

APARNA CHENNA PRAGADA

LENNY'S PODCAST

BILINGUAL TRANSCRIPT

ORIGINAL BY

Lenny Rachitsky

@lennysan • x.com/lennysan

ANALYSIS BY

@Penny777 • x.com/penny777

Aparna Chennapragada - 双语对照

This is the complete bilingual transcript for Lenny's Podcast featuring Aparna Chennapragada (CPO at Microsoft).

[00:00:00] Aparna Chennapragada

English:

I have a cheesy Chrome extension. Literally whenever I open a new tab, it just says, how can you use AI to do what you're going to do right now?

中文翻译:

我装了一个挺“土”的 Chrome 插件。字面上意思上，每当我打开一个新标签页，它就会显示：你现在要做的事，怎么用 AI 来完成？

[00:00:06] Lenny Rachitsky

English:

How do you see the future of product development being different?

中文翻译:

你认为未来的产品开发会有什么不同？

[00:00:09] Aparna Chennapragada

English:

If you're not prototyping and building to see what you want to build, I think you're doing it wrong. It becomes even more important to have that territorial and taste-making at the heart of it because, otherwise, you just have a Frankenstein product.

中文翻译:

如果你不通过原型设计和实际构建来观察你想做的东西，我认为你的方法就错了。拥有核心的把控力和品味变得更加重要，否则，你最终只会做出一个像“科学怪人”一样东拼西凑的产品。

[00:00:23] Lenny Rachitsky

English:

There's this acronym that you taught me, NLX. What is that?

中文翻译:

你教过我一个缩写词，NLX。那是什么意思？

[00:00:26] Aparna Chennapragada

English:

Natural language interface. NLX is the new UX. Often I hear a product builders say, "Oh, yeah. With AI, the model eats the products." That doesn't mean it's not designed. You and I are having a conversation. It's a podcast. I'll have another conversation at Microsoft and that's a meeting. Conversations also have grammars. They have structures. They have UI elements. They're invisible. What are the new principles, new constructs in natural language as an interface?

中文翻译:

自然语言界面（Natural Language Interface）。NLX 就是新的 UX（用户体验）。我经常听到产品开发者说：“哦，有了 AI，模型就把产品‘吃掉’了（即模型取代了传统产品逻辑）。”但这并不意味着它不需要设计。你和我正在交谈，这是一个播客；我在微软会有另一种交谈，那是会议。对话也有“语法”，有结构，有 UI 元素，只不过它们是隐形的。那么，作为界面的自然语言，它的新原则和新结构是什么呢？

[00:00:52] Lenny Rachitsky

English:

I just saw that Cursor hit 300 million ARR in two years. Interestingly, you guys were very well positioned to do really well in this AI coding tool space. You guys had Copilot, the first tool in the world at this stuff. So ahead of everyone, what happened?

中文翻译:

我刚看到 Cursor 在两年内达到了 3 亿美金的 ARR（年度经常性收入）。有趣的是，你们（微软/GitHub）在 AI 编程工具领域本该占据极佳的优势。你们有 Copilot，这是世界上第一个此类工具。领先所有人这么多，结果发生了什么？

[00:01:06] Aparna Chennapragada

English:

I would say...

中文翻译:

我想说的是……

[00:01:08] Lenny Rachitsky

English:

Today my guest is Aparna Chennapragada. Aparna is chief product officer at Microsoft where she oversees AI product strategy for their productivity tools and their work on agents. Previously, she was chief product officer at Robinhood, vice president at Google, where she worked on Google lens, search, shopping, augmented reality, AI assistant, and a lot more. She was also a long-time engineering leader at Akamai, and on the board of eBay and Capital One.

(00:01:32):

In our conversation, we chat about how working in B2B is like being Jean-Claude Van Damme doing the splits across two moving trucks, how she's operationalizing her team living in the future so that they're building towards where things are going, why people still need to learn to code, why the PM role isn't going anywhere, why NLX is the new UX, and so much more. If you enjoy this podcast, don't forget to subscribe and follow it in your favorite podcasting app or YouTube. Also, if you become an annual subscriber of my newsletter, you get a year free of a bunch of products including Linear, Superhuman, Notion, Perplexity, and Granola. Check it out at lennysnewsletter.com and click bundle. With that, I bring you Aparna Chennapragada.

中文翻译:

今天的嘉宾是 Aparna Chennapragada。Aparna 是微软的首席产品官 (CPO)，负责监督生产力工具的 AI 产品策略以及智能体 (Agents) 方面的开发。此前，她曾担任 Robinhood 的首席产品官，谷歌的副总裁，负责过 Google Lens、搜索、购物、增强现实、AI 助手等众多项目。她还曾长期在 Akamai 担任工程负责人，并且是 eBay 和 Capital One 的董事会成员。

(00:01:32):

在我们的对话中，我们聊到了在 B2B 领域工作就像让尚格·云顿 (Jean-Claude Van Damme) 在两辆行驶的卡车之间劈叉；她如何让团队在运作中“生活在未来”，从而朝着事物发展的方向构建产品；为什么人们仍然需要学习编程；为什么 PM (产品经理) 角色不会消失；为什么 NLX 是新的 UX，以及更多内容。如果你喜欢这个播客，别忘了在常用的播客应用或 YouTube 上订阅和关注。此外，如果你成为我时事通讯的年度订阅者，你可以免费获得一年的一系列产品，包括 Linear、Superhuman、Notion、Perplexity 和 Granola。请访问 lennysnewsletter.com 并点击 bundle 查看。下面，让我们欢迎 Aparna Chennapragada。

[00:02:11] Lenny Rachitsky (Ad Segment)

English:

This episode is brought to you by Eppo. Eppo is a next-generation AB testing and feature management platform built by alums of Airbnb and Snowflake for modern growth teams. Companies like Twitch, Miro, ClickUp, and DraftKings rely on Eppo to power their experiments... (00:03:28)

中文翻译:

本集节目由 Eppo 赞助。Eppo 是由 Airbnb 和 Snowflake 的校友为现代增长团队打造的下一代 AB 测试和功能管理平台。像 Twitch、Miro、ClickUp 和 DraftKings 这样的公司都依赖 Eppo 来支持他们的实验…… (00:03:28)

[00:03:28] Lenny Rachitsky (Ad Segment)

English:

This episode is brought to you by Pragmatic Institute, the trusted leader in product expertise. Pragmatic Institute helps product professionals turn ideas into impact through proven courses, workshops, and certifications designed for real-world success... (00:04:35) Aparna, thank you so much for being here and welcome to the podcast.

中文翻译:

本集节目由 Pragmatic Institute 赞助，它是产品专业知识领域值得信赖的领导者。Pragmatic Institute 通过经过验证的课程、研讨会和认证，帮助产品专业人士将想法转化为影响力…… (00:04:35) Aparna，非常感谢你能来到这里，欢迎来到本播客。

[00:04:35] Aparna Chennapragada

English:

Thank you, Lenny. Thanks for having me.

中文翻译:

谢谢你，Lenny。感谢邀请我。

[00:04:37] Lenny Rachitsky

English:

When I asked a lot of people that work with you, what I should ask you about and what I should know about you, something that came up again and again, it's something that I think most people don't know about you, which is that you're big into stand-up comedy, and you take it semi-seriously. Just how serious are you about this? How much of your life is this and most importantly, how does this help you build better products?

中文翻译:

当我问很多和你共事的人我该问你什么、该了解你什么时，有一个话题反复出现，我想大多数人可能都不知道：你非常热爱脱口秀（单口喜剧），而且你是半认真地在做这件事。你到底有多认真？这占据了你生活的多少比例？最重要的是，这如何帮助你构建更好的产品？

[00:04:59] Aparna Chennapragada

English:

It's hard to say I'm serious about a funny business, but I do watch and do stand-up comedy. I do open mics. I've done a few shows.

中文翻译:

很难说我对一个“搞笑事业”很认真，但我确实经常看也经常表演脱口秀。我会去参加开放麦（Open Mics），也演过几场正式的秀。

[00:05:09] Lenny Rachitsky

English:

Wow.

中文翻译:

哇。

[00:05:09] Aparna Chennapragada

English:

I have one set brewing that is around AI, unsurprisingly AI and tech and Silicon Valley. It's really interesting for me. This was an accidental discovery. I had always been an SNL fan and Discovery fan, but I went to an open mic because my son sings, and he went to the open mic for singing and he is like, "Mom,

you should go do this." And I was like, "Oh, let me go give it a try," and I found that I enjoyed it and was good at it. To your question though, about building better products, I'd say both have PMF, I mean, product market fit, punchline market fit.

(00:05:49):

Actually, there are a couple of things that I do find really powerful and useful because in open mics or even when you're testing these things, it's a very tight cycle of iteration, and you get live... Open mics are the real live experiments. You put something out there, you get very clear micro feedback from users, and then you get tough feedback sometimes. And I think as product builders, that's actually one of the great skills to have, which is you sometimes launch stuff that have a fantastic vision, but the first version is not quite there. And Reid Hoffman says this, "Hey, if you don't launch the first version and are not embedded, you're doing it too slow." Just that gap in closing that, it's good resilience.

中文翻译:

我正在准备一套关于 AI 的段子，不出所料，是关于 AI、科技和硅谷的。这对我来说很有趣，是一个偶然的发现。我一直是《周六夜现场》(SNL) 和《探索频道》的粉丝。我去参加开放麦是因为我儿子唱歌，他去开放麦唱歌时说：“妈，你也该去试试。”我想：“好吧，去试试看。”结果发现我很享受，而且还挺擅长。至于你问的如何帮助构建更好的产品，我会说两者都有 PMF——我指的是产品市场契合度 (Product Market Fit) 和笑点市场契合度 (Punchline Market Fit)。

(00:05:49):

实际上，我发现有几点非常强大且有用。因为在开放麦或者测试段子时，迭代周期非常紧凑，你会得到实时的……开放麦就是真实的现场实验。你抛出一个梗，会从观众那里得到非常清晰的微观反馈，有时甚至是严厉的反馈。我认为作为产品构建者，这实际上是一项伟大的技能：你有时发布的东西愿景很棒，但第一个版本并不完美。Reid Hoffman 说过：“如果你发布第一版时没感到尴尬，说明你动作太慢了。”在弥合这个差距的过程中，这种韧性非常重要。

[00:06:30] Lenny Rachitsky

English:

Yeah. I never saw these corollaries between these two things. I didn't realize you actually did shows, and you're working on a set. I wasn't going to ask you for a joke, but if you're working on a whole thing about AI, is there something that you can share from that set?

中文翻译:

是的，我以前从未发现这两者之间的关联。我没意识到你真的在演出，还在准备整套段子。我本来没打算让你讲个笑话，但既然你在准备一整套关于 AI 的内容，能不能分享一点其中的段子？

[00:06:43] Aparna Chennapragada

English:

One joke I'd maybe share is people think about these AI chat products as women because you don't know what's going on. It's a black box, and you don't know what they're thinking. There's an entire set around that, but obviously on the flip side too, that they're probably more like men in the sense that they hallucinate a lot. They kind of are not yet reliable.

中文翻译:

我可以分享一个：人们觉得这些AI聊天产品像女性，因为你不知道发生了什么，它们就像个黑匣子，你不知道它们在想什么。有一整套段子是关于这个的。但显然，从另一方面看，它们可能更像男性，因为它们经常产生“幻觉”(Hallucinate，指AI一本正经胡说八道)，而且目前还不怎么靠谱。

[00:07:12] Lenny Rachitsky

English:

I'm afraid to laugh with this a little bit. This is great.

中文翻译:

我都有点不敢笑了。这太棒了。

[00:07:16] Aparna Chennapragada

English:

And even when they don't know the answer, they make up stuff. They're very confident.

中文翻译:

而且即使它们不知道答案，也会瞎编。它们非常自信。

[00:07:21] Lenny Rachitsky

English:

This is good. Where are we going to be seeing the show by the way?

中文翻译:

讲得好。顺便问一下，我们什么时候能看到这场演出？

[00:07:22] Aparna Chennapragada

English:

TBD.

中文翻译:

待定。

[00:07:24] Lenny Rachitsky

English:

Okay. This is great. Okay, let's get serious again. So you worked at most of your career at a lot of consumer internet companies. You worked at Google, Robinhood, you're on the board of eBay, you're on the board of Capital One. Now, you're at Microsoft. I'm curious just what is most different about working at a company like Microsoft and building product at a company like Microsoft?

中文翻译:

好，这很棒。我们回到严肃话题。你职业生涯的大部分时间都在消费级互联网公司工作。你在谷歌、Robinhood 工作过，还是 eBay 和 Capital One 的董事。现在你在微软。我很好奇，在微软这样的公司工作，以及在微软构建产品，最大的不同是什么？

[00:07:46] Aparna Chennapragada

English:

I think intellectually I knew that, hey, enterprise, particularly the area that I look at most at Microsoft is focused on enterprise and productivity and transforming companies through AI. And to me, I think two things really strike as very different. One, in fact, I just posted about this the other day saying, in consumer, you're kind of like, "Oh, we have a playbook for make the product work or make the feature work and make it delightful," but I think in the enterprise, you almost have... Every time you think you have one use case, you have really two, which is how do you make sure that the feature works well and there's governance of the feature.

(00:08:25):

If you think about even something as simple as sharing a link to a document, you want it to be easy, frictionless, but at the same time, you want that to be secure and safe and being able to have auditability and all of those things. And often, I find that when you go from a consumer enterprise, you fall into a trap of either disregarding that and say, "Oh, we'll just focus on one side of the house," or overly crippling the user experience side and leaning on the other side. So I think there's an art and science and nuance and playbook there too, so that's one big learning for me. The other learning, and especially in the AI era for me has been about this... I think there's a famous trailer from the 2000s on Van Damme on these two buses [inaudible 00:09:13] splits.

中文翻译:

我想在理智上我知道，企业级市场（Enterprise），特别是我在微软关注的领域，是专注于企业、生产力以及通过 AI 转型公司。对我来说，有两点非常不同。第一点，实际上我前几天刚发过相关的帖子：在消费级产品中，你有一套让产品或功能运行良好且令人愉悦的“剧本”；但在企业级产品中，每当你以为只有一个用例时，实际上有两个——即如何确保功能好用，以及如何对该功能进行治理（Governance）。

(00:08:25):

即使是像分享文档链接这样简单的事情，你希望它简单、无摩擦，但同时你又希望它是安全可靠的，具备可审计性等等。我发现，当你从消费级转向企业级时，经常会掉进陷阱：要么忽视治理，说“我们只关注用户端”；要么过度削弱用户体验，完全倒向治理端。所以，这其中也有一门艺术、科学、细微差别和剧本，这是我的一个重大收获。另一个收获，特别是在 AI 时代，是关于……我想 2000 年代有一个著名的预告片，尚格·云顿在两辆行驶的巴士之间做劈叉。

[00:09:14] Lenny Rachitsky

English:

Like doing the splits.

中文翻译:

就像做劈叉那样。

[00:09:14] Aparna Chennapragada

English:

Yeah, doing the splits, exactly. I feel like a lot of the companies, including the tech companies, but certainly the enterprises that I talk to are in these two modes where one hand, this is the most compressed tech cycle that we've ever experienced. It's in the order of weeks and months versus years and decades. If you think about mobile and cloud and internet, and there's just so much happening, the intelligence overhang. On the other hand, there's also humans and habits that... Productivity habits change. It's hard to change and change management through the company is also hard. You don't want to be rash on that. So it's like the future is unevenly distributed but even within the companies.

中文翻译:

是的，就是劈叉。我觉得很多公司，包括科技公司，当然还有我接触的企业，都处于两种模式中：一方面，这是我们经历过的最压缩的科技周期，是以周和月为单位，而不是年或十年。想想移动端、云和互联网，现在发生了太多事情，智能爆发。另一方面，人类和习惯……生产力习惯很难改变，公司的变革管理（Change Management）也很难。你不想在这方面操之过急。所以，未来是不均匀分布的，甚至在公司内部也是如此。

[00:10:10] Lenny Rachitsky

English:

On the second bucket of the bus that Van Damme's riding on of governance and adoption and changing behavior and stuff, is there something you've learned about how to get past that, help that along more?

中文翻译:

关于尚格·云顿踩着的第二辆巴士——治理、采用和改变行为等方面，你有没有学到什么方法来克服这些困难，或者更好地推动它？

[00:10:10] Aparna Chennapragada

English:

The thing not to do is hold back folks who are early adopters. I think that's the other one learning. In fact, I think that's one of the reasons why recently... I've been working with folks to say, "Can we have both," which is the longer-term change management, being able to do it in a trusted way, at the same time do this program we are calling Frontier program and roll out cutting edge experimental features. We just built this world's first deep research agent made for work, post-trained for work. And of course, it has all sorts of edges, rough edges, but if there are only adopters in an enterprise or outside, how can we put that in the hands of those folks without insisting that all of the company be completely developing different muscles?

中文翻译:

绝对不能做的事就是阻碍那些早期采用者（Early Adopters）。这是我的另一个教训。事实上，这也是为什么最近……我一直在和同事们探讨“能否两者兼得”：既有长期的、以可信方式进行的变革管理，同时又推行一个我们称之为“Frontier（前沿）”的项目，推出尖端的实验性功能。我们刚刚构建了世界上第一个专为工作设计的深度研究智能体（Deep Research Agent），并针对工作场景进行了后训练。当然，它还有各种各样的棱角和不成熟的地方，但如果企业内部或外部有早期采用者，我们如何能把这些东西交到他们手中，而不必强求整个公司都立即培养出完全不同的能力？

[00:11:14] Lenny Rachitsky

English:

This program Frontier you're talking about, I wanted to spend a little time on it. So what is the idea? The idea here is people are working in this futuristic environment. How does that actually work?

中文翻译:

你提到的这个 Frontier 项目，我想多花点时间聊聊。它的核心理念是什么？是让人们在一种未来主义的环境中工作吗？具体是怎么运作的？

[00:11:14] Aparna Chennapragada

English:

Yeah, I think the idea is exactly this, which is I want to kind of institutionalize and operationalize my personal model of living one year in the future and say, "What does this..." Imagine a company or a setup like Frontier Consulting Group or Frontier Inc., right? And if you did live in that environment where you had all the AI tools and really advanced deep research intelligence on tap, what are the kinds of questions you'd be asking? What's the kind of work you'd be doing? How would you change how you're going about your work day? So that's the premise and you'd say, "Hey, how does it change an individual?" But also down the lane, we want to think about what does a Frontier team look like. We talk a lot about Frontier labs and models. I think models layer is amazing and obviously that's what empowers all these product building to happen, but I want to push us to think about what does a Frontier product look like? And more importantly, how does a Frontier way of working like? What does a team with three people and tons of compute and AI tools look like?

中文翻译:

是的，理念正是如此。我想把我个人“生活在未来一年”的模式制度化和运营化。想象一个像“Frontier 咨询集团”或“Frontier 公司”这样的机构，对吧？如果你生活在那个环境里，随手就能使用所有的 AI 工具和极其先进的深度研究智能，你会问什么样的问题？你会做哪种工作？你会如何改变你度过工作日的方式？这就是前提。你会问：“这如何改变个人？”但随后，我们还想思考“Frontier 团队”是什么样的。我们经常谈论 Frontier 实验室和模型，模型层很棒，显然是它赋能了所有这些产品的构建，但我更想推动大家思考：Frontier 产品长什么样？更重要的是，Frontier 的工作方式是怎样的？一个只有三个人，但拥有海量算力和 AI 工具的团队是什么样的？

[00:12:20] Lenny Rachitsky

English:

So how exactly does this work? There's a team within Microsoft that's like your job is to use all of our latest tools and build product using that. Does that work?

中文翻译:

那具体是怎么运作的？微软内部是不是有一个团队，他们的工作就是使用所有最新的工具并以此来构建产品？是这样吗？

[00:12:29] Aparna Chennapragada

English:

That is the setup. We are just a few weeks into that setup, but meanwhile what we have done is we've actually set up a fake company and said, "Hey, if you are somebody who wants to come play with some of

the cutting edge science projects and deep research agents and agents at work, come party here."

中文翻译:

确实是这样设置的。我们才运行了几周。与此同时，我们实际上建立了一家“虚拟公司”，并说：“嘿，如果你想来玩玩这些尖端的科学项目、深度研究智能体和工作智能体，欢迎来这里加入派对。”

[00:12:51] Lenny Rachitsky

English:

Wow. And it's only a few weeks in. Okay, so TBD how it all goes.

中文翻译:

哇。才开始几周。好吧，那结果如何还有待观察。

[00:12:54] Aparna Chennapragada

English:

Yeah. And again, these are micro... Let's see. The meta point here also is that in the traditional way, we've kind of always thought about across the companies, across industries, really thinking about roll-outs in these macro ways. You build something and you kind of roll it out, you have a general availability for, and then you take the time. And that's really important too because, again, we are talking about pharma companies, legal companies relying on this. So we do want to have that. But at the same time, given the compressed cycles of AI, how do we start to have people experience what's the one year in the future?

中文翻译:

是的。而且，这些是微观的……我们要看。这里的观点（Meta Point）是，传统上，跨公司、跨行业，我们总是以宏观的方式思考发布。你构建一个东西，然后发布，有一个“正式发布（GA）”阶段，然后慢慢推进。这确实很重要，因为我们谈论的是制药公司、法律公司在依赖这些工具。所以我们需要严谨。但与此同时，考虑到AI极其压缩的周期，我们如何让人们开始体验“未来一年”的生活？

[00:13:42] Lenny Rachitsky

English:

Let's follow this thread in a few different directions. There's how product development changes, there's how engineering changes. There's also just agents. I know you're spending a lot of time in agents, feels like you're not an AI company these days if you're not working on agents or building an agent.

中文翻译:

让我们沿着这个思路往几个方向深入。包括产品开发如何改变，工程如何改变，还有智能体（Agents）。我知道你在智能体上花了很多时间，感觉现在如果你不研究智能体或构建智能体，你都不算是一家AI公司了。

[00:13:42] Aparna Chennapragada

English:

Lenny, you're doing this wrong. You didn't use the word agents so far into the conversation.

中文翻译:

[00:13:51] Lenny Rachitsky

English:

I try hard to push it out as far as I can. It's like every conversation in San Francisco, it's just like how long until I start talking about AI? It's like three minutes. Average, I bet. Oh, man. Okay, so with agents, I know that you're leading a lot of this work at Microsoft and a lot of people are wondering what the hell does this mean? What is going to change? Give us just a glimpse into how you see the world being different in a world of agents being around more.

中文翻译:

我尽量把它推迟。就像在旧金山的每一次谈话，大家都在赌多久会开始聊AI。我敢打赌平均也就三分钟。噢，天哪。好吧，关于智能体，我知道你在微软领导很多这方面的工作，很多人都在想：这到底意味着什么？会有什么改变？请给我们描绘一下，在智能体更加普及的世界里，你认为世界会有什么不同。

[00:14:18] Aparna Chennapragada

English:

There's a short term and there's a long term, right? There's a lot of hyperventilated, excited talk about the eventual future and all of that. I take a much more practical product building lens on this, and I think about these. At the end of the day, there are tools. Yes, underneath it, there's stochastic models versus very deterministic programming models. You can tell I'm a computer scientist like the way that worldview definitely shapes how I think about this. To me, the short term is there's an evolution. We had apps, and now I think we are firmly in the assistance era where there's human driving the... That's what we think of as co-pilot, right? I think the human driving in the driver's seat but having a lot of assistance from AI.

(00:15:07):

So I think of this as then you look at the dimension of almost autonomy and delegation and intelligence. As the intelligence, for example, when deep reasoning unlock happened, of course, then you can delegate more to the agent. So I think, to me, there's one dimension where you say, "Hey, agents are somewhat independent software processes that can kind of run tasks," and you're not just thinking about handholding and fine motor stuff. You're saying, "Hey, here's my goal. Go make this happen." I'll give you an example. So we are working on this researcher agent for work. And last night, I said, "Hey, I have an important meeting coming up with the leadership team. I really want to present these frameworks here and this is the roadmap here. Go back and look at all the people that are in the meeting. What are their views on this topic and come up with how I should I be thinking about the right persuasion pitch here?"

(00:16:07):

And what's magical about this is not just that it's saving time. Typically, we think about the, so far, AI as summarizing a document or saving time. This is like fighting synapses that I didn't quite have and actually giving me new insights and giving me, dare I say, superpowers. So that's a natural evolution of AI, I would say. So when I think about agents, I think about three things. One is an increasing level of autonomy and kind of independence that you can delegate higher and higher order tasks. Second thing I think of it is complexity. So it's not just a one-shot, "Hey, create this image or do this thing or summarize the document," it's build me this prototype that expresses my idea of, say, an augmented reality app. It's a complex task. And then the third thing I would say is asynchronous. It works when you are not working. I think that's the other big thing about these things that you don't have to sit in front of it.

中文翻译:

这有短期和长期之分，对吧？现在有很多关于最终未来的过度兴奋的讨论。我用一种更务实的产品构建视角来看待这些。归根结底，它们是工具。是的，底层是随机性模型（Stochastic Models）与非常确定性的编程模型。你能看出来我是个计算机科学家，这种世界观决定了我思考问题的方式。对我来说，短期内是一种进化。我们曾有过 App 时代，现在我认为我们正处于“助手（Assistance）”时代，由人类主导……这就是我们所说的 Copilot（副驾驶），对吧？人类坐在驾驶座上，但得到 AI 的大量协助。

(00:15:07):

所以我从自主性、授权和智能这几个维度来看待它。随着智能的提升，例如当深度推理（Deep Reasoning）被解锁时，你当然可以向智能体授权更多任务。所以对我来说，一个维度是：智能体是某种独立的软件进程，可以运行任务，你不再只是考虑手把手的微操。你会说：“嘿，这是我的目标，去实现它。”举个例子，我们正在开发这个工作研究员智能体。昨晚我说：“嘿，我有一个重要的领导层会议。我想展示这些框架和路线图。去查一下参会的所有人，看看他们对这个话题的看法，然后帮我构思一下该如何进行说服性的陈述。”

(00:16:07):

这其中的神奇之处不仅在于节省时间。通常我们认为目前的 AI 只是总结文档或节省时间，但这就像是激活了我原本没有的神经突触，给了我全新的洞察，甚至给了我——敢说吗——超能力。我会说这是 AI 的自然进化。所以当我想到智能体时，我会想到三件事：第一，自主性和独立性水平的提高，你可以授权更高阶的任务；第二，复杂性，它不再只是单次交互（One-shot），比如“生成这张图”或“总结文档”，而是“帮我构建一个表达我增强现实应用想法的原型”，这是一个复杂的任务；第三，异步性，它在你休息时也在工作。我认为这是这些东西的另一个重大特点，你不需要一直守在它面前。

[00:17:10] Lenny Rachitsky

English:

This answers the question of what is an agent essentially, these three bullet points. So what are the three again?

中文翻译:

这三个要点本质上回答了什么是智能体。能再重复一下那三点吗？

[00:17:10] Aparna Chennapragada

English:

When I think about agents, I think about these three things. So one, it's autonomy like being... And it's a spectrum, it's not a zero-one, it's how do I actually delegate things that it can do. Second, I think of as complexity. It's not a one-shot, "Hey, summarize this document, generate this image, but it's build me this prototype or help me knock this meeting out of the park." And then the third one I think of is it's a much more natural interaction. That doesn't just mean chat, but it may be actually jumping on a meeting with the agent and being able to talk through all of it or point it to things that I wanted done differently. So I think all three things, the autonomy, the complexity, and the natural interaction are at least product principles that will shape really good ones, good agents.

中文翻译:

当我想到智能体时，我会想到这三点：第一，自主性。这是一个光谱，不是非黑即白的，而是我如何实际授权它去完成任务。第二，复杂性。它不是单次的“总结文档”或“生成图片”，而是“帮我构建原型”或“帮我在这场会议中大获全胜”。第三，更自然的交互。这不只是指聊天，可能实际上是和智能体一起开会，能够讨

论所有细节，或者指出我希望它以不同方式完成的事情。所以我认为自主性、复杂性和自然交互，是塑造优秀智能体的产品原则。

[00:18:10] Lenny Rachitsky

English:

That is really helpful. Along this line of agents, there's this acronym that you taught me as we were chatting ahead of this podcast, NLX, what is that and how does that relate to agents and why are people not thinking about this enough?

中文翻译:

这非常有帮助。顺着智能体这个话题，我们在播客前聊天时你教过我一个缩写词 NLX，那是什么？它与智能体有什么关系？为什么人们对它的思考还不够？

[00:18:10] Aparna Chennapragada

English:

Oh, that's one of my Roman empires these days. The natural language interface. NLX is the new UX. Here's the deal. To me, I think traditionally we've thought very consciously about GUI because the graphical interfaces are not something natural, and so they have had to be explicitly designed, but they're rigid interfaces. What we have with conversational interface and natural language is it's a much more elastic, right? That doesn't mean it's not designed. Often, I hear a product builders say, "Oh, yeah. With AI, the model leads to the product. So it's just you chat with it." You and I are having a conversation, it's a podcast. I'll have another conversation at Microsoft and that's a meeting.

(00:18:59):

So conversations also have grammars, they have structures, they have UI elements, they're invisible. And so one of the things that I see and I'm really excited about is what are the new principles, new constructs in natural language as an interface? I'll give you a few examples. And actually a lot of startups as well as big companies are really experimenting with this stuff. One is if you think about it, prompt itself is a new construct and that's a new UI element just like a dropdown was or a menu was. But others that are emerging, especially for agents, I think are plans. So when you give a high level goal, what we are seeing is that when the agent comes back with a plan, preferably an editable plan, that's a new construct.

(00:19:44):

The other one that I think about a lot is showing the work, progress. You see this with different products. You see with the Copilot, you see with ChatGPT, DeepSeek, this idea of thinking aloud and it's kind of showing the work, but how much do you do it? If it's too verbose, it feels like I'm running some cron job and scripts, but if it's too terse, then I don't know if it's going in the right path, and I don't have the confidence yet. So there are all these new elements. So if you are a product whittler, this is a fun new space to be digging in for product design.

中文翻译:

噢，那是我最近的“罗马帝国”（指脑子里挥之不去、经常思考的事）。自然语言界面（Natural Language Interface）。NLX 就是新的 UX。情况是这样的：传统上，我们非常自觉地思考 GUI（图形用户界面），因为图形界面不是天然存在的，必须经过显式设计，但它们是僵硬的界面。而对话式界面和自然语言则更有弹性，对吧？但这并不意味着它不需要设计。我经常听到产品开发者说：“噢，有了 AI，模型就是产品，你直接跟它聊天就行了。”你和我正在交谈，这是播客；我在微软会有另一种交谈，那是会议。

(00:18:59):

对话也有语法、结构和 UI 元素，只是它们是隐形的。所以我看到并感到兴奋的一点是：作为界面的自然语言，有哪些新的原则和结构？我给你举几个例子，实际上很多初创公司和大公司都在实验这些。第一，Prompt（提示词）本身就是一个新结构，它是一个新的 UI 元素，就像以前的下拉菜单或菜单栏一样。但对于智能体来说，正在出现的其他结构还包括“计划（Plans）”。当你给出一个高层目标时，我们看到智能体返回一个计划（最好是可编辑的计划），这就是一个新结构。

(00:19:44):

另一个我经常思考的是“展示工作过程（Showing the work）”。你在不同的产品中都能看到这一点，比如 Copilot、ChatGPT、DeepSeek，这种“出声思考（Thinking aloud）”的想法，它在展示工作进度。但展示多少才合适？如果太啰嗦，感觉像是在跑脚本；如果太简略，我又不知道它是否走在正确的路径上，缺乏信心。所以这些都是新元素。如果你是一个产品打磨者，这是一个非常有趣的、值得挖掘的产品设计新空间。

[00:20:47] Lenny Rachitsky

English:

This is really interesting because I think people chat with all these chat bots and it just feels like this is just the way it is, but you actually are designing every element of the interaction, how much to share, but how much you're thinking, here's my plan, what do you think. So I think this will surprise a lot of people, just realizing there's so much that goes into just designing even these what seemingly are simple conversations.

中文翻译:

这非常有趣，因为我觉得人们在和这些聊天机器人聊天时，觉得“本来就该是这样”，但实际上你们在设计交互的每一个元素：分享多少信息、展示多少思考过程、“这是我的计划，你觉得呢”等等。我想这会让很多人感到惊讶，意识到即使是这些看似简单的对话，背后也有如此多的设计工作。

[00:20:47] Aparna Chennapragada

English:

Yeah. Another good example is follow-ups, right? You could say, "Look, you asked me a question," and then I could ask a follow-up set of things, and that's explicitly should be designed for success. So for example, if I said, "Hey, create an image," and it created a black and white like a clip art version of something. What are the next obvious follow-ups that it should be suggesting proactively? Now, too much and you are kind of annoying me, but too little in some sense, you've lost an opportunity to direct me or guide me into a happy path here.

中文翻译:

是的。另一个好例子是“后续追问（Follow-ups）”。你可以说：“你看，你问了我一个问题，”然后我可以问一系列后续问题，这应该为了成功而进行显式设计。例如，如果说“生成一张图片”，它生成了一个黑白的剪贴画版本。那么它应该主动建议哪些显而易见的后续操作？建议太多会让人烦，但太少则意味着失去了引导用户走向“快乐路径”的机会。

[00:21:25] Lenny Rachitsky

English:

This resonates a lot with when we had Kevin Weil on the podcast, he talked about this question of just how much to show about what you're saying. And it's interesting that DeepSeek went the extreme of just showing everything and people liked it too. I think that was interesting.

中文翻译:

这让我想起 Kevin Weil 上播客时提到的，关于展示多少思考过程的问题。有趣的是，DeepSeek 走向了极端，展示了所有内容，而人们竟然也很喜欢。我觉得这很有意思。

[00:21:39] Aparna Chennapragada

English:

Yeah, and I think it's a point in time thing too, Lenny, because in some sense right now these things are such black boxes. They're almost like peeking under the hood for anything. Even if it's verbose feels like, "Oh, I know what's happening," especially because the compute inference time, it's taking long to think. So it just feels like if you just went silent, I'd be very uncomfortable, I think.

中文翻译:

是的，我认为这也是一个阶段性的现象，Lenny。因为在某种意义上，现在这些东西还是黑匣子。能窥视一下内部情况，即使很啰嗦，也会让人觉得“哦，我知道发生了什么了”。特别是由于计算推理需要时间，它需要思考很久。如果它只是保持沉默，我想我会感到非常不安。

[00:22:02] Lenny Rachitsky

English:

I know.

中文翻译:

我懂。

[00:22:05] Aparna Chennapragada

English:

Exactly. So I do feel like there's that point in time, but over time, I also feel like this is an area ripe for personalization. For example, again in human, my API would be very different from somewhere. My interface is probably different from others, and I might just want the direct, "Hey, give me the TLDR," versus the, "Oh, so I went here and then I went there," and I'm like...

中文翻译:

没错。所以我觉得这是现阶段的特点，但随着时间的推移，我觉得这个领域非常适合个性化。例如，就像人与人之间，我的“API”可能和别人很不一样。我的界面偏好也不同，我可能只想要直接的“太长不看 (TLDR)”版总结，而不是听它说“哦，所以我先去了这里，然后又去了那里”，我会觉得……

[00:22:28] Lenny Rachitsky

English:

Following the start a little bit. We're talking about just how the future is going to be different. There's designing for these chat experiences, there's agents, kind of zooming out to just product development in general, it feels like you're at the forefront of a lot of the tools that are going to change the way we build products and also your teams are working with a lot of these tools that no one else has access to. So let me just ask, how do you see the future of product development being different from today most, and what do you think product builders should be preparing for doing to succeed in that future?

中文翻译:

顺着刚才的话题。我们正在讨论未来会有什么不同。包括设计这些聊天体验、智能体，放大到整个产品开发领域，感觉你正处于改变我们构建产品方式的工具前沿，而且你的团队正在使用很多别人还没法使用的工具。所以我问一下，你认为未来的产品开发与今天最大的不同是什么？你认为产品构建者应该做哪些准备才能在未来取得成功？

[00:22:59] Aparna Chennapragada

English:

I'll start with one stark statement that I say internally and externally, and I am trying to live it is that in this day and age, if you're not prototyping and building to see what you want to build, I think you're doing it wrong. I call it the prompt sets of the new PRDs. I really insist on folks saying if you're building new projects, new features of course come with prototypes and prompt sets. And I think the notion is not to say, "Hey, now everybody's just a biggest version of a software engineer." It is to say you have the fastest path to seeing and experiencing what's in your mind to be able to communicate, right? It's a much more high bandwidth way of communication. I think about that as a really a loop accelerator in terms of product building. That's number one. When in doubt, as someone put it, demos before memos, right? I think that's really number one.

(00:24:04):

I would say number two, this one is a little bit tricky I'd say, is that what I'm seeing is that the time to first demo is much shorter, but the time to a full deployment is going to take longer. So I think that there's going to be an uneven cadence. So typically, I think there was much more of a you've been this thing, you take a few weeks and then you can iterate and so on. But that inner loop of prototyping and iterating and getting even user research through AI conversations, all of that gets shortened. But I think the bar for scale, therefore becomes much high. In some sense, if you look at it, there's going to be a supply of ideas, a massive increase in supply of ideas in prototypes which is great. It raises the floor, but it raises the ceiling as well. In some sense, how do you break out in these times that you have to make sure that this is something that rises above the noise? So I would say that it's simultaneously thinking about not chasing after every idea. I think is the second one.

(00:25:14):

I'd say the third thing is there's a lot of conversation around full stack builders. What does the team of the future look like? A product building team. What I think about is I think that is inevitable in terms of there will be a few folks that are, especially at the prototyping early idea discovery stage that the lines of blurred, there'll be a few taste makers at the same time. I think you can still have a lot of people experimenting. It becomes even more important to have that editorial and taste making in a Frontier, one or a few at the heart of it because otherwise you just have Frankenstein product. That definitely doesn't change.

(00:25:58):

I have one other additional bonus thing, which is a lot of folks think about, "Oh, don't bother studying computer science or the coding is dead," and I just fundamentally disagree. If anything, I think we've always had higher and higher layers of abstraction in programming. We don't program in assembly anymore. Most of us don't even program in C, and then you're kind of higher and higher layers of abstraction. So to me, they will be ways that you will tell the computer what to do, right? It'll just be at a much higher level of abstraction, which is great. It democratizes. There'll be an order of magnitude more software operators. Instead of Cs, maybe we'll have SOs, but that doesn't mean you don't understand computer science and it's a way of thinking and it's a mental model. So I strongly disagree with the whole coding is dead.

中文翻译:

我先抛出一个我在公司内外都会说的鲜明观点，而且我也在努力践行：在这个时代，如果你不通过原型设计和实际构建来观察你想做的东西，我认为你的方法就错了。我把这称为“Prompt 集是新的 PRD（产品需求文档）”。我坚持要求大家在构建新项目、新功能时，必须带着原型和 Prompt 集。我的意思不是说“现在每个人都变成了软件工程师”，而是说你拥有了最快的路径去观察和体验你脑海中的想法，并以此进行沟通。这是一种更高带宽的沟通方式。我认为这是产品构建中的“循环加速器”。这是第一点。正如有人所说，“Demo 胜过备忘录（Demos before memos）”。

(00:24:04):

第二点，这有点微妙：我观察到，做出第一个 Demo 的时间大大缩短了，但完成全面部署的时间反而变长了。所以节奏会变得不均匀。通常情况下，你做一个东西，花几周时间，然后迭代。现在，原型设计、迭代、甚至通过 AI 对话进行用户研究的内部循环都缩短了。但规模化（Scale）的门槛因此变得更高。某种意义上，想法和原型的供应量会大规模增加，这很棒，它提高了底线，但也拉高了天花板。在这样的时代，你如何脱颖而出？你必须确保你的东西能从噪音中突围。所以第二点是，要同时思考如何不盲目追逐每一个想法。

(00:25:14):

第三点是关于“全栈构建者”的讨论。未来的产品团队长什么样？我认为，在原型设计和早期想法探索阶段，职能界限模糊是必然的。同时会有一些“品味制造者（Taste Makers）”。你可以让很多人进行实验，但拥有核心的编辑能力和品味变得更加重要，否则你只会做出一个“科学怪人”式的产品。这一点绝对不会改变。

(00:25:58):

我还有一个额外的观点：很多人认为“别费劲学计算机科学了”或者“编程已死”，我根本不同意。如果说有什么变化的话，那就是编程的抽象层级一直在提高。我们不再用汇编语言编程，大多数人甚至不用 C 语言，抽象层级越来越高。对我来说，你仍然会告诉计算机该做什么，只是在更高的抽象层级上，这很棒，它让技术民主化了。未来会有数量级增长的“软件操作员（Software Operators）”。也许我们不再叫工程师（SE），而是叫操作员（SO），但这并不意味着你不需要理解计算机科学。它是一种思维方式，一种心理模型。所以我强烈反对“编程已死”的说法。

[00:26:54] Lenny Rachitsky

English:

That's awesome. I love that. And SO is a software operator, what is that? What that stands for?

中文翻译:

太棒了，我喜欢这个观点。SO 是软件操作员（Software Operator），那是代表什么？

[00:27:00] Aparna Chennapragada

English:

Yeah, I just made it up but yes.

中文翻译:

是的，我刚编的词，但就是那个意思。

[00:27:04] Lenny Rachitsky

English:

Okay, cool. This idea of prototyping as being kind of core to building these days, is there anything you do within Microsoft to operationalize that and make that just a thing everyone has to do? Is it just culturally do it or is it like you must show me a prototype before you show me it.

中文翻译:

好，酷。关于原型设计作为当今构建核心的想法，你在微软内部有没有采取什么措施将其运营化，使其成为每个人必须做的事情？是纯粹靠文化驱动，还是说“在给我看文档之前必须先给我看原型”？

[00:27:20] Aparna Chennapragada

English:

Again, the future is here, unevenly distributed, even in Microsoft I would say, but there is certainly a strong cultural momentum and shift and desire to say, "Hey, let's actually look at live demos, live prototypes, and to even communicate the ideas. And to me, I mean, it's not always possible because obviously there are things that are deeply... If you're trying to change something in the bowels of Excel, you probably don't. There's even enough depth in the product that what you need to do, and you don't need to prototype that. But if you're especially thinking about new things and new products, new features, absolutely.

中文翻译:

还是那句话，未来已来，只是分布不均，即使在微软内部也是如此。但确实有一种强烈的文化势头、转变和愿望，大家会说：“嘿，让我们看看现场演示、活生生的原型，甚至用它们来交流想法。”对我来说，这并不总是可行，因为显然有些东西非常深层……如果你想改变 Excel 核心深处的某些逻辑，你可能不需要原型。产品本身已经有足够的深度，你只需要去做。但如果你在思考新事物、新产品、新功能，那绝对需要原型。

[00:28:01] Lenny Rachitsky

English:

Okay, let's talk about product management. There's this fear that emerged as soon as all these AI coding tools came out of just like PMs are dead, we don't need PMs. We could just build things ourselves. What are these people hanging around for? And what I found is it's actually the opposite that now that coding is easy. Now, the question is more and more, what should we be building? Why should we be building it? Is this right? Is this the right solution? Then getting adoption for it, which is what PMs are really good at. I feel like it's the opposite. PMs are the most important role. It'll change too, but let me get your take. Just what do you think the future of product management looks like? Do you think it's dead? Do you think it's going to thrive? Do you think it's going to change?

中文翻译:

好，我们聊聊产品管理。自从这些 AI 编程工具出现后，就有一种恐惧感：PM（产品经理）死定了，我们不需要 PM 了，我们可以自己构建东西，这些人留着干嘛？但我发现事实恰恰相反：既然编程变容易了，那么问题就越来越多地变成了“我们该构建什么？”、“为什么要构建它？”、“这正确吗？”、“这是正确的解决方案吗？”以及如何让用户采用它，而这些正是 PM 擅长的。我觉得 PM 反而成了最重要的角色。虽然角色会变，但我想听听你的看法。你认为产品管理的未来是怎样的？你认为它会消失，还是会蓬勃发展，或者会发生改变？

[00:28:41] Aparna Chennapragada

English:

Yes. Look, if you are a TPS report, mostly process person, and a lot of companies do get confused about product management and process and project management, I think then you do have a question of, "Hey, what is the value add here," especially if AI can read and write 50,000 meeting notes and track things and send emails and so on, but what I do think on the flip side is the taste making and the editing function becomes really, really important. In a world where the supply of ideas, supply of prototypes becomes even more like an order of magnitude higher, you'd have to think about what is the editing function here.

(00:29:34):

So that does mean that the bar is higher for product folks, but I think there's an interesting side effect I am observing in startups that I'm advising, companies and even within the companies that there used to be more gatekeeping I would say, in terms of like, "Oh, we should ask the product leader what they think." And again, there is a role for that editing function, but you have to earn it now. You just don't get it because of this title, but there's also just unlock of latent really good ideas from smart engineers, smart user researchers, smart designers who now have this expert in their pocket to kind of round out all the other things that they're not typically skilled at to bring forward their ideas and that's amazing, I think.

中文翻译:

是的。听着，如果你只是一个写报告、跑流程的人（很多公司确实把产品管理与流程管理、项目管理搞混了），那么你确实该担心：“我的价值在哪里？”特别是当 AI 可以阅读和撰写五万份会议纪要、跟踪进度、发邮件等等。但另一方面，我认为“品味制造”和“编辑功能”变得极其重要。在一个想法和原型的供应量呈数量级增长的世界里，你必须思考：这里的“编辑功能”是什么？

(00:29:34):

这意味着对产品人的要求更高了。但我观察到一个有趣的副作用，无论是在我担任顾问的初创公司还是在微软内部：以前会有更多的“守门人”行为，比如“哦，我们得问问产品负责人怎么想”。虽然编辑功能仍然存在，但现在你必须凭本事赢得这个地位，而不仅仅是因为头衔。同时，这也释放了聪明工程师、用户研究员、设计师心中潜藏的好想法，他们现在口袋里装了一个“专家（AI）”，可以帮他们补齐那些不擅长的技能，从而把想法推向前台。我认为这太棒了。

[00:30:25] Lenny Rachitsky

English:

And I think that expert, it's interesting, I'm working with an engineer on some stuff and he uses ChatGPT to even communicate to me in a more effective ways like, "Turn his pitch into something that will convince Lenny, this is a good idea."

中文翻译:

我觉得那个“专家”很有意思。我正和一个工程师合作，他甚至用 ChatGPT 以更有效的方式跟我沟通，比如“把他的提案变成能说服 Lenny 这是一个好主意的文案”。

[00:30:39] Aparna Chennapragada

English:

By the way, that is actually one of my common use cases, which is the WWXD I call it. What would X do? I use it to say, "Hey, what would Satya think about this particular set of conversations or ideas that we are pitching and so on." This is the power of, I think deep reasoning plus relevant context, right? This engineer you're talking about has that context about you and so it's kind of very interesting.

中文翻译:

顺便说一下，这实际上是我常用的用例之一，我称之为 WWXD (What would X do, X 会怎么做)。我会用它来问：“嘿，Satya (微软 CEO) 会怎么看我们正在推销的这组对话或想法？”我认为这就是深度推理加上相关上下文的力量。你提到的那个工程师拥有关于你的上下文，所以这非常有趣。

[00:31:17] Lenny Rachitsky

English:

If only everyone was as famous as Satya and had so much information out there, but I guess you can import all their emails or whatever tools exist to just understand from the conversations you've had with that person.

中文翻译:

要是每个人都像 Satya 那么出名，有那么多公开信息就好了。不过我想你可以导入他们的所有邮件，或者利用现有的工具，从你与那个人的过往对话中去理解他。

[00:31:17] Aparna Chennapragada

English:

Yeah. And I think this goes back to actually what you were saying too, which is I think this idea of what is the... There's like a coil spring. There's an intelligence overhang that I just see across the board. And I think the part of product development has to almost rewire ourselves to, I think, Tobi from Shopify calls it the reflexive AI usage. And that's not as easy, and I've been thinking about why. Basically, I mean, I have a cheesy Chrome extension. Literally whenever I open a new tab, it just says, "How can you use AI to do what you're going to do right now?" It's very cheesy, but it kind of helps to pause and think, "Oh, what am I trying to do here?"

(00:31:56):

But the reason I find it hard, and when I talk even people who are living and breathing in this space, they find it hard is that the updating of the priors is really hard. The models couldn't do some things one year ago. I mean, image generation was full of spellings or reasoning. You just couldn't have deeper and smarter answers. You couldn't do data analysis. So my impression of it from change, trying it a few months ago, that prior needs to be updated. And it's hard to do that, right? You have to do something almost counterintuitive and against the grain to say, "No, no, ignore what you learned about what this can or cannot do." The baby just grew up to be a 15-year-old in a month.

中文翻译:

是的。这又回到了你刚才说的话。我认为现在有一种“智能过剩 (Intelligence Overhang)”的现象。产品开发的一部分工作必须几乎重新连接我们的神经，Shopify 的 Tobi 称之为“反射性 AI 使用 (Reflexive AI usage)”。这不容易，我一直在思考原因。基本上，就像我那个“土土的”Chrome 插件，每当我打开新标签页，它就问：“你现在要做的事怎么用 AI 完成？”虽然很土，但它能帮你停下来思考：“哦，我到底想在这里做什么？”

(00:31:56):

但我发现这很难，即使是那些在这个领域深耕的人也觉得难，原因在于“更新先验知识 (Updating of the priors)”非常困难。一年前，模型还做不到某些事。比如当时的图像生成到处是拼写错误，或者推理能力不行，你得不到深度和聪明的回答，也没法做数据分析。所以我对它的印象停留在几个月前尝试的时候，而这个“先验印象”需要更新。这很难，对吧？你必须做一些几乎违背直觉、逆流而上的事情，告诉自己：“不，不，忘掉你之前学到的关于它能做什么或不能做什么的知识。”这个“婴儿”在一个月内就长成了 15 岁的少年。

[00:32:40] Lenny Rachitsky

English:

I think that last point is so important that we've tried these tools over the years. And so far, it hasn't been amazing and then all of a sudden it is, and you kind of don't know that and you've given up almost and things change.

中文翻译:

我觉得最后一点非常重要。这些年来我们一直在尝试这些工具，到目前为止表现平平，然后突然之间它就变得惊人了，而你可能还不知道，甚至已经快放弃了，结果事情发生了变化。

[00:32:53] Aparna Chennapragada

English:

I think that's actually... If you are a product builder listening to it, that's a really interesting arbitrage thing for you. If you can kind of cut against the grain and say, "No, I won't have that scar tissue around." This didn't work a few months ago and keep setting high expectations and demand more of the AI today, I think you can unlock more.

中文翻译:

我认为这实际上……如果你是一个正在听播客的产品构建者，这对你来说是一个非常有趣的“套利”机会。如果你能逆向思维，说：“不，我不会留下那些心理阴影 (Scar tissue)。”不要因为几个月前它不行就放弃，而是继续设定高期望，对今天的 AI 提出更多要求，我认为你能解锁更多潜力。

[00:33:15] Lenny Rachitsky

English:

There's a lot of alpha in doing that.

中文翻译:

这样做确实有很多超额收益 (Alpha)。

[00:33:18] Aparna Chennapragada

English:

That's right.

中文翻译:

没错。

[00:33:19] Lenny Rachitsky (Ad Segment)

English:

Today's episode is brought to you by Coda. I personally use Coda every single day to manage my podcast and also to manage my community... (00:34:45)

中文翻译:

今天的节目由 Coda 赞助。我个人每天都使用 Coda 来管理我的播客和社区…… (00:34:45)

[00:34:45] Lenny Rachitsky

English:

I'm going to come back to this cheesy plugin, say more about this. So this is a plugin that just lets you put a custom message on every new tab, and you have it say, how can you use AI to do this?

中文翻译:

我想回到那个“土土的”插件，再多聊聊。所以这是一个让你在每个新标签页上放一条自定义消息的插件，你让它显示“怎么用 AI 来做这件事？”

[00:34:45] Aparna Chennapragada

English:

Yeah, it's as cheesy as that. And it's interesting because it works. In the last few weeks alone, I've been doing this experiment to say, "Hey, how much more AI pill can I get?" Both at work and in personal life to say, "When I'm trying to do anything manual, should I be demanding the AI to do this?"

中文翻译:

是的，就是这么土。但有趣的是它很管用。就在过去几周，我一直在做这个实验，想看看我还能吃下多少“AI 药丸”。无论是在工作还是个人生活中，我都会问：“当我尝试手动做任何事情时，我是不是应该要求 AI 来做？”

[00:35:08] Lenny Rachitsky

English:

That's so cool. Do you know the name of this Chrome extension by any chance otherwise?

中文翻译:

太酷了。你知道这个 Chrome 插件叫什么名字吗？

[00:35:12] Aparna Chennapragada

English:

No. No. I built it.

中文翻译:

不知道，因为是我自己写的。

[00:35:13] Lenny Rachitsky

English:

You built a Chrome extension. That's so cool. Okay. Did you use AI to build it?

中文翻译:

你自己写了一个 Chrome 插件，太酷了。好吧，你是用 AI 写的吗？

[00:35:20] Aparna Chennapragada

English:

Of course.

中文翻译:

当然。

[00:35:21] Lenny Rachitsky

English:

Wow. Which tool did you use to do that? Some kind of Microsoft tool I imagine.

中文翻译:

哇。你用什么工具写的？我猜是某种微软的工具。

[00:35:25] Aparna Chennapragada

English:

Yes. No, actually, it was just like, I mean, I live in GitHub and GitHub Copilot, so I just was like, "Okay, let's go build this Chrome extension."

中文翻译:

是的。不，实际上，我一直用 GitHub 和 GitHub Copilot，所以我就想：“好吧，咱们来写个 Chrome 插件吧。”

[00:35:33] Lenny Rachitsky

English:

Are you releasing this for the general public?

中文翻译:

你会把它公开发布吗?

[00:35:36] Aparna Chennapragada

English:

No, I mean, that's the amazing thing. It took me like 10 minutes to do this.

中文翻译:

不，我的意思是，最神奇的地方在于我只花了 10 分钟就搞定了。

[00:35:43] Lenny Rachitsky

English:

Okay, let's link to it. Let's get it out there, open source this thing. Okay. You mentioned Satya, I have a question about this. So you're one of the very few people that have worked very closely with both Satya and Sundar at Google. Let me ask you this. How do their leadership styles differ, and is there just a fun story you could share about each of them?

中文翻译:

好吧，咱们给它个链接，把它发布出来，开源掉。好，你提到了 Satya，我有个问题。你是极少数既和 Satya 紧密共事过，又和谷歌的 Sundar 紧密共事过的人。我想问：他们的领导风格有什么不同？能不能分享一个关于他们每个人的趣事？

[00:36:02] Aparna Chennapragada

English:

Yeah. I do feel lucky to have a window into these two amazing leaders of this generation. I would say, I mean, again, no surprise as you'd expect from CEOs of multi-trillion dollar market cap tech companies, they are 99.99 percentile in almost every dimension you'd think of intellect, empathy, leadership, product, strategy. There are, of course, flavors of differences. I was the technical advisor for Sundar for the first... At Google and set up the office of the CEO there. And they're, again, a matter of time and context because there's a lot more consumer-oriented focus there. So what I did find Sundar great at it is being really calm and measured and thoughtful in terms of making sure that things have... Dealing with the complex ecosystems.

(00:36:57):

If you think about the phone ecosystem or even the search and publisher and advertiser ecosystem, it's a very complex ecosystem. He was a master at that. He's a master at that. And I think on Satya, I find it amazing the appetite he has for learning and fine tuning his mental models and just the zoom levels that he can operate at. The macro, the strategy, what's the game? Also the micro, "Hey, why are we not..." Here's a specific insight that I saw on Twitter, and you can count on the fact that he's ahead of pretty much everybody else in terms of spotting those early things too. So it's just been learning from the firehose as they put it.

中文翻译:

是的。我很幸运能近距离观察这两位当代杰出的领导者。我想说，正如你对数万亿市值科技公司 CEO 的预期那样，他们在智力、同理心、领导力、产品、战略等几乎所有维度上都是前 0.01% 的水平。当然，风格上有所不同。我曾担任 Sundar 的第一任技术顾问，并在谷歌建立了 CEO 办公室。那里的重点更多是在消费端，这取决于时间和背景。我发现 Sundar 非常擅长保持冷静、克制和周全，特别是在处理复杂的生态系统方面。

(00:36:57):

想想手机生态系统，或者搜索、出版商和广告商生态系统，那是非常复杂的。他是处理这些问题的大师。至于 Satya，我发现他学习的欲望以及微调自己心理模型的速度非常惊人，而且他能操作的“缩放级别”非常广。宏观上，他看战略、看局势；微观上，他会问：“嘿，为什么我们不……”他会关注我在 Twitter 上看到的某个具体洞察，你可以相信，他在发现这些早期苗头方面领先于几乎所有人。正如人们所说，这就像“对着消防栓喝水”一样的学习过程。

[00:37:39] Lenny Rachitsky

English:

What a cool opportunity to work with two incredible folks. Okay, let's go in a whole different direction. Let me just ask you this question that I've been asking people more and more. What's the most counterintuitive lesson that you've learned about building products that goes against common startup wisdom, common product building wisdom.

中文翻译:

能和这两位杰出人物共事真是太棒了。好，让我们换个方向。我想问一个我最近经常问别人的问题：在构建产品方面，你学到的最违背直觉的教训是什么？也就是那些违反常识性的创业智慧或产品构建智慧的东西。

[00:37:57] Aparna Chennapragada

English:

I don't know if it's as common as it should be, and it's like a counterintuitive thing, but I've repeatedly learned that when you're doing something new, zero-to-one, the temptation is to kind of think about... It's like that South Park episode. Step one, think about the problem. Step two, question-

中文翻译:

我不知道这是否像它应有的那样普遍，但这确实很违背直觉：我反复学到，当你做一个从零到一的新东西时，诱惑往往是……就像《南方公园》里的那一集。第一步：思考问题。第二步：问号——

[00:38:19] Lenny Rachitsky

English:

Underpants. I think it's Underpants, step one.

中文翻译:

内裤。我想第一步是“内裤”。（注：引用《南方公园》内裤侏儒的梗，讽刺没有逻辑的商业计划）

[00:38:19] Aparna Chennapragada

English:

Underpants. Exactly, right? So I do feel like there's a temptation to rush and say to go to scale before solve. So I've always said to my teams solve before scale. So what that does mean is there's a different posture and different mode when you're trying to solve a problem versus scaling something that's either post-product market fit or even at least in roughly in the ballpark. So to give you a couple of examples, I think when you look at the solved stage, there are wide lurches. You got to be very comfortable with the fact that you're day one thinking about, "Hey, a plant detection tool." And then day 15, you're like, "Oh, actually, the tech is really good for translating foreign language." By the way, this is not hypothetical. This is what we kind of looked at in Google Lens back then and said, "Okay, what is the intersection and so on?"

(00:39:17):

So from the outside, it looks like chaos, but actually, in the... And you should be very comfortable... Not only tolerant, I think you should have an appetite for that because the last thing you want is prematurely fix on one local hill. And then you're climbing that in start-ups and entire product areas and companies, big companies make that mistake and three years later you're like, "Oh, how do I get off this hill?" So I'd say that's one big counterintuitive. When you're trying to think about what mode you're in, are you in the solved mode? Are you in the scale mode? One example is kind of making sure that you're comfortable with the chaos. I think the other lesson I've learned is the danger of metrics. And I think again, if you have worked on Google Search or if you worked on Office products, you really have a very fine-grained sense of what are the metrics for this product?

(00:40:11):

You have the input metrics out, you have the whole shebang, but when you're looking at something zero-to-one. If you decide on a metric two prematurely, that's false precision first of all, right? I mean, CTR. When you have a thousand people, it doesn't mean anything. Retention also may not mean anything. So really being very wary of this big guy, big girl of grownup metrics as I call it, right? You are looking for more qualitative, the sound of click, and what is your... The other kind of the handler uses, what is your set timer and play music? So if you look at Alexa and Siri and Google Assistant and all these things, they had a very promising broad interface. You could say anything, but I think there was one or two things that it was really good at. You could set a timer, you could play music, and you could play trivia. And so you've got to nail those things before you say, "Oh yeah, here you can do anything with it," which is not a good recipe.

中文翻译:

内裤，没错。我觉得人们总有一种诱惑，想要在“解决问题”之前就先“规模化”。所以我总是对团队说：先解决，再规模化（Solve before scale）。这意味着当你试图解决一个问题时，你的姿态和模式应该与规模化一个已经达到 PMF 的产品完全不同。举几个例子，在“解决”阶段，会有剧烈的摇摆。你必须习惯这种状态：第一天你在想“做一个植物识别工具”，第 15 天你可能觉得“哦，其实这项技术更适合翻译外语”。顺便说一下，这不是假设，这是我们当年做 Google Lens 时的真实经历。

(00:39:17):

从外面看，这像是混乱，但实际上……你应该对这种状态感到舒适，甚至应该渴望它。因为你最不希望看到的就是过早地固定在一个“局部小山头”上，然后拼命往上爬。很多初创公司、整个产品领域甚至大公司都会犯这个错误，三年后你会想：“噢，我该怎么从这座山上下来？”所以这是第一个违背直觉的点：分清你是在“解决模式”还是“规模化模式”，并确保你能接受混乱。第二个教训是指标的危险性。如果你在谷歌搜索或 Office 部门工作过，你对指标会有非常精细的理解，有一整套输入输出指标。但当你做从零到一的东西时，过早决定一个指标首先会导致“虚假精度”。比如点击率（CTR），当你只有一千个用户时，它没有任何意义；留存率可能也没意义。所以要警惕我所说的“成年人指标”。你应该寻找更定性的东西，比如“点击的声音”，或者像别人说的“设定闹钟和播放音乐”。看看 Alexa、Siri 和 Google Assistant，它们都有一个看起来无所不

能的界面，但其实它们最擅长的只有一两件事：设闹钟、听音乐、玩问答。你必须先钉死这些核心功能，然后再说“你可以用它做任何事”，否则就不是一个好方案。

[00:41:11] Lenny Rachitsky

English:

Not so funny. That's exactly what I use my Google Home for, so basic. I don't do the trivia thing now maybe I got to give this shot.

中文翻译:

太逗了。这正是我用 Google Home 做的事，非常基础。我还没玩过问答，也许我该试试。

[00:41:20] Aparna Chennapragada

English:

Got to try that. Yeah.

中文翻译:

得试试，没错。

[00:41:21] Lenny Rachitsky

English:

There's something along these lines that I've also seen you talk about, which is how to go zero-to-one with something, just a little framework for helping you know if this is the right time for this idea. How do you think about that?

中文翻译:

关于这一点，我也听你谈过如何做一个从零到一的东西，有一个小框架可以帮助判断现在是不是实现这个想法的正确时机。你是怎么想的？

[00:41:33] Aparna Chennapragada

English:

Yeah. And when you think about the solved mode, and this is again sticking with my whole living in one year in the future, I gravitated towards the zero-to-one and solved mode products completely thinking about new category of products. And what I've found, both the hard way I would say, is that you do want to look for at least two out of these three factors, inflection points here if you want to make a really good product. Number one is there a... Shift is a step function in the tech. That's somewhat obvious I would say. Deep learning was one for Google lens. Back then, speech recognition was a step function for conversational search. I would say for Robinhood, the generational shift was very clearly, and the fact that phones were a primary means for you could actually have mobile app for finance that you could use. So look for that inflection. What is the tech inflection? And right now, of course, like LLMs and reasoning models are that step function, but that's not enough.

(00:42:35):

I would say the second factor that we should look for is, what is the consumer behavior shift? So to give you an example, when we started working on Google Lens, what we said is, "Look, people were taking mostly pictures for sharing, selfies and sunsets and so on. And suddenly, when storage became free, mostly free, and everybody had phones everywhere all the time, you took pictures of everything. And then you had enough of pictures or you use the camera as the keyboard for your world, for the real world. And so how do you then say, "Oh, this consumer shift is big, and so therefore, as you go order of magnitude more photos, then you want more to come out of them and you can apply AI to that."

(00:43:24):

And I'd say the third inflection point, particularly I would say in enterprise but also in consumer, is the business model shift. Is there an inflection natural inflection point in the business model? So any great products, if you think about all the way from search, again, the second price option and the fact that you had CPCs, same thing with SaaS and the fact that you could actually charge or monetize enterprise products in a different way. And with AI, of course the monetization is a whole different... We've just barely scratched the surface of whether you do seat monetization, usage like on tap, and then of course outcome-based stuff, outcome-based monetization. Hey, have you solved the problem for me and then I will pay you some fees. So all three to me are kind of like, great, but at least two out of three for a good product.

中文翻译:

是的。当你思考“解决模式”时，这又回到了我“生活在未来一年”的理念。我倾向于从零到一和解决模式的产品，完全思考新类别的产品。我发现（通过一些惨痛教训），如果你想做一个真正伟大的产品，你需要寻找这三个因素中至少两个拐点。第一，技术上的阶跃式变化（Step function）。这比较显而易见。对 Google Lens 来说是深度学习；当时对对话式搜索来说是语音识别。对于 Robinhood 来说，是代际转变，以及手机成为主要工具，让你可以在手机上使用金融 App。所以要寻找技术拐点。现在，大模型（LLM）和推理模型就是那个阶跃函数，但这还不够。

(00:42:35):

第二，我们要看消费者行为的转变。举个例子，当我们开始做 Google Lens 时，我们发现人们以前拍照主要是为了分享，拍自拍、拍日落。突然之间，当存储变得几乎免费，且每个人随时随地都有手机时，人们开始拍下一切。你有了海量的照片，或者说你把摄像头当作了现实世界的“键盘”。那么你如何利用这个巨大的消费者行为转变？当照片数量增加一个数量级时，你希望从中获得更多信息，这时你就可以应用 AI。

(00:43:24):

第三个拐点，特别是在企业级市场，但也包括消费级，是商业模式的转变。商业模式是否存在自然的拐点？想想伟大的产品，从搜索（二阶价格拍卖和 CPC 模式）到 SaaS（以不同方式对企业产品收费）。对于 AI，变现模式完全不同……我们才刚刚触及皮毛，是按席位收费、按使用量收费，还是按结果收费（Outcome-based）？比如“你帮我解决了这个问题，我付你一笔费用”。对我来说，这三个因素都很棒，但一个好产品至少要占两个。

[00:44:21] Lenny Rachitsky

English:

So this essentially... When investors look at startups, they're always asking, why now? Why is this the time to start this thing? And so your advice here is there's three ways to look at it. Two of these three should be true. There should be a shift in technology, some new technology that has enabled this now recently. There's a shift in consumer behavior, and then there's maybe a new sort of... Or you've invented a new business model, any way to monetize something that it gives you an advantage over folks trying to do it today.

中文翻译:

所以这本质上……当投资者看初创公司时，他们总会问“为什么是现在？”你的建议是，从三个维度来看，其中两个必须成立：技术变革（最近出现的新技术）、消费者行为转变，以及新的商业模式（或者你发明了一种变现方式，让你比现在的竞争对手更有优势）。

[00:44:51] Aparna Chennapragada

English:

Yep, absolutely.

中文翻译:

是的，完全正确。

[00:44:52] Lenny Rachitsky

English:

Awesome. You did mention Robinhood, I think in that example. That was another good example of phones-

中文翻译:

太棒了。你在例子里提到了 Robinhood，那是另一个关于手机的绝佳例子——

[00:44:56] Aparna Chennapragada

English:

Yeah, I mean, talk about the business model of, again, not having a zero fees. And again, that combination of all of these things is what can unlock it. You can't just say, "Oh, we'll just have a much more better intuitive interface and hope that people will switch to it."

中文翻译:

是的，我是说它的商业模式，比如零佣金。这些因素的结合才能解锁成功。你不能只是说“哦，我们做一个更直观的界面，然后希望人们会换过来用”。

[00:45:16] Lenny Rachitsky

English:

Okay, so speaking of zero-to-one products, I'm going to take us to a occasional segment on this podcast that I call Hot Seat Corner. And I have a question for you that is on my mind and it's come up in a couple recent podcasts actually. So there's these companies like Cursor, VZero, Lovable, Bolt, Replit that are the fastest growing company's history. I just saw that Cursor hit 300 million ARR in two years. Interestingly, you guys were very well positioned to do really well in this space, this AI coding tool space. You guys had Copilot, the first tool in the world at this stuff, so ahead of everyone. You build VS Code, which is what all these companies are forking to build on. You have incredible AI infrastructure, incredible AI talent. So this could have been your market. What happened? What happened, Aparna?

中文翻译:

好，说到从零到一的产品，我要进入播客的一个偶尔出现的环节，我称之为“热座角（Hot Seat Corner）”。我心里有一个问题，实际上在最近的几期播客中也提到过。现在有像 Cursor、VZero、Lovable、Bolt、Replit 这样的公司，它们是历史上增长最快的公司。我刚看到 Cursor 在两年内达到了 3 亿美金的 ARR。有趣的是，你们（微软）在这个 AI 编程工具领域本该占据极佳的优势。你们有 Copilot，这是世界上第一个此类工具，领先所有人。你们开发了 VS Code，而所有这些公司都是基于 VS Code 的分支（Fork）构建的。你们有顶级的 AI 基础设施和人才。这本该是你们的市场。发生了什么？到底发生了什么，Aparna？

[00:46:01] Aparna Chennapragada

English:

It's interesting, the framing... So I'm a dead user of GitHub Copilot, and I would say, "Look, if you unpack..." I think the beauty of this is that code generation has become an amazing tool that LLMs have unlocked. So it is actually really good excitement and action that now code generation has just opened up all of these things that... We talked about the whole idea of prototyping, goes from idea to marks and idea to a clickable prototype in a few minutes. Those are the kinds of things that, of course, we should expect code generation to enable. The way I think about how we are positioned and what we do with GitHub is... So it's a system, not just a product or a set of features.

(00:46:52):

If I think about GitHub, it's for folks who have the repo there and you have... Of course, you have the assistance in terms of autocomplete and you can chat, but now we have the agent board. It's one of the fastest loops that we are seeing, really strong positive feedback. So in some sense, when you have a system, what you are looking for in terms of building and designing it is not just a single product that can grow, but what is the repository? What is your context? What are the set of features that grow from your expertise? If you're a really expert coder, you want the assistance this product needs to scale for that. If you're a wide coder, you should still be able to do that and so on. So that I think is the way that GitHub is positioned to build on and grow honestly really well.

中文翻译:

这个提问角度很有趣。我是 GitHub Copilot 的忠实用户。我想说，如果你拆解来看，大模型解锁的代码生成已经成为一个惊人的工具。代码生成开启了我们之前讨论的所有可能性，这确实令人兴奋。比如原型设计，几分钟内就能从想法变成可点击的原型。这些正是我们期望代码生成能实现的事情。我如何看待我们的定位以及我们在 GitHub 所做的事情……它是一个系统，而不仅仅是一个产品或一系列功能。

(00:46:52):

如果你看 GitHub，它是为那些把代码库放在那里的人准备的。当然，你有自动补全的协助，你可以聊天，但现在我们还有了“智能体面板（Agent board）”。这是我们看到的反馈最快、最积极的循环之一。所以从某种意义上说，当你拥有一个系统时，你构建和设计的不仅仅是一个可以增长的单一产品，而是：代码库在哪里？你的上下文是什么？哪些功能是基于你的专业知识生长出来的？如果你是一个专家级程序员，你需要产品能为此提供协助；如果你是一个初级程序员，你也应该能使用它。我认为这就是 GitHub 的定位，而且坦白说，它增长得非常好。

[00:47:46] Lenny Rachitsky

English:

That's so interesting. So the core of this is everyone ends up in GitHub anyway, no matter what tool they use and that's kind of the-

中文翻译:

很有趣。所以核心是，无论人们使用什么工具，最终都会回到 GitHub，这就是……

[00:47:53] Aparna Chennapragada

English:

Yeah. The idea again is that code generation as a tool will unlock lot more products. I mean, they're not all competitors to the fact of... They're not all kind of doing the same job. I think when you are... At the end of the day, you are building code for companies to run on, you need to have a system. You need to have kind of the ability, an entire Swiss Army toolkit, not just the autocomplete, not just a chat, not just like a software agent that runs and you kind of hand hold. You need all of this to work together, and that's what the GitHub product is going after.

中文翻译:

是的。代码生成作为一种工具会解锁更多产品，它们并不全是直接竞争对手，也不全是在做同样的工作。归根结底，当你为公司构建运行代码时，你需要一个系统。你需要一整套“瑞士军刀”式的工具箱，而不仅仅是自动补全、聊天或一个需要你手把手教的软件智能体。你需要所有这些协同工作，这就是 GitHub 产品所追求的目标。

[00:48:30] Lenny Rachitsky

English:

All roads lead to GitHub. On the flip side of this question, there have been probably 5,000 startups that have tried to disrupt Excel and you guys just keep winning, so something there is working really well.

中文翻译:

条条大路通 GitHub。这个问题的另一面是，大概有 5000 家初创公司试图颠覆 Excel，但你们一直立于不败之地，所以那里的某些东西运作得非常好。

[00:48:46] Aparna Chennapragada

English:

That is so interesting you say that. So when I came to Microsoft, and I'm an Excel fan, so I actually had a conversation with one of the OG Excel product folks. I was like, "an, what is it about this product?" And he said a couple things that were really interesting for me that just stuck with me. One is and I said, "Hey, Excel is a proof that non-coders also have to program." Programming is really powerful and it's the tool that gives all of the non-coders a really powerful programming ability, and I thought that was just really striking.

(00:49:22):

And then the second thing that I found out super cool, I don't know if you know this, but I didn't know at least before two years ago that there are these amazing Excel championships like World Excel championships where you see folks who can do just magic. And to me, I think the insight here is also that some tools are harder to learn. Perhaps in the beginning there's friction in terms of learning, but great to use. So it's a very good case of, hey, the learning curve initially, the one-time learning curve might be tricky, but it is because there's so much power and depth in the tool.

中文翻译:

你这么说很有趣。当我来到微软时，作为一个 Excel 粉丝，我曾和一位 Excel 的元老级产品人员聊天。我问：“伙计，这个产品到底有什么魔力？”他说了几点让我印象深刻。第一，Excel 证明了非程序员也需要编程。编程非常强大，而 Excel 是赋予所有非程序员强大编程能力的工具，我觉得这非常震撼。

(00:49:22):

第二件我觉得超级酷的事——我不知道你知不知道，但我两年前才知道——竟然有 Excel 世界锦标赛，你能看到那些人像变魔术一样操作 Excel。对我来说，这里的洞察是：有些工具很难学，初期可能有学习摩擦，但用起来非常棒。这是一个很好的案例：初始的一次性学习曲线可能很陡峭，但那是因为工具本身蕴含了巨大的力量和深度。

[00:50:02] Lenny Rachitsky

English:

That's so interesting. I never thought of Excel as a programming language, but it makes sense and I feel like once you get used to it and this is just the way things work, you're kind of stuck there and everything else has to basically copy that model, which is hard to be as good.

中文翻译:

很有趣。我从未把 Excel 看作一种编程语言，但确实说得通。我觉得一旦你习惯了它，习惯了这种工作方式，你就离不开它了，其他所有产品基本上得模仿这个模式，而很难做得同样出色。

[00:50:13] Aparna Chennapragada

English:

Yeah. And I think the depth then the attention that the team has given, and again, that's the compounding effect over decades of working on deep, deep signal from people who depend on it day in and day out.

中文翻译:

是的。我认为这归功于团队投入的深度和关注，这是几十年来从那些日复一日依赖它的用户身上获取深度信号所产生的复利效应。

[00:50:29] Lenny Rachitsky

English:

Okay. To kind of start to close out our conversation, I want to ask this question around your career. I find that most people have one moment in their career that changes the trajectory of their career. It could be a manager they had, it could be a project they worked on, it could be just the job they landed. What would you say is the most pivotal moment in your career that eventually led you to becoming chief product officer at Microsoft?

中文翻译:

好。在结束对话之前，我想问一个关于你职业生涯的问题。我发现大多数人的职业生涯中都有一个改变轨迹的时刻。可能是一位经理，一个项目，或者仅仅是得到了一份工作。你认为你职业生涯中最关键的时刻是什么，最终引导你成为了微软的首席产品官？

English:

Actually, there is one moment where it was a turning point for me. I was in Google Search, I was working on this idea that I thought should just work and it didn't. I said, "Hey, these phones are becoming a thing. Personalization has to be important." So I probably banged my head against the wall for a year or so trying to make personalization work. And it turns out when you have a query that you put into Google Search, the personalization didn't matter as much. And so we disbanded the team, but then I think I started working on this product called Google Now, which was a twist on that, which said, "Hey, actually on the phone, we should be able to push content. It's not about searching with personalization." For example, if you have a flight coming up, we should be able to say, "Hey," connect the dots and say, "you should leave now given the traffic and where you need to go," and so on or if you're deeply interested in stand-up comedy with deadpan artists, you should check out Mitch Hedberg.

(00:52:00):

These are kind of these really moments that the smartphone should be smarter. So I let that product through the initial zero-to-one phase, and that was a pivotal moment. It made me realize two things. One, I really love seeing around the corner and kind of seeing where things go and building the product rise to the occasion way more than the scaling and sustaining products. Second, it's harsh, but being early is the same as being wrong. This is pre-LLMs, pre-deep learning a lot of the really amazing ideas in terms of next token predictor, et cetera. We'd been thinking of it but didn't have the horsepower to go... The interface was great, the intelligence wasn't there. And I'd say the third thing that stuck with me is I got to work with some really smart... They talk about talent density now, and I think really smart people who have gone on to do amazing things, and so it gave me a taste of what a small group of people can do.

中文翻译:

实际上，确实有一个时刻是我的转折点。当时我在谷歌搜索部门，我正在研究一个我认为理所当然会成功但却失败了的想法。我说：“嘿，手机正在普及，个性化一定非常重要。”我大概花了一年时间死磕个性化，结果发现，当你向谷歌搜索输入查询词时，个性化其实没那么重要。于是我们解散了团队。但随后我开始研发一个叫 Google Now 的产品，它是那个想法的变体，它说：“其实在手机上，我们应该能主动推送内容，而不仅仅是带有个性化的搜索。”例如，如果你即将有航班，我们应该能连接信息并提醒：“嘿，考虑到交通状况和你需要去的地方，你现在该出发了。”或者如果你对冷面笑匠的脱口秀感兴趣，它会推荐你去看看 Mitch Hedberg。

(00:52:00):

这些时刻让我觉得智能手机应该更聪明。我带领那个产品度过了最初的从零到一阶段，那是一个关键时刻。它让我意识到两件事：第一，我非常喜欢“预见未来”，观察事物的发展方向，并构建能应对未来挑战的产品，这远比规模化和维持现有产品更吸引我。第二，虽然很残酷，但“太早出现”和“做错了”是一样的。那是大模型和深度学习之前的时代，很多关于“下一个 Token 预测”等惊人想法我们都想过，但没有足够的算力去实现……界面很棒，但智能还没跟上。第三点让我印象深刻的是，我得以与一些极其聪明的人共事（现在人们谈论人才密度），这些聪明人后来都成就了伟大的事业，这让我领略了一个小团队能爆发出的巨大能量。

English:

It's such a great story because it didn't work out in the end. Google Now kind of went away. And by the way, I super remember that product. It was very cool. I remember looking at it was very delightful and happy. And so I also have this segment on the podcast called Failure Corner, where people share a story of failure and how that helped them. And I love this as a combination of those two.

中文翻译:

这是一个很棒的故事，因为它最终并没有成功，Google Now 后来消失了。顺便说一下，我非常记得那个产品，它很酷，用起来让人感到愉悦和快乐。我的播客里有一个叫“失败角（Failure Corner）”的环节，让人们分享失败的故事以及它是如何提供帮助的。我喜欢这个故事，因为它结合了转折点和失败。

[00:53:20] Aparna Chennapragada

English:

Yeah. I mean, I'm not going to lie. I think it was painful when you do that because you see the vision of what can be and what is, and sometimes it's hard limitations. Sometimes, in this case, it takes five years or 10 years to really unlock the intelligence, but sometimes it's one or two key click stops away from the product being great and part of figuring out is knowing when you're in what situation.

中文翻译:

是的。我不打算撒谎，当时确实很痛苦，因为你看到了愿景，看到了可能性，但现实却有硬性限制。有时，比如在这个案例中，需要 5 年或 10 年才能真正解锁智能。但有时，产品距离伟大可能只差一两个关键的点击步骤。搞清楚自己处于哪种情况，也是产品开发的一部分。

[00:53:50] Lenny Rachitsky

English:

How long was that period from starting out until just moving on and it's not working?

中文翻译:

从开始到最终意识到行不通并离开，这段时间有多长？

[00:53:54] Aparna Chennapragada

English:

Yeah, I would say in that case, one of the good things is, again, it led the foundation of... It was one of the foundations of the Google Assistant. And of course, as the LLMs step function happened now with Gemini, it kind of works out. And I think it's the same thing across the board, which is sometimes you want to figure out the invariance that do work that then go on to the next version of the product. And other times, you just have to start over.

中文翻译:

我想说，在那件事中，好的一面是它为 Google Assistant 奠定了基础。当然，随着大模型阶跃式的发展，现在的 Gemini 终于实现了那些想法。我认为各行各业都是如此：有时你需要找出那些确实有效的“不变量”，并将它们应用到产品的下一个版本；而有时，你只能推倒重来。

[00:54:26] Lenny Rachitsky

English:

Is Google Now the first agent before agents? That's what it feels like.

中文翻译:

[00:54:27] Aparna Chennapragada

English:

That was certainly the idea, but it is fascinating to me that the interface, that there, we had the opposite problem. Whether you think about all the voice assistants, the interface is like we overshot and the intelligence wasn't there. Today, I feel like there's an opposite problem. I think these things have amazing intelligence and the interface we have largely is like the AOL Dial-Up Modem Chatbot.

中文翻译:

那确实是当时的初衷。但让我着迷的是，当时我们遇到了相反的问题。无论是哪种语音助手，界面都做得太超前了，但智能没跟上。而今天，我觉得问题反过来：这些东西拥有惊人的智能，但我们的界面在很大程度上还停留在“AOL 拨号上网时代的聊天机器人”水平。

[00:54:55] Lenny Rachitsky

English:

We've covered a lot of ground. Is there anything that you wanted to chat about or leave listeners with, maybe a last nugget of wisdom before we get to a very exciting lightning round?

中文翻译:

我们聊了很多内容。在进入令人兴奋的闪电轮提问之前，你还有什么想聊的，或者想留给听众的最后一点建议吗？

[00:55:06] Aparna Chennapragada

English:

I think I would say one thing that I'm really excited about is this idea of figuring out how we as people and agents collaborate together. I think there's some great set of products and experiences to be reimagined. That's my other Roman empire, which is how do we actually have this co-working space where you have the humans and agents and how do you actually have an output that's much, much more significant than what any one of us or any few of us can produce?

中文翻译:

我想说，我非常兴奋的一点是：探索人类和智能体如何协作。我认为有一系列伟大的产品和体验值得重新构思。这是我的另一个“罗马帝国”：我们如何拥有一个人类和智能体共存的协作空间？我们如何产生远超个人或少数人所能创造的产出？

[00:55:40] Lenny Rachitsky

English:

Well, I need to hear more about this. When do you imagine a co-working space of humans and agents? What does this look like? Is this Microsoft teams or is this a physical place with little robots?

中文翻译:

我得听听更多细节。你想象中的人类与智能体协作空间是什么样的？它长什么样？是 Microsoft Teams 这种软件，还是一个到处是小机器人的物理空间？

[00:55:51] Aparna Chennapragada

English:

Oh, I had a thought of the physical place, but I am thinking a lot about... Right now, all of these experiences are very civil player, and I do think there's an opportunity to think about how do we... Again, I'm living one year in the future, how do we actually have collaborate with each other, but also with agents and really figure out, for example, what tasks can we delegate? What can we inspect? How do we actually have information that flows between people that agents can mediate, and so on.

中文翻译:

噢，我确实想过物理空间，但我更多在思考……目前，所有这些体验都是“单机模式”。我认为有机会去思考如何……再次强调，我生活在未来一年，我们如何不仅彼此协作，还与智能体协作？并搞清楚：哪些任务可以授权？哪些需要检查？信息如何在人与人之间流动并由智能体进行调解？等等。

[00:56:24] Lenny Rachitsky

English:

All right, I'm curious to see what you guys got cooking. With that, we've reached our very exciting lightning round. Are you ready?

中文翻译:

好吧，我很想看看你们在酝酿什么。那么，我们进入了非常刺激的闪电轮环节。准备好了吗？

[00:56:32] Aparna Chennapragada

English:

Let's do it.

中文翻译:

开始吧。

[00:56:32] Lenny Rachitsky

English:

Let's do it. First question, what two or three books that you find yourself recommending most to other people?

中文翻译:

好。第一个问题：你最常向别人推荐的两三本书是什么？

[00:56:38] Aparna Chennapragada

English:

Oh, I have recency bias, but I've been reading this book called The Brief History of Intelligence, phenomenal book and like lots of underlining from me. And I think it kind of... The premises too, it looks at the evolution of intelligence like human intelligence and the brain development and connects that to what we are seeing with AI.

中文翻译:

噢，我有“近因偏见”，但我最近在读一本叫《智能简史》(The Brief History of Intelligence)的书，非常精彩，我画了很多重点。它的前提是观察人类智能的进化和大脑发育，并将其与我们现在看到的AI联系起来。

[00:57:02] Lenny Rachitsky

English:

Do you have a favorite recent movie or TV show that you've really enjoyed?

中文翻译:

你最近有没有特别喜欢的电影或电视剧？

[00:57:05] Aparna Chennapragada

English:

Hacks. I've been watching this. It's about a woman who's a great standup comedian of... I think it's set in the fact that she grew up in the '70s and '80s and really tried to break through in an industry that hasn't traditionally been very friendly to women, so really fun and quirky.

中文翻译:

《天后与草莓》(Hacks)。我一直在看这个。它讲述了一个伟大的女性脱口秀演员的故事。背景设定在她成长于70、80年代，并努力在一个传统上对女性不太友好的行业中突围。非常有趣且古怪。

[00:57:31] Lenny Rachitsky

English:

Do you have a favorite product that you've recently discovered that you really love, could be an app, could be some physical?

中文翻译:

你最近有没有发现什么特别喜欢的产品？可以是App，也可以是实物。

[00:57:36] Aparna Chennapragada

English:

I do use a lot of Microsoft products, GitHub Copilot being one of them, but I think the one that maybe I'll pick is Granola, I think, is the name of the app. I found it really useful. I just gave it a spin the other day and I'm like, "Oh, this is really useful in terms of being able to, again, without being intrusive, just capture the thoughts, notes, and structure it, put some..." It felt like one of those things where, yep, the confidence of a few things like we were talking about like the transcription, real-time transcription tech

has gotten really good. Voice recognition is great, and then enough of the LLM magic on top of it to make it structured and contextual.

中文翻译:

我确实用很多微软的产品，GitHub Copilot 是其中之一。但我可能会选一个叫 Granola 的 App。我觉得它非常有用，我前几天试了一下，心想：“噢，这太好用了。”它能在不干扰你的情况下捕捉想法和笔记，并将其结构化。这就像我们讨论过的，实时转录技术已经非常成熟，语音识别很棒，再加上大模型的魔力，使其变得结构化且符合上下文。

[00:58:18] Lenny Rachitsky

English:

I am a huge fan of Granola. I'll give a quick picture here. If you become an annual subscriber of my newsletter, you get a year free of Granola for your entire company.

中文翻译:

我是 Granola 的超级粉丝。我顺便做个广告：如果你成为我时事通讯的年度订阅者，你可以为全公司免费获得一年的 Granola。

[00:58:28] Aparna Chennapragada

English:

Did not know that.

中文翻译:

这我还真不知道。

[00:58:29] Lenny Rachitsky

English:

There we go, and then just check that out, lennysnewsletter.com, and you click the word bundle and you'll see how to do that.

中文翻译:

没错，去 lennysnewsletter.com 点击 bundle 就能看到怎么操作。

[00:58:29] Aparna Chennapragada

English:

Very cool.

中文翻译:

非常酷。

[00:58:34] Lenny Rachitsky

English:

Very cool. Two more questions. Do you have a favorite life motto that you often come back to when you're dealing with something maybe you share with folks that they find useful as well in work or in life?

中文翻译:

很酷。最后两个问题。你有没有什么人生格言？当你处理事情时经常会想起它，或者分享给别人，让他们在工作或生活中也觉得有用？

[00:58:46] Aparna Chennapragada

English:

I have one. In fact, actually, this is my email signature for, I don't know, for the last 20 years or so. It says the best way to predict the future is to invent it. I think it's a quote by Alan Kay. I find it useful for two things. One is no one knows anything. When you think about all the folks who think about, "Hey, this is exactly how everything's going to look and this is exactly the sequence," and so on, I think there's no substitute to experientially building it. I think the second part is if you think there's something that should exist, go build it.

中文翻译:

我有一个。事实上，这大概是我过去 20 年来的邮件签名。它写着：“预测未来最好的方式就是去发明它。”我想这是 Alan Kay 的名言。我觉得它在两方面很有用：第一，没人知道未来到底什么样。当所有人都在预言“未来一定会是这样，步骤一定是那样”时，我认为没有什么能替代亲手去构建它。第二，如果你认为某样东西应该存在，那就去建造它。

[00:59:24] Lenny Rachitsky

English:

I love that. Final question. We've talked about standup comedy a bit. Is there a favorite under the radar standup comedian that you think people should go check out?

中文翻译:

我喜欢这个。最后一个问題。我们聊了不少脱口秀，有没有哪个还没大红大紫但你非常喜欢的脱口秀演员，值得大家去关注？

[00:59:34] Aparna Chennapragada

English:

Oh, there's a couple of them. So one, I think, there's an Indian American or I think a British Indian standup comedian. Her name is Sindhu Vee, super smart, mom comedy, and I think the other one that... This is definitely not under the radar, but I just love his stick is Nate Bargatze. He's just so good.

中文翻译:

噢，有几个。一个是印裔英国脱口秀演员 Sindhu Vee，非常聪明，擅长“妈妈类”幽默。另一个虽然已经很有名了，但我非常喜欢他的风格，那就是 Nate Bargatze。他真的太棒了。

[01:00:08] Lenny Rachitsky

English:

Aparna, this was amazing. Two final questions. Where can folks find you online if they want to reach out maybe and follow up on anything you shared and how can listeners be useful to you?

中文翻译:

Aparna, 这太精彩了。最后两个问题：如果大家想联系你或跟进你分享的内容，在哪里可以找到你？以及，听众可以为你提供什么帮助？

[01:00:08] Aparna Chennapragada

English:

You can find me on LinkedIn and Twitter. Aparna CD is the handle. I do post stuff a lot more on LinkedIn these days, so would love to hear thoughts, comments, conversations there. I'd say one thing that would be super interesting is if any of this stuff spark conversations, particularly around this, what can a small team with a lot of AI tools do or new products that folks are really excited about, saying that they should exist, hit me up.

中文翻译:

你可以在 LinkedIn 和 Twitter 上找到我，账号是 Aparna CD。我最近在 LinkedIn 上发的东西比较多，很乐意在那里听到大家的想法、评论和交流。我想说，如果这些内容激发了任何讨论，特别是关于“一个拥有大量 AI 工具的小团队能做什么”，或者大家觉得“应该存在”的新产品想法，请随时联系我。

[01:00:42] Lenny Rachitsky

English:

Amazing. Aparna, thank you so much for being here.

中文翻译:

太棒了。Aparna，非常感谢你能来。

[01:00:45] Aparna Chennapragada

English:

Thank you.

中文翻译:

谢谢。

[01:00:46] Lenny Rachitsky

English:

Bye, everyone. Thank you so much for listening. If you found this valuable, you can subscribe to the show on Apple Podcasts, Spotify, or your favorite podcast app... See you in the next episode.

中文翻译:

再见，各位。非常感谢收听。如果你觉得内容有价值，可以在 Apple Podcasts、Spotify 或你喜欢的播客应用中订阅……下期节目见。