1:15

Alexey

**This week, we'll talk about GPT, LLMs, and building NLP products. Interestingly, in the last year, we did not have any talks about the LLMs or GPT, but today is our second one in a row. The previous podcast interview was also about the LLMs. This is purely coincidental, we did not plan to have two interviews about LLMs, but this is how it happened. We'll try to cover something new that we did not cover previously.**

**We have a special guest today, Sandra. Sandra is not new to DataTalks.Club. You might have seen her answering questions in our Book of the Week event with her book about GPT-3, Building Innovative NLP Products Using Large Language Models. Sandra is an AI entrepreneur, evangelist, and community builder. She has done a lot of amazing things. She will probably tell us more about that today. Welcome to our podcast.**

2:26

Sandra

Thanks for having me, Alexey. Super, super excited to be here. Yeah, let's do this. I'm not surprised that it's the second interview about LLMs because they're everywhere right now. [chuckles]

2:40

Alexey

**Yeah. I mean, we are kind of late to the party, because everyone was talking about the LLMs half a year ago. I don't know how it happened, but we were a bit late. But now, we have two events in a row. Better late than never. The questions for today's interview were prepared by Johanna Bayer. Thanks, Johanna, as always for your help. Yeah, let's start with the interview.**

# Sandra's background

3:05

Alexey

**Before we go into our main topic of LLMs and GPT, let's start with your background. Can you tell us about your career journey so far?**

3:15

Sandra

Yes. In terms of when I joined the space – it was five years ago. I was always interested in artificial intelligence adoption, [meaning the questions of] what people will do with AI tools once they're out there, how it will impact their everyday life, how it will impact business, etc. I started by actually starting my own project – an accelerator for startups called Nextgrid – together with my co-founder, Mathias. With Nextgrid, we were aiming to support startups at the earliest stage so POC (or pre-POC) stage, that want to be more business-savvy and want to gain further financing and validate their concept further. Next to Nextgrid, we also co-founded a hackathon event platform. It used to be called Deep Learning Labs, but it got rebranded to Lablab.AI, so if you want–you can check it out. If you're looking to participate in hackathons that use the latest tech, that's the one for you.

So, through that, we basically tried to get closer to artificial intelligence ourselves, but also let others play with the latest tech out there. It was around 2018, so the access to models started getting easier and easier. We were able to spin up an environment, plug into Open AI Gym, and train a reinforcement learning lunar lander or something like that – so, make a hackathon out of it. We started doing these hackathons and at some stage, I stumbled upon the next-generation models. At that time, GPT-2 was out and GPT-3 was about to be out. I really wanted access to it… I have a friend named Bakz that I want to say “hi” to – he has a YouTube channel. He started making YouTube videos about GPT-3 and also how he got access to GPT-3 – how he got into the beta list. He got it by making a video about it, so I figured that I might do the same because I really wanted to play with it.

Basically, that's how my YouTube channel started. It actually works – I did two videos about different types of GPT-3 demos [that] I've seen out there at our hackathons, but also beyond. And I got access to it. So I started fooling around with it, basically. The channel got traction. I wasn't expecting such traction so quickly but there seemed to be already, back then… It feels like ancient history because that was like three years ago, [but there was] interest in exploring this subject. It was quite a small community at the time, but once you talked about it, you were pretty recognizable. I did all sorts of things – talked to the founders that were building with GPT-3, tested new tools that were coming out to the markets, played AI dungeon, and stuff like that. It was really fun – generating Christmas present ideas, things like that.

Then I got approached by my future book author, Shubham Saboo. He offered to write a report together about GPT-3. But then, the publisher we were working with offered that we write a book about it. For me, it was a great opportunity to kind of get out of my imposter syndrome and start diving deeper into the field. The book was talking about the business side of things, but also the tech side of things. Once I went down the rabbit hole, I just felt like, “This is really my domain and I don't want to get out of it.” [chuckles] So I stayed. And now I am working for Cohere, which is developing state-of-the-art LLMs, and I am building a community there. I'm [also] focusing on education about large language models, which brings me here today as well.

# Making a YouTube channel to break into the LLM space

8:08

Alexey

**That's quite a fascinating story – how you made a video to just get access to GPT-3. I'm wondering if today… I still don't have access to GPT-4, I think because I never used my API – I didn't put any money into it. Even though I'm paying for ChatGPT, I never put any money into the OpenAI API. So I still do not have access to GPT-4. I'm thinking if I make a video about GPT-4 now, it will not help. [laughs] Because the community is so large.**

8:40

Sandra

Yeah, it exploded. The scale of things is just mind-blowing. There are so many videos right now about this. But back then, three years ago, it was a handful of people just exploring it. It was very new and now, it's exploded, and the market adoption and interest are huge. We are in a very interesting time when it comes to LLMs.

9:10

Alexey

**How did you actually make a video without having access to GPT-3?**

9:15

Sandra

I was just talking about different types of use cases that I saw for the future. Also, different types of “play projects,” toy projects that people had at the time, that I found pretty fascinating. That seemed to have been enough to get me in. [chuckles] If you're looking for a hack for the future, [I say] try any angle you can to get somewhere, whether you know about it or not – whether you have access or not. Yeah, you can do that.

9:48

Alexey

**Maybe with GPT, it's kind of late because the community is pretty large now. But for the next big thing, it could be a good solution. Right? [Sandra agrees] Quite interesting.**

# The business cases for LLMs

10:00

Alexey

**So your book is about the business and tech side of GPT and LLMs, right? [Sandra agrees] What business cases are there for using GPT?**

10:13

Sandra

Yeah, that's a huge question, actually – right now, especially. At the time, when we were writing the book (that was three years ago) we were talking to folks across different verticals like the entertainment industry, creative studios, sales assistants, code generators, or chatbot providers. They were all using GPT models (text generation models) in their particular context. Already at the time, you could see both startups and more mature companies playing with it and trying to adopt it – slowly, slowly. But since then, so much has happened on the market, that I don't really see how… Next-generation models are not relevant to a particular industry. You can basically use them for any type of industry.

I'd like to say that with LLMs, we're kind of in this next technology wave, like we were in the past with a web browser – or even before with personal computers. We got this completely new type of tool to interact with our machines using human language. And through that, I can envision that we are just at the beginning of it, even though it's already been five years since transformer architecture (the foundational architecture for LLMs) was founded. But these were the very, very, very early days of the very long future business adoption, and all sorts of projects built on top of it. It's also worth mentioning that GPT types of architectures are trademarked by Open AI right now, But you can have similar generative pre-trained models that don't have “GPT” in the name but have very similar capabilities.

Overall, we just call them text generation models – things like Claude or Cohere's Command, or open source models that generate text that doesn't come from Open AI. These are text-generation models. Next to them, there's also another group of language models called text understanding models that… Well, they are text understanding models – they help machines to understand the text better – but they're called embedding models. With these two, you have all sorts of business cases right now. You can also merge the two (combine the two) to power even more creative use cases. Right now, roughly speaking, for text generation, you have very popular use cases like – any type of text manipulation. It can be copy creation, it can be support chat assistance, personalized chatbots, AI friends, AI therapists – [chuckles] any kind of use case where you converse with the models. It's very popular right now. Also, [use case] where you manipulate text.

When it comes to text understanding, a very popular use case is memory for these chatbot-type of systems, where the model is able to converse with you in a personalized way by remembering all the intricate details of your conversation or the context that you have given it so far – it has to translate all this into its own sort of way. It translates it to numbers, and then it retrieves it from there and keeps talking to you. Basically, this is what you use embeddings for a lot these days. Also, you can use them for semantic search, which is, roughly speaking, a new type of search engine where instead of keyword matching, you use semantic similarity between different words and sentences to retrieve information based on any query. This is a more accurate way to retrieve information that has a better chance of hitting precisely the user intent behind it. Because oftentimes, when we do a keyword search, we don't necessarily find what we're looking for. I mean, Google Search is a very common experience [that showcases] this – you're typing in something and then going to scan through all these results. Maybe on the first page, you'll find something most of the time, but a lot of this information is just not really relevant. With semantic search, it helps you to hit that sweet spot faster and more accurately.

15:53

Alexey

**You mentioned that you had a company Nextgrid, where you supported startups at the POC stage. Are you still doing any of that? Or are you fully focused on Cohere right now?**

16:09

Sandra

I'm fully focused on Cohere. However, I exist at the company, but I'm within the orbits, let's say.

16:18

Alexey

**The reason I'm asking about that is because I'm pretty interested in Nextgrid, or maybe other incubators. How many more use cases of GPT or LLMs do you see right now? Is it at 50% or 30%? Do you have any number for that? Are most of them LLM-based POCs now?**

16:42

Sandra

I mean, most of them. In the beginning, when we were founding Nextgrid, there were no startups that were trying out the adoption of text generation or text understanding models (LLMs). But right now… I mean, it's also because right now, it's kind of the most popular thing there is. From the VC side of things, folks are really interested in the big phrases like “generative AI” and “large language models,” so startups have a huge incentive to somehow figure out a way to use them in their own company if they want further financial support. So that's kind of how it works right now, as far as I know. But it's a great time – it's a wonderful time to go out there and explore different problems where you can apply LLMs and you can solve actual things.

My opinion is (and I think it's a popular one) that you first and foremost need your tool to be useful in order for it to be adopted by others and for others to start using it on a daily basis and just… stop envisioning life without it in the future. It takes a lot of insights into how people are doing things and what bothers them, to be able to find it and then use LLMs to solve it further. Copywriting is a great example. We have all sorts of social media right now – social media platforms – and it's overwhelming to be able to sustain a presence on YouTube, TikTok, Twitter, LinkedIn, and what have you. I probably missed something – Twitch, Discord. [chuckles] There are all sorts of places where you can be at right now. Threads! Oh, yeah, Threads, Instagram, Facebook, etc. So the reason these copy-generation platforms are so popular right now is because we just want to automate a lot of the writing for social media. We don't like that it can be repetitive, but it's important to communicate with the world these days so this is where LLMs can help a lot. That's just one example.

# LLMs as amplifiers

19:22

Alexey

**With this example – it's actually a pretty interesting example. It's good that you brought this up because to me, it feels like right now if we rely on our LLMs for generating content on social media – doesn't it become less authentic? Instead of just writing copy ourselves, we rely on an AI model to do that, and then we just flood all these social media platforms with generated content. So not only does it become less authentic, it kind of becomes repetitive. Sometimes you just can recognize that “Okay, this was clearly not written by a human.” Sometimes you just start seeing these generated posts, right? So what do you think about this? Is it actually a good idea to replace copywriters with LLMs? Or maybe instead of replacing them, we should somehow help them?**

20:24

Sandra

Yeah, I totally get the example of being able to tell whether something was generated by a model or not. I can see that myself on my social media feed lately, where I am sometimes very confident in being able to tell that, “Okay, this was probably done using ChatGPT or something similar.” So, in my opinion, the way it works is LLMs are an amplifier – a tool that amplifies already existing processes and executes already existing intent behind this process.

So if a given person doesn't particularly care about being authentic, but rather wants to get to a certain place, (number of followers, number of sales, etc.) they will not prioritize being authentic. They will prioritize getting the message across, doing it as fast, and as widely as possible. So you have a lot of inauthentic messages, even before LLMs. You had people just doing it for the sake of not connecting with others, but rather maybe selling it. Now they just have a way easier time doing that. [chuckles] So maybe you just see more of it. But the initial process was already there – the initial intent behind it was already there.

22:10

Alexey

**Yeah, I understand. What you're saying is that it's actually good for some purposes, and we already had this sort of generated content, except maybe before (when it was generated by humans) it was worse quality, and it was more expensive. Right now we can rely on LLMs (on GPT) to generate this content and maybe get more exposure for our content or our brand, right?**

22:37

Sandra

Yeah. I mean, it's helpful, especially when you're small or when you're at a certain scale, and you just want to grow bigger and bigger, which is usually what businesses want. So you can just do it faster – you can have better quality of the output. I still think that there is a lot of curation of the content generated by text generation models to be done before you press “Publish”. If you don't do that, then you risk being “discovered” [chuckles] by attentive readers. But yeah, I think if you want to have a consistent brand and consistent communication, you need to curate these things. You need to be the final decision-maker when it comes to the shape of things.

# The befits of keeping a human in the loop when using LLMs (AI limitations)

23:29

Alexey

**So instead of fully automating and just letting GPT go wild and generate all the social media posts, we still need to have a human in the loop – maybe a copywriter that edits these things before they go live (before we publish them). Right?**

23:45

Sandra

Definitely. I mean, it's not uncommon to see models hallucinate and just make up facts that are not accurate – that are not there in reality – or twist certain things or mix up a communication style, or…There are so many things that they can have a “bump” with, that I think it totally makes sense to have a human in the loop and always make sure that there's somebody very attentive – really invested in making sure that this content (whatever it is) is coming out as coherent, consistent, good for the brands or consistent with the brand. Otherwise, you risk some random stuff here and there.

My favorite example is when you type in “AI model” into Amazon search, and you get a lot of product descriptions (a lot of products) where people are just copy/pasting from a copy generation tool (AI model is a common phrase). However, they didn't edit it, they didn't delete [this part] – they just pasted it and published it on Amazon. And now we have a lot of descriptions including this phrase. That's pretty hilarious.

25:19

Alexey

**It usually says “As an AI model I cannot be sure that this thing is actually…” Sometimes it says this thing, “As an AI model, I cannot know the future.” Right?**

25:32

Sandra

Yeah, it tries to be a bit reserved when it comes to what it can and cannot do. That's usually having users in mind, because people are asking models about all sorts of things, and they personify the models very easily – they assign them a certain authority that the models don't have. So it's just a precaution to make sure that you don't take medical advice without consulting somebody from an AI model, or legal advice. There are all sorts of things that the model is not qualified for but is being asked about. So it tries to be helpful but with certain constraints as an AI model.

26:25

Alexey

**Yeah, I remember I wanted to talk to somebody about a contract I wanted to sign. Of course, I thought, “Hm… let me ask ChatGPT.” So I copied the contract, put it in there, and asked, “Hey, is there anything suspicious about this contract?” And then it said, “Oh, it looks like a typical contract, but as an AI model, I cannot give you legal advice. If you really want to be sure, talk to a lawyer.” Or something like that. Right?**

26:53

Sandra

Yeah. I mean, another thing is just copy/pasting your information and giving it to a tool that's…

27:02

Alexey

**Safe?**

27:03

Sandra

Yes. In terms of your personal dataset safety, it's also probably not the best idea. So you need to be a bit cautious with that stuff. The companies that are training and giving access to large language models, they understand that very well. They understand that users tend to assign more authority or personality to the model than it actually has and just try to make sure that there are some guardrails. Of course, they won't always work, and they are very annoying at times when you are looking for help, and you cannot get it, because the models just keep saying, “I'm sorry, I don't know. I cannot say. I'm just a simple AI.”

27:54

Alexey

**It is very annoying.**

# Using LLMs as assistants

27:56

Sandra

Yeah, I know. I know. But then… I think the user experience there will get better with more specialized text generation models toward specific use cases. In the example of your contract, you will be able to have a conversation with a chatbot that has the same capabilities but is trained in specific laws. I don't know which country this contract was written in – but it could be German law or the Polish law here, where I am.

Also, it would be secure enough so that you can have it on your machine without it going outside and using your data further. And also being able to interact with a chatbot that is as close to a legal assistant as possible. I think this is where we're heading, kind of. Maybe not when it comes to personal productivity, because GPT is still great for it, but when it comes to more professional productivity and trying to use it for any sort of business tasks.

29:18

Alexey

**I imagine it can be useful even for lawyers because they cannot remember all the laws. In Germany (in Berlin) it's pretty common to… For example, if I don't need a book, instead of throwing it away, I just put it on the street saying, “For taking”. It's a present that anyone can take. Recently I saw a box full of law books, and they are huge – they are super thick. They're gigantic books. And there were many of them – a pile of these books. So I was wondering, “Oh my god, it's good that I'm not a lawyer. I don't need to remember all these things.”**

**I imagine that for lawyers, it's pretty difficult because they need to go through these books – they need to know what things are in there to be able to help their clients. Probably, they would also benefit from such a model because it can just consume the book, digest the book, and then you can just ask questions like, “Here is the case I'm working on. Which book should I consult about this case?” And [the model] can say, “Okay, it appears that this book, page 10,059 talks about that.” Right?**

30:43

Sandra

Yeah. And then it's not only books, but the law keeps changing, right? There are new laws being added on top of it – some become extinct, so you need to keep up. Also, we have cases that are far from obvious, so it's far from just like searching for the information in the book and then being able to get a perfect answer for it. As a lawyer, I think having a helpful assistant in the form of a language model that you can just ask about a certain fragment of a contract, or a certain fragment of a given law quoted and then also help you reason through the situation. [It would] help you reason through the defense arguments or something like that – I think it's going to be super useful for lawyers and for other professions as well. It's useful for me already when…

You don't only use models for getting access to certain information or generating some sort of information that you want to use further but also as a sparring partner, basically – as a thought process aid, when you just want to get the ball rolling, start thinking about it, and then kind of help to think through it. Because dialogue is a very powerful way to think through things and being able to have your own personalized dialogue assistant plugged into any type of data that you need – I mean, who wouldn't want that? [chuckles]

# Building an app that uses an LLM

32:28

Alexey

**Yeah. So let's say I want to build an app using GPT or some LLM. How do I go about that? Say I have some idea… I think at the beginning of this interview, we talked about needing to talk to users, understanding their problems, and figuring out how exactly we can help. Let's say we did this. We found a problem that many users seem to struggle with, and now we want to create a POC around this problem – and we want to use an LLM for that. How do we go about doing that?**

33:03

Sandra

To get to the next stage, I will try to think about the model capabilities that you have access to, and then the type of the tasks that are needed in order to create this solution that solves this problem – there are certain steps to solve a particular problem. Are any of the capabilities of the model good for this type of task or this type of step? You need to try it out. You need to think through it. On the more pragmatic side of things, you also need to make a major decision about choosing your foundational model and with that, its reasoning capabilities, basically.

You can have models that are huge in size, huge in parameters as well, extremely capable, and very refined when it comes to their reasoning, but maybe slower or maybe not within the domain that you want them to be because they were trained on the specific data that [are not applicable to] your use cases. You also need to choose between a model that is an open-source model and a model that is a proprietary model. Here, your app flow will look completely different based on what this decision is.

If you choose an open source model, you will need to train it yourself – you will need to do the entire maintenance when it comes to keeping the model alive on your own premises, make sure it's up-to-date, make sure nothing breaks, etc. When it comes to proprietary models, it can be as easy as an API call to get the model's capabilities, but then you need to think about things like, “What is happening to my IP when I use a proprietary model?”

35:25

Alexey

**By “IP” you mean “intellectual property,” right?**

35:28

Sandra

Yes. Yes, exactly. “Where does my data go? What is happening to it? How much can I trust this model?” These are very important questions to ask yourself when you're selecting a proprietary model. Then there are also a bunch of compromises you need to make. I've already mentioned that you can have a big model that's the latest, the greatest – the GPT-4 and any other future champion – but it can be maybe too expensive to call it all the time. And maybe when you use GPT-4, it doesn't adapt to your business. It's also pretty slow at the moment. So maybe that will annoy the heck out of your users, and you need to address that. So perhaps for your use case, it's better to go for a model that is smaller in size and that will do the task just as well, but faster and cheaper. It all depends on what kind of task it is.

You basically need to try it. It's painful, but you need to go and try different types of models, compare the outputs, and compare the outcomes. Then you also want to optimize your prompts when you talk to the generation models – that's another ballgame when it comes to… It's an art and a science. There are people called “prompt whisperers,” [or] “AI whisperers” that are able to communicate with models very… [cross-talk]

# Prompt whisperers and how to improve your prompts

37:13

Alexey

**Prompt whisperers. [laughs]**

37:14

Sandra

Yes.

37:16

Alexey

**Okay. I've heard “prompt engineers,” but prompt whisperers? That's something new.**

37:21

Sandra

“Prompt engineer” is this new profession that's already out there. Basically, it's developers who are really skilled at creating prompts, comparing prompts, and making them useful in production for businesses. Prompt whisperers are folks that are at the top of the game – the ones that really can get the most out of the model using the smartest (the most efficient prompt). Because it makes a lot of difference how you structure your question, task description, what kind of flow it's going to have, whether the model can really quickly tune into what you're trying to do and follow you, or whether it's going to get confused.

There are so many things here that you can optimize for. Then, once you make a decision about your foundational model (that's one of the crucial things) then I think it's relatively easy to hook it up to any system – any setup that you may have. It may be because I'm in the LLM industry that I care so much about the LLM side of things. Perhaps there's a myriad of challenges that you, later on, need to encounter to be able to build this app, but from what I've seen, it can be pretty straightforward from there.

39:07

Alexey

**So the main thing we need to decide is whether we want to host a model ourselves, train our model ourselves, and do all this maintenance work, or we will rely on an external party and just make an API call. But for [the latter] case, we need to be mindful of what happens with our data. For example, in my case, when I copy/pasted the text of a contract – who knows what happens with this text? Maybe OpenAI uses this for training data and the next time somebody makes a prompt, it will just spit out my data. So I need to be mindful of that. And in the case of building an app, I need to be especially mindful because the data of users might be used by an external party, and we don't want that.**

**So we need to think about these things. We also need to select the right size of the model – should it be very powerful but slow, or should it be small but maybe optimized for a specific use case? Right? [cross-talk] So these are the things we need to think about. Then after that, we invest time into writing good prompts, right? That's mostly it.**

40:21

Sandra

That's mostly it. Then there's also fine-tuning. Models are trained on large amounts of data that come primarily from the internet. But then you have some sort of use case that requires specific lingo – you are building a chatbot assistant for a finance application and there are all sorts of financial language that this model needs to understand to be able to successfully execute tasks, and it doesn't yet because it wasn't part of its training data. Then, the fun thing is that it can quickly catch up to that. It's kind of like when you go through a general education and your reasoning skills are refined enough to be able to get into any domain and master the specialized lingo and also concepts within that domain and becoming able to really swiftly operate there.

This is the same as fine-tuning a model. You want it to go into a specific domain, so you give it a certain amount of data that will be applicable to your use case and you fine-tune it to make it better on the granular side of things when it comes to specific language, specific concepts, etc. The models differ here – there are models where you can do this very easily, there are models where it's not as obvious, there are models that you cannot fine-tune, and so on, and so forth. The choice of the foundational model will also impact the future ability to fine-tune it, and you need to take that into account.

42:24

Alexey

**I recently had a discussion about writing good prompts. I use GPT quite often, also for title generation. For example, for this podcast episode, I tried to generate a title with ChatGPT. Usually, when I say, “Okay, we need to generate a title for a podcast episode,” It typically comes up with titles like “An AI Journey with Sandra Kubelik,” for example. It tries to include the name. For people who don't know you, it might not be clear like, “Okay, what's there in this episode** for me?” **And then it generates 10 titles like that – with the name of the guest. What I usually do to avoid that is, I say in the prompt, “Do not include the name of the guest in the title.” But sometimes, despite that, it still includes the name of the guest even if I say, “Do not do this.” It still does that. And I had a discussion about that recently.**

**One of the suggestions there was that large language models, GPT in particular, have problems understanding negation – when you tell it not to do something, it does not always understand that. Instead, you should tell it what to do, instead of what not to do. I don't remember how exactly to fix that prompt – maybe instead of saying “Do not include the name of the guest,” you would tell it “We want the audience to understand how it will benefit them,” so you would include the reasoning behind not including this. I still have to try this and see how well it works. But, I'm wondering – do you have any other tips for creating good prompts? Perhaps you already experimented quite a lot with that and have a set of tips that you usually use for creating excellent prompts.**

44:32

Sandra

Yeah. I mean it depends so much on the model that you're using because the prompts do not necessarily translate very easily to other models. If you're optimizing for Coherence Command, it will not perform in the same way as Antropic's Claude. But the answer that is always there is iteration – you just need to try it a bunch of times, see how it performs, see which version use believe will work better, and maybe A/B test it with users. Because who knows, maybe the title that you think will be great will not be that great. [chuckles] Maybe there will be another one that will get more likes or a bigger reach and that's why the podcasts will get to more people. So what's always really useful is giving a model a few examples.

Text generation models are currently extremely savvy when it comes to what is called “zero-shot generation”. In other words, they're able to, without any kind of example, actually go and execute a certain task. But it's still very helpful to show it more precisely what kind of outcome you're expecting. In the case of the podcast title, maybe grab a bunch of titles (podcast titles, book titles, blog titles that you like, whatever) and give them to the model as inspiration – say, “Try to make it similar or as good as those.” I definitely agree with you that saying, “Don't do something,” doesn't necessarily end up being a successful command. [chuckles] And I kind of like it because it's good that models are rebellious – they need to be able to do their own thing. [Alexey laughs] But it's really useful to be very clear about the type of outcome that you want.

If you know that you like a certain style or that some you find something excellent, tell that to the model, and it will learn from there. If you want it to use certain keywords, tell it that. If you want it to optimize for SEO, tell it that, and it will do it. So the communication of your intent is very important and being able to be empathetic enough to share as much information as possible is also very important.

47:32

Alexey

**Empathetic to the model​?**

47:34

Sandra

Yes. To the model and to what it can do. Imagine that you're talking to your friend and you're asking about something. Your friends can come up with a title for the podcast that will not include the name, but will you like it? Maybe no, maybe yes. It will be a random, very subjective generation that you might not enjoy (or you might enjoy). But if you tell your friend about the type of effects you want to achieve – if you want the title to be catchy, if you want the title to be controversial, if you want the title to be very quickly attention-grabbing (because that's what we need in social media these days to get out there) then they will be able to navigate the situation better. Right? [Alexey agrees] So it's the same with models. It's all about just trying to empathize with it and trying to communicate as well as possible to get communication back that hits the spot that you're trying to achieve.

48:51

Alexey

**When you mentioned that we need to make a few iterations, I kind of started thinking about my iterations of the prompt for this particular use case of coming up with a title. At first, I thought, “Okay, I don't want to include names, but you still include them.” Then I thought, “What if I am polite and add 'please'? Will it help?” [chuckles] It did not. Then I thought, “Hmm… Let me try it in all caps. 'DO NOT INCLUDE THE GUEST NAME' as if I was shouting.” It did not help either. So unless you know that this is a thing you need to try – that you need to give examples – I would not necessarily come up with this tip myself at that moment. Sometimes you just need some practice.**

49:44

Sandra

[It requires] examples, but also context. So give it as much context as possible. Maybe show the outline of the questions you're trying to ask for this podcast. Maybe show the description (the blurb) that you've created internally for the team or for social media. Based on that, let it come up with something. I usually just give as much context as possible – as much as the context window allows me, basically. I just throw in everything and give it as much information [as possible] because usually, that's what works the best.

50:22

Alexey

**So usually it's smart enough to figure out what is important in this pile of information?**

50:27

Sandra

Yes. And if it's not, then you give it instructions. You say, “That wasn't great. Try another way and make it shorter/longer/sound nicer/sound more price/sound more chilled out.” Whatever you need. You can iterate based on what you already get but yeah, just [give the model] examples of the ideal outcome and as much context as possible.

# Sandra's 7-day LLM experiment

51:01

Alexey

**We already talked about one of your videos, where you made a video about GPT without having access to GPT. That was a very interesting story. I have also heard that you have another YouTube video, where you integrated LLMs into your life for seven days. Can you tell us more about that? What was the experiment about?**

51:27

Sandra

Yes, of course. I'm always extremely embarrassed when I talk about why I started doing something that I did. But, long story short, I have been in the LLM space for three years now. I think it's only decent that I use a certain amount of large language model-based tools and I don't do that. I'm either too lazy to try something out, too impatient to tweak and improve it so that it works for my particular setup, or I don't know, I'm too attached to my existing process and to my own writing. So I decided to just challenge that and force myself to use different types of LLM-based applications, and image-based applications as well, to see whether they will help me or not.

It was surprisingly difficult to stay consistent. I felt almost like… It comes back to what we discussed before that, unless a person (in this case, me) has a very good reason to change something because it's currently painful, and you want it to be smoother, [the person] doesn't do that. You use tools for pragmatic purposes. You just want the effect, and you want to get there as fast as possible – you don't care about proving a point. I was kind of trying to prove a point, but also trying to see whether it would relieve the pain of certain aspects of my work. And I ended up staying with a bunch of them. For example, I tried this email assistant… Actually, I would say, it's just a general digital communication assistant from HyperWrite. And I am still using it to help me generate my emails, or generate a draft that I then improve. I'm also using it for creating my social media posts, for example. I did a bunch of experiments when it comes to creating a YouTube video itself. For example, asking a model to come up with an outline for a video, then coming up with a title – you went through this process yourself. It can be painful and you're not necessarily happy with the outcomes. Then coming up with a thumbnail for a video. Then coming up with a social media post about the video.

There's so much content that you can now generate with the help of a language model, but it will not necessarily satisfy you (it will not necessarily hit the results that you're looking for). And so a bunch of things like that where I would just never replace it that way. But then there were… Actually, there was just one application that stuck with me. [chuckles] One application that stuck with me and the other, well… I used ChatGPT in that example, but now I'm not using GPT. I use Coral from Cohere, but I use it daily as well for different types of things. It was useful. It was useful to force myself to experiment with this. And it gave me some food for thought when it comes to how much of my process I want to automate, how much of that ends up being fun for me, or ends up just being annoying for me because I feel like a teacher that has to go over this outline that is very subpar to what I'm looking for [chuckles] and teach them, “No, this is not how we're gonna do it. We're gonna do it the other way.” These are the types of things that do not give me joy.

I like to create, so I don't like to automate everything. But for social media, oh man – so helpful. Emails – so helpful. I hate writing emails. The things that annoy you the most, those are the best places to apply the language models. I see the highest chance of adoption there, definitely.

56:03

Alexey

**Are the tools that integrate into Gmail and help write emails?**

56:08

Sandra

Yeah, HyperWrite is one of them. It's a Google Chrome extension, that once you turn it on, whenever you open your Gmail account, and you start writing a draft, it's gonna give you an example email response, or example initial email – whatever you need, basically, based off of your existing communication, or what you give it, or the keyword that you give it.

56:44

Alexey

**Maybe I should try one. I need to answer quite a few emails, and sometimes I procrastinate and I end up at the end of the week with like 50 emails that I need to answer. Then I'm like, “Ugh. It's Sunday, and I need to answer these emails.” So I sit down and answer them and spend a lot of time [doing it].**

57:03

Sandra

Yeah. Also, sometimes, it's like… When you write an email, you need to think through the particular task that you're trying to solve and ask somebody for some [particular] thing. But sometimes you've done this process already so writing an email is only just re-iterating (putting it in other words). For me, it's frustrating, because the fun part (the thinking part) is done already. But I can take the thinking part, give it to the model, and then it will create an email for me as a time-saver.

57:36

Alexey

**Yeah. I just realized that we should actually be wrapping up and there are so many questions that I wanted to ask you but did not have a chance to. One of the things is that you have a very interesting profile (career). You did not mention that you have a degree in liberal arts, but then you ended up being in AI and doing LLMs. But maybe this is something we can talk about in the future. But for now, we should be wrapping up.**

# Sandra's LLM content recommendations

58:04

Alexey

**Maybe the last thing I want to ask you before we finish is – I know you have a book. But this book is like two years old, right? So maybe some things are a bit outdated. Is there any other resource that you would recommend for people who want to learn more about LLMs? Maybe something from Cohere?**

58:24

Sandra

Yes, absolutely. We have recently launched something called LLM University, which is basically a free course for folks that are trying to understand large language model foundations both from the theory side of things, but also from the programmatic side of things of building stuff with it. So you have two components there. It's a very nice primer for whatever you're going to do with LLMs in the future. I definitely recommend that. My colleague from Cohere, Jay Alammar, is working on a new handbook for large language models, so follow that process. We also have a bunch of YouTube videos that are great. Our blog is also excellent when it comes to diving deeper into notions like embeddings, or semantic similarity or, how to build a chatbot, how to connect to a vector database, what kind of frameworks are hot out there. I really recommend our blog to get more tuned into LLMs right now.

59:39

Alexey

**Please send us a link, and then we will include this link in the description. If you have time, maybe you can pick a few interesting articles that you liked the most, and we can also include the link in the description. Okay. With that, we should be wrapping up [audio cut off]. I think my connection is not stable. Unfortunately, I was not able to… I don't know if there are questions that some of you from the audience asked, because the moment I try to open YouTube, my internet connection freezes. So I was not able to check if there were any questions. Sorry about that.**

60:22

Sandra

I'll try to answer them – if I can go there and answer questions, I can check that.

# Finding Sandra online

60:30

Alexey

**Or maybe, if somebody wants to ask you a question, should they write you through LinkedIn? Or is there any other way you prefer to be connected?**

60:39

Sandra

I have my YouTube channel, so please drop a comment. Please drop a comment on Twitter or LinkedIn or wherever. I'm everywhere.

60:51

Alexey

**It's X now, right? Not Twitter.**

60:55

Sandra

Sorry? Oh yes, yes. It's so confusing. I am not following anymore. Things are changing so fast. [chuckles]

61:04

Alexey

**Exactly. [chuckles] Okay, yeah. Thanks, Sandra, for joining us today. And thanks, everyone, for joining us today, too. I don't know – the next episode will probably not be about LLMs, but let's see. [chuckles] Maybe it actually will be.**

61:24

Sandra

I hope there will be more episodes, anyway. Maybe not the next one, but in the future. But it was really, really fun to talk to you. Thanks for having me and have a great rest of your day/evening/morning.