've gone through, yeah, a bit of a roller coaster kind of journey as a company and we've, you know, felt quite a lot of those growing pains. I think they're common to companies of our size And I guess on the compute side, we've seen our kind of compute cluster kind of slowly slowly build. We've kept lots of the old cards that we've we've had and, yeah, as long as they don't fry, we'll we'll keep training things on them, which is great. There's also questions around culture, think, around how you maintain some of that dynamism and energy from the early days where, you know, people can just roll up their sleeves and get almost anything done. You have that feeling whereas, yeah, that feeling of coordinating and communicate needing communication to be strong, you start to become very aware that that's that's that's the

Tim Scarfe 01:37:40

bottleneck now. How's that panning out? You know, like, OpenAI as an example, they at the beginning, they wanted to save the world, and then they started making lots of money, you know, selling to consumers and so on. Is it a similar thing here that, you know, when the company grows, you start to become focused on, you know, building out products and being quite commercially focused, and then, you know, some of the the original kind of like engineering pluckiness erodes over time. How have

Will Williams 01:38:05

you countered that? I've always wanted to make sure there's a sort of core components looking at more fundamental questions and that's more or less always been the case and that will drive projects that are always on sort of slightly longer term. So yeah, there will be a product roadmap that's like, I don't know, about 1 year long, then you've got the eye on the sort of 18 18 months, 2 year, maybe 3 even 3 year kind of projects that are gonna deliver a bit of a step change or a paradigm change and you know as far as I've been able I've always pushed for us to maintain that which has always been, it's been a bit of a struggle and particularly when there are projects that are quite difficult to describe to the rest of the business. But we've always managed to to keep some kind of allocated effort there which which I think has served us really well actually, particularly in a field that's been changing a lot. I mean if you if we just stuck around with some of the early approaches, we'd be we'd be dead in the water. So that mindset was I think it's not just required in general but particularly in machine learning, which has just been such a fast moving field in the last 10 years.

Tim Scarfe 01:39:06

I know I was going to ask about that because when you look at machine learning research, it feels like it's moving at 1000000 miles an hour. And to build a company on this stuff, you need to sort of like take it to an insane level of engineering maturity. So there's always the trade off between do I allocate 20% of people you know, just trying out all of the latest models or do I try and focus on polishing what I've already got?

Will Williams 01:39:27

So there has to always be some level of incremental work. But despite there being like a fire hose of, you know, activity on archive and, you know, people publishing an exponential number of papers in Europe and and the rest, there is just a common core of important work to focus on and that feels tractable actually. So I think as long as you do allocate sufficient percentage to staying on top of that, you can you can walk that tightrope.

Tim Scarfe 01:39:53

It's very possible. So you're based in Cambridge, which is really good for talent because you can kind of siphon off all of the bright young students from from Cambridge and so on, but you've said to me before that, you know, talent density is really important. So it's actually better to have fewer, smarter people than, you know, than than to kind of have too many people. So what's your hiring process look like? What what do you what do you look for? And how is that happening at Speechmatics?

Will Williams 01:40:16

We've had real success with hiring interns straight out of university who are really hungry, really capable and training them up and seeing them do great things and we've done this multiple times over. Investing in a few like that is very rewarding but it seems to be very good for for progress and Mhmm. And retaining institutional knowledge as well. So, yeah, we will hire for that kind of general capability and flexibility and and maybe also generalist mindset. So, you know, I have this mental model as to 2 types of people. There's people who have a very incremental mindset and people who are kind of generalist and scrappy. And people really do neatly, in my mind, fall into these 2 categories. And so we'll push hard to find people who generalists and who are scrappy and hungry and can make things happen. And that has served us very, very well. What does generalist mean to you?

Tim Scarfe 01:41:11

Does it mean someone who's end to end in terms of they can build UIs and apps and they know the research or is it someone who can kind of like understand the business requirements and the technology at the same time?

Will Williams 01:41:22

I think it's more on the computer science end of things. So so people who can, yeah, implement anything across the stack really. And as far as machine learning goes, we've not really hired for machine learning experience sorry, speech recognition experience, which again I think has served us well. I think you can pick up speech recognition if you need to very, very quickly and there's no issue there. And as it's turned out over the last 10 years, general engineering ability has become more and more important. Our systems have become more more homogenous, people spending their time scaling transformers, doing distributed compute. So, yeah, that's that's been another sort of driving force that's sort of worked well alongside I think how we've approached hiring.

Tim Scarfe 01:42:11

Yeah, because it's quite interesting and as you say that it's almost becoming about software engineering skills more than anything else, which is very important, but at some point if you lean into that, you know, there might be some other interesting things that you might be, you know, potentially missing out on. An instructive question I think is if an intern, you know, sent you an email and said, well, I want to work at Speechmatics, you know, what are

Will Williams 01:42:32

the biggest signals for you? An interesting set of projects. Yeah. Maybe I what I've just said implies no experience, but I think having having evidence that you've rolled your sleeves up on some real life projects, got things working, and have, you know, got to done done on those projects is is really impressive, probably more so than anything else. Mhmm. So that's 1 of the big indicators that I look out for.

Tim Scarfe 01:42:57

Yeah. I completely agree. I mean, when I'm hiring engineers, that kind of self directed autonomy and ambition is really important. So look at their GitHub, how how active are they, how self directed are they. Now, just on the software engineering fundamentals, know that we share many of the same beliefs on this. So Will, a man after my own heart, has instigated a mono repo at Speechmatics. Tell me

Will Williams 01:43:20

about that. Yeah. So this was this was a fairly big project in getting us over to a mono repo. I think things have just evolved quite naturally previously and and so we've got quite a complicated build process involving Debian packages and lots of individual repos and the way that things are assembled was becoming quite painful. Mhmm. And I have this principle if things move together they should live together. Everything was moving at the same pace and we essentially haven't had a mono build but it it was it was it was distributed. So so each of those components should have moved forward together Mhmm. And they could have been compiled together but we weren't getting any of any of the benefits of them moving forward independently. So it made sense to bring things together and we did that and it's just it has really really simplified our build. We can get to like a final act like much much quicker. Although we've had to up our maturity on the DevOps side of things, I think it's part the trade off that you go for with the with the mono repo. But that's that's been good for us and so we're just like going

down that kind of that journey and and making the most of that now.

Tim Scarfe 01:44:27

Yeah. Mean it's a bit like your comment before that sometimes rather than focusing on, you know, I want to do something now, you need to take a slightly longer term view. Yeah. And Google did this famously, but many people say that it's created a bureaucracy and it's it takes months to get your code checked in and so on, because, you know, they've got this wonderful, consistent code base, but, you know, you you almost pay for it a little bit because you you have to get things approved, and you have to speak to product owners, and code owners and stuff like that. So it's always a bit of a bit of a trade off. How have you, you know, approached that operating model? So I don't

Will Williams 01:45:00

think we're operating at the scale where we've hit some of those kind of those kind of issues. So you know, if you refactor a component and it's affecting like 10 downstream services, you've got to go and then, you know, deal with that refactor yourself. We're obviously not in that that situation but we have we spent time in our CI pipelines to speed things up. We've had to back it with more compute. We're testing more frequently, so integration happens earlier when you do mono repos. Mhmm. Yeah. We've had to think a bit more seriously about that. But, yeah, definitely seeing it pay off.

Tim Scarfe 01:45:29

Will, this has been amazing. Thank you so much for coming on the show. I mean, do you have any, you know, take home messages for the audience?

Will Williams 01:45:37

Yeah. Well, first, thanks so much for having me. And it's been a really fun conversation. Really enjoyed it. I guess my 1 takeaway would be like, beware supervised learning. Mhmm. We want to shoot for bigger, ambitious goals. Unsupervised learning's gonna have to pull a heck of a lot more weight. And, yeah, I'd encourage listeners to work on it and see how far we can push that paradigm.

Tim Scarfe 01:46:01

Amazing. And where can people find out

Will Williams 01:46:02

more about you? You can find me on speechmatics.com or on LinkedIn. We'd love to connect with a whole bunch of people. It'd be great. Amazing.

Tim Scarfe 01:46:11

Will, thank you so much.

Will Williams 01:46:12

Thank you.

References

[1] **Speechmatics PDF shownotes**
https://www.dropbox.com/scl/fi/d94b1jcgph9o8au8shdym/Speechmatics.pdf?rlkey=bi55wvktzomzx0y5sic6jz99y&st=6qwofv8t&dl=0
paper

[2] **Speechmatics**
https://www.speechmatics.com/
company

[3] **Will Williams**
https://x.com/wjwwilliams
person

[4] **GFlowNets**
https://arxiv.org/abs/2106.04399
paper

[5] **Ursa 2 model**
https://www.speechmatics.com/company/articles-and-news/ursa-2-elevating-speech-re
paper