Markus’ Academy with Kevin Bryan
05.11.2023
Unedited Transcript

1
00:05:50.720 --> 00:06:00.690
Markus Brunnermeier: Welcome back everybody for another Webinar at Princeton, for everyone worldwide. They're very happy to have Kevin Brian with us from the University of Toronto. Hi, Kevin. Good to have you.

2
00:06:02.120 --> 00:06:03.670
Kevin Bryan: hey? How are you, Marcus?

3
00:06:03.760 --> 00:06:11.060
Markus Brunnermeier: Good morning, Kevin. Good to see you. Kevin will talk today about a user sky to Gpt. And Llms for economic research.

4
00:06:11.230 --> 00:06:23.670
Markus Brunnermeier: Kevin was virtually at Vincent already some weeks ago, and he will tell us now how we can use Gpt. And Jet gpt much more widely for our economic research and outline various aspects to it.

5
00:06:24.100 --> 00:06:35.310
Markus Brunnermeier: And before we start to give the floor to Kevin we go to our usual poll questions, and, thanks to all of you for answering the poll questions. Here are you? Answers, and Kevin will then illuminate us

6
00:06:35.420 --> 00:06:40.170
Markus Brunnermeier: what the correct answers are down the road. First question was.

7
00:06:40.190 --> 00:06:54.900
Markus Brunnermeier: Does open eyes, Api's. Do the the cost to a category? 10,000 sentences from my historical business census, for example, and you want to discuss whether there is some adoption of new ideas or not. How much will this cost?

8
00:06:55.040 --> 00:07:01.830
Markus Brunnermeier: So you will it to $20, $200 $1,000 or $5,000, and the honors were 81%,

9
00:07:01.910 --> 00:07:11.510
Markus Brunnermeier: 21%, 6, and 1%. I give the answer away for this one the maturity is right with 81%.

10

00:07:12.920 --> 00:07:28.280
Markus Brunnermeier: The second question was, Does jeopardy improve collectivity for humans on on which tasks will it help the weaker users to catch up with the stronger ones. But will it help the stronger users to pull away from the weaker ones? So the high skilled workers will benefit more.

11
00:07:28.860 --> 00:07:31.070
Markus Brunnermeier: How does it depend on the tasks

12
00:07:31.290 --> 00:07:38.110
Markus Brunnermeier: and what it will while it work? Finally, the very autonomously. So it that doesn't depend much on human skills at all.

13
00:07:38.180 --> 00:07:45.300
Markus Brunnermeier: and the answers were 1716, 62. So it depends on the task was a majority.

14
00:07:45.510 --> 00:08:00.400
Markus Brunnermeier: and 5% said it works very. The third question was about hallucinations, and we all know from Gpt that they are Halloween nations, and I guess Kevin will tell us how to control allucinations with a more

15
00:08:00.680 --> 00:08:02.810
Markus Brunnermeier: sophisticated prompting.

16
00:08:02.900 --> 00:08:15.790
Markus Brunnermeier: Is it very common? But it can be fixed to 99 that's you know 48% that it is impossible to fix it's very common, but impossible to fix it's 23%.

17
00:08:16.340 --> 00:08:27.080
Markus Brunnermeier: Is it very something very specific to open a eyes? A Gpt. Or is it generally for L. A. Models also, for you know, Google's and so forth. That's what. 17%.

18
00:08:27.450 --> 00:08:30.400
Markus Brunnermeier: Why is it like? An I thing and 12%

19
00:08:30.450 --> 00:08:46.580
Markus Brunnermeier: what it so yeah, phenomenon? And finally, the last question is, what is Llms or the large language models primary advantage to research, intensive startups? Is it helping them to predict the market trends and consumer behavior. That's 23% thought. This Why.

20
00:08:46.830 --> 00:08:54.280
Markus Brunnermeier: it is, allows the business to fast and more efficient decision. Making 42% thought this way.

21
00:08:54.870 --> 00:09:00.870
Markus Brunnermeier: and it eliminates the need for human employees and cuts down on costs 13%,

22
00:09:01.500 --> 00:09:02.400
Markus Brunnermeier: and

23
00:09:02.870 --> 00:09:17.650
Markus Brunnermeier: it can be easily integrated without disruptions in the business processes. That's 22% thought there might be some growth to all of them. So but we will figure out from Kevin what's the most prominent correct Answer. But again 2342, 13, and 22.

24
00:09:18.530 --> 00:09:25.250
Markus Brunnermeier: Finally, I would like to allude to another a paper which by Anton Koreank the he wrote that any

25
00:09:25.400 --> 00:09:31.640
Markus Brunnermeier: outland a little bit. How does jet Gpt and other large language models help us in research and economic research?

26
00:09:31.670 --> 00:09:50.670
Markus Brunnermeier: It's in different stages of economic research. One is the ideas by creating the idea. Brainstorming with Chat Bt: You can have some communication with Jet Gbt: Get some more background research going as well, but it also helps a lot in writing and synthesizing some text with you, using the abstract from 200 words.

27
00:09:50.670 --> 00:09:56.340
Markus Brunnermeier: We're reforming the references also just the signing tweets after what's promot your research.

28
00:09:56.500 --> 00:10:11.360
Markus Brunnermeier: but it can also help your encoding, especially in debugging and explaining things and translating from not one software language to another software, language data analysis. Scanning was your and connecting, getting some texture analysis out of text.

29
00:10:11.390 --> 00:10:14.360

Markus Brunnermeier: so extracting some sentiment. But there are many, many things that

30
00:10:14.500 --> 00:10:20.040
Markus Brunnermeier: many, many papers coming up, and I think we will learn today more tips and ticks how to do that.

31
00:10:20.130 --> 00:10:22.490
Markus Brunnermeier: So with this I will pass on the floor to Kevin.

32
00:10:22.600 --> 00:10:28.800
Markus Brunnermeier: just to give you a little bit of appetizer. And now comes the main course and the desert

33
00:10:29.010 --> 00:10:30.010
from Kevin.

34
00:10:30.270 --> 00:10:32.000
Markus Brunnermeier: Thanks again, Kevin, for doing it

35
00:10:32.370 --> 00:10:39.520
Kevin Bryan: great. Thank you very much, Marcus, for the introduction I should mention for the folks who know Anton Cornick's paper.

36
00:10:39.580 --> 00:10:56.880
Kevin Bryan: It's well worth reading. I'm. Going to go into a little more detail on practical use of of Lms, and how to avoid some of the problems that people sometimes face with them. But he gives a a kind of much wider overview of of tasks that that go well beyond what i'm going to discuss today. So I recommend you. Take a look at that

37
00:10:57.030 --> 00:11:01.240
Kevin Bryan: as Well, I also should mention on our poll.

38
00:11:01.350 --> 00:11:08.660
Kevin Bryan: So of course, the Modal, the modal response was correct on all 4 questions. The fun thing was the last question. We actually have a

39
00:11:08.690 --> 00:11:25.070
Kevin Bryan: a piece of software that I developed that we're using up here, Toronto that can automatically generate Question banks for your classes. And I happen to have a class where you have a bunch of AI related documents, but it's trained on. And so that last question was a chat. Gbts question for you, not mine.

40
00:11:25.140 --> 00:11:28.120
Kevin Bryan: Okay. I still agree with the answer, though.

41
00:11:28.850 --> 00:11:36.860
Kevin Bryan: All right, so let's let's get started. I'm gonna open with just the very basics of of Lms.

42
00:11:37.380 --> 00:11:41.560
Kevin Bryan: By then it's just to make sure we're on the same page. Then we're going to talk about practicality. So

43
00:11:42.150 --> 00:11:52.740
Kevin Bryan: these large language models essentially, I mean, they're really new. So there's a paper in 2,017. A computer science paper named Detention is all you need, introduces this architecture called the transformer

44
00:11:53.980 --> 00:12:13.640
Kevin Bryan: that it basically underlines all these large language models. This came from a team at Google, including actually an undergraduate here at the University of Toronto, who was a summer intern at Deepmind as one of the authors on the paper he's now started his own company. They raised like a 1 billiondollars company called Co. Here. So everyone's really interested in this technology, this this idea.

45
00:12:13.640 --> 00:12:18.140
Kevin Bryan: and it's simple enough. You can just read the paper and understand the kind of fundamental insight that they

46
00:12:18.530 --> 00:12:27.390
Kevin Bryan: this is applied immediately to text and images with like so so results. You were a little bit impressed, but you weren't like, blown away

47
00:12:27.510 --> 00:12:40.520
Kevin Bryan: until you start seeing Gpt. 3 in 2,021, and then these image generators, like dally and stable diffusion in mid 2,022, especially the latter, really impressed people. It just seemed like impossible

48
00:12:40.630 --> 00:12:42.210
Kevin Bryan: for a

49
00:12:42.310 --> 00:12:53.340

Kevin Bryan: for AI to say paint a picture that's indistinguishable from, say, a painting by Van Gogh, but that's a well within the capability of these models to name.

50
00:12:53.430 --> 00:12:59.930
Kevin Bryan: So we're already pretty impressed, and then November the thirtieth 2,022 chat gpt

51
00:13:00.020 --> 00:13:02.790
Kevin Bryan: that is released. And you know

52
00:13:03.000 --> 00:13:13.440
Kevin Bryan: the entrepreneurship program I work with here at the University of Toronto. Now we've had a relationship with open AI for years. So so we've had access to like earlier versions of Gpt.

53
00:13:13.450 --> 00:13:22.380
Kevin Bryan: We've had companies coming through who've been working with large language models literally since 2,017, when the transformers invented, and nonetheless we were still amazed, like we could not believe

54
00:13:22.620 --> 00:13:39.970
Kevin Bryan: the step forward that we saw. November thirtieth. We're running back and forth down the hallway. Can you believe what what this can do? And since November thirtieth, literally tens of millions of usd and market value been created funding companies in this area in this area, even though

55
00:13:39.970 --> 00:13:47.300
Kevin Bryan: the Vc. Market's quite weak at the moment. So companies like go here and and open a. I get a huge investment for Microsoft, Jasper, Anthropic

56
00:13:47.310 --> 00:13:50.100
Kevin Bryan: stability, mid journey, and so on. Okay.

57
00:13:50.520 --> 00:14:03.580
Kevin Bryan: incredibly, incredibly hot area. Everyone in the computer science world is aware of what's going on here, and if you're in our world in economics, you need to understand why they're so excited. Okay. So what can these these lms do.

58
00:14:04.050 --> 00:14:06.110
Kevin Bryan: We've had a few academic papers.

59
00:14:06.130 --> 00:14:24.890
Kevin Bryan: Actually, most of the academic research you see on lms is Pre-chat Tpt, which makes sense, and then it's only 5 months old. So these are like kind of actually understanding how how amazing the technology is. But to give you 2 examples that might motivate you to think I really need to understand this technology on the right.

60
00:14:24.950 --> 00:14:31.050
Kevin Bryan: I I believe this is a through Github's copilot. They gave a bunch of people

61
00:14:31.390 --> 00:14:40.670
Kevin Bryan: relatively straightforward coding tasks and give some of them access to copilot, which is basically a a large language model that helps you interpret debug and write code.

62
00:14:40.770 --> 00:14:48.310
Kevin Bryan: And the time it took to perform the task, for the average coder fell from 2 h 41 min to 1 h 11. So, like, you know, more than half

63
00:14:48.450 --> 00:14:51.720
Kevin Bryan: the time to form little these people's jobs.

64
00:14:51.950 --> 00:14:57.910
Kevin Bryan: and this is a randomized trial. On the left hand we have a paper that looked at

65
00:14:57.930 --> 00:14:59.810
Kevin Bryan: human sales, people

66
00:14:59.870 --> 00:15:16.180
Kevin Bryan: who did or did not have access to essentially an AI that would suggest using a large language model phrasing on their sales calls. And again, you see, just like the kind of productivity bumps that yeah, the customer person rate from 2.8 to 4.5 like this.

67
00:15:16.180 --> 00:15:21.370
Kevin Bryan: This Basically, takes your kind of average salesperson makes them your ninetieth percentile salesperson overnight.

68
00:15:21.620 --> 00:15:26.360
Kevin Bryan: So we see these kind of resulted over and over and over again in industry

69
00:15:26.870 --> 00:15:33.710
Kevin Bryan: for my perspective, you know, having worked, you know, I've I've I've helped run an AI based entrepreneurship program since 2,015.

70
00:15:34.840 --> 00:15:38.050
Kevin Bryan: I've not seen a technological breakthrough

71
00:15:38.140 --> 00:15:44.610
Kevin Bryan: promising as large line as models, I think, since the Internet would probably be the last, the last breakthrough of this.

72
00:15:44.730 --> 00:15:51.440
Kevin Bryan: So you really need to understand and and understand how you can use technology and and what it can do for you as an economist. All right.

73
00:15:51.680 --> 00:15:57.320
Kevin Bryan: So how do these things work? You sometimes hear large language models called stochastic parents.

74
00:15:57.560 --> 00:16:01.880
Kevin Bryan: That's true. but it's not negative. So, for example.

75
00:16:02.400 --> 00:16:08.140
Kevin Bryan: if I were to ask you, tell me what comes next, you know if I save the sun.

76
00:16:08.190 --> 00:16:23.760
Kevin Bryan: And I ask Marcus, you might say something like this: sun rises. The sun is yellow something like this. Okay, the sun shines. It's a beautiful day here here in Boston, where I am right now, and the sun is shining out the way you can see it behind me. It looks like a cathedral.

77
00:16:23.940 --> 00:16:28.750
Kevin Bryan: So you know, we've got some distribution of words that follow

78
00:16:28.880 --> 00:16:34.330
Kevin Bryan: the in all of the written language we've ever seen. Okay, so imagine that

79
00:16:34.630 --> 00:16:36.360
Kevin Bryan: a prior distribution.

80
00:16:36.960 --> 00:16:53.690
Kevin Bryan: And now think what if I want to control writing so that it just doesn't pull like the kind of most common word which is probably is to follow the sun. What if I wanted to do something like, like, say, change it to French? Let's so lay. Now you know the next word to be

81
00:16:53.850 --> 00:16:57.260
Kevin Bryan: a not is right. So it'd be whatever is in French.

82
00:16:57.350 --> 00:16:58.180
Kevin Bryan: And

83
00:16:58.760 --> 00:17:03.190
Kevin Bryan: the idea basically of everything you're going to do with an Lm.

84
00:17:03.390 --> 00:17:18.750
Kevin Bryan: Is, you're going to try to adjust the posterior distribution of words that will follow words that already exist to get the response that you're trying to get out of the model. Okay, now, it seems crazy.

85
00:17:18.980 --> 00:17:26.880
Kevin Bryan: I in that. It seems crazy. Then all you have to do, and this is really all the underlines most lens. It's not that much more complicated than this.

86
00:17:26.910 --> 00:17:28.089
Kevin Bryan: We're going to take

87
00:17:29.250 --> 00:17:31.510
Kevin Bryan: a sense of the entire Internet Okay.

88
00:17:31.530 --> 00:17:38.990
Kevin Bryan: all the words we're going to use this to predict the distribution of the next word from the words that preceded it. We're going to add a little bit of noise.

89
00:17:39.080 --> 00:17:41.070
Kevin Bryan: so it's not totally deterministic.

90
00:17:42.050 --> 00:17:49.920

Kevin Bryan: And this trick is is actually quite successful. It might not be that surprising. Okay. So let's imagine.

91
00:17:50.650 --> 00:17:57.950
Kevin Bryan: I give you the following words, and I said, Tell me whether this sentence is discussing foreign Affairs or the economy, Colon.

92
00:17:58.100 --> 00:18:03.520
Kevin Bryan: You know the inflation rate rose to 3.2 yesterday, said the Fed

93
00:18:03.960 --> 00:18:07.060
Kevin Bryan: like. Answer. Colon.

94
00:18:08.710 --> 00:18:15.950
Kevin Bryan: you would be able to say economics right? And this sentence has never appeared anywhere on the Internet. I'm quite sure

95
00:18:16.150 --> 00:18:18.670
Kevin Bryan: it's never never been written by man ever

96
00:18:18.810 --> 00:18:23.360
Kevin Bryan: and yet, like there's so much context here that, like

97
00:18:23.750 --> 00:18:24.800
Kevin Bryan: it would be

98
00:18:26.150 --> 00:18:27.200
Kevin Bryan: it would be.

99
00:18:27.320 --> 00:18:40.180
Kevin Bryan: It's pretty easy for a model that can understand the structure of sentences implicitly by having foreign prior distribution, based on all the words on the Internet. To know the next word better be foreign affairs or the economy, 250;

100
00:18:40.220 --> 00:18:45.410
Kevin Bryan: and to understand that the economy is more likely given some of the terms that we've seen here in the way there were.

101
00:18:46.010 --> 00:18:51.970
Kevin Bryan: So that's really all that's going on ends. And we're just gonna apply this to more and more complicated tasks. Now.

102
00:18:53.460 --> 00:19:07.570
Kevin Bryan: what if I want to control it. In the same way I can control. Say, like a regression, I run a regression with a given set of data. It's going to give you the same result every time. Controlling an Lm. Is is very, very difficult.

103
00:19:07.590 --> 00:19:26.850
Kevin Bryan: is is almost more of an art than a science at this point. But we do know. Once we understand what's going on under the head of an Lm. And from some research that's been done by computer scientists, we actually can control these things pretty well. I'll show you some tricks that we'll get the Lms. To do what you want them to do that look very, very different

104
00:19:26.850 --> 00:19:37.540
Kevin Bryan: from say, just going to chat gpt on the Internet and typing something in and looking at the answer. So that's way, way way, far from the state of the art for the kind of task you need to do as an economist researcher.

105
00:19:37.940 --> 00:19:50.910
Kevin Bryan: The third thing that's that's important to fix is hallucinations or data back like, Why do we make up facts? Well. like literally, all we're doing is predicting the next word on the basis of the words that have come before.

106
00:19:51.000 --> 00:20:03.030
Kevin Bryan: We're not like checking. What are those words are true? We're not saying what it every basis. And so if I said something like, you know, when I was reading Charles Dickens novel.

107
00:20:03.990 --> 00:20:05.000
Kevin Bryan: Dot.

108
00:20:05.140 --> 00:20:12.350
Kevin Bryan: Probably it's gonna say a real novel. But you know, like there are many people who've written like fan fiction that

109
00:20:13.090 --> 00:20:31.720
Kevin Bryan: are related to Charles Dickens. There are many kind of translations of the novels that we translate it back into English. Don't. Look the same. It's not that surprising. You might hallucinate, given how Lms work. But this is actually quite fixable, I would say, in in kind of the used cases I have. I just don't worry about hallucination. I'm able to fix it so i'll show you how to do that.

110
00:20:31.820 --> 00:20:33.720
Kevin Bryan: The fourth thing to understand.

111
00:20:33.850 --> 00:20:36.050
Kevin Bryan: and the rate of improvement is so quick.

112
00:20:36.190 --> 00:20:40.590
Kevin Bryan: So Chat Gbp: 5 months and 10 days ago came out.

113
00:20:40.940 --> 00:20:46.310
Kevin Bryan: Gbt: 4 came out just over a month ago. Okay. So like

114
00:20:46.630 --> 00:20:48.140
Kevin Bryan: everything i'm seeing

115
00:20:48.150 --> 00:21:05.060
Kevin Bryan: is subject to the kind of revision that you should expect from a technology that's moving that quickly. At the end of the talk i'll give you some examples of things that will definitely be fixed Soon you'll be able to do with an Lm. And some things you definitely won't be able to do, and but in between those polls.

116
00:21:05.060 --> 00:21:09.470
Kevin Bryan: you know. I'll be surprised as you in 2,024 was possible.

117
00:21:09.710 --> 00:21:19.960
Markus Brunnermeier: You think kevin that our application process is way too slow to take advantage of that, because my third division of the paper Yeah, the technology will be totally outdated.

118
00:21:20.380 --> 00:21:27.280
Kevin Bryan: Yeah, of course, you know that luckily for economists. We have working papers that who cares about publications, anyway? Right?

119
00:21:27.460 --> 00:21:30.830
Kevin Bryan: And so, once you have 10 years, put your working paper on your website and and leave it be

120
00:21:33.050 --> 00:21:38.000
Kevin Bryan: was, I believe it was Hersani right who were, who eventually refused to

121
00:21:38.320 --> 00:21:48.790
Kevin Bryan: to submit any paper where you got referee reports. So you see, his papers published in the most random journals. Because he was like, Why should I? I already wrote the paper You want to revise it. Write your own. Follow up.

122
00:21:48.900 --> 00:21:59.390
Kevin Bryan: That's the way we'll have to go. Okay. And last thing I understand when I say Rodchat, Gbt: as far from the state of the art. What I mean is generally you're going to want to use the Api

123
00:21:59.910 --> 00:22:10.550
Kevin Bryan: either for open AI or for their competitors and some code. So that's gonna give you just much more control of what's going to come out of of an Lm. I'll show you how to do that.

124
00:22:11.110 --> 00:22:11.830
Kevin Bryan: All right.

125
00:22:12.290 --> 00:22:16.570
Kevin Bryan: The main research tasks that we'll talk about today that Lms are really useful

126
00:22:16.650 --> 00:22:18.300
Kevin Bryan: the first one

127
00:22:18.780 --> 00:22:21.970
Kevin Bryan: and any recognition in unstructured data.

128
00:22:22.280 --> 00:22:25.330
Kevin Bryan: So all the time we're trying to identify

129
00:22:25.350 --> 00:22:32.520
Kevin Bryan: something like. I have what I was talking to, Marcus. You have all these old scanned documents. I want to know.

130
00:22:32.920 --> 00:22:34.550
Kevin Bryan: When did mergers happen

131
00:22:35.170 --> 00:22:48.750
Kevin Bryan: in early twentieth century? Germany, and you give me like thousands of pages of text. I could read it and figure it out. It's not in any structure that's easy to deal with. So it's quite difficult for us

to. We can't like run a regression. There's nothing we can do until we clean that data up

132
00:22:48.830 --> 00:22:54.270
Kevin Bryan: for structured data where the form is well known. Everything's like in numbers.

133
00:22:54.580 --> 00:23:03.880
Kevin Bryan: There's probably better tools and Lms. Honestly for most of those tasks, but unstructured data, the kind of thing. Where, like you're going to hire some Ra. And have them to stare at stuff

134
00:23:03.930 --> 00:23:12.210
Kevin Bryan: to try to figure out what's going on in this data. Those type of tasks are really really replaceable by Lms. I'll show you a couple of examples.

135
00:23:12.420 --> 00:23:29.120
Kevin Bryan: Number 2, If you're not already doing this, i'm just telling you you're making a mistake to to program anything is actually much, much faster using an Lm. In my case for sure, like the the doubling of of code, you can write in any given amount of time is probably an understatement.

136
00:23:29.140 --> 00:23:37.770
Kevin Bryan: and this is true of not just like writing some python or some art, but also true for creating like a graphics for your papers. For example, you do a lot worker.

137
00:23:38.070 --> 00:23:50.370
Kevin Bryan: The third use case is a you know. We had spell check. Then we had some spell check that can handle some grammar. And now you're like Gmail. We'll correct you if you like. You know you get 2 words backwards.

138
00:23:50.480 --> 00:24:08.180
Kevin Bryan: The kind of like improvement in your writing and get out of an Lm. Is just way beyond that. So if you write clunky sentences, you want to rewrite it less clunky. You can just say, rewrite this in a less clunky way. It'll do it, which is super useful, especially for people who are foreign language speakers that are writing, not in their need of time.

139
00:24:08.350 --> 00:24:27.180
Kevin Bryan: The fourth thing that I use this for all the time is to summarize a literature. So you know, we a lot of us have a little folder on our computer with a bunch of documents all these Pbs. We read at some

point, and then we're like I'm trying to remember. What was the paper that did X. What exactly did they find? And we scan through 20 pages and 20 pages and 20 pages.

140
00:24:27.180 --> 00:24:33.110
Kevin Bryan: That problem is completely solved. I never do that anymore. I'll show you some code that will allow you basically to

141
00:24:33.170 --> 00:24:38.700
Kevin Bryan: to query and natural language, like your entire folder of papers, and just figuring out the answer really quickly

142
00:24:38.980 --> 00:24:44.790
Kevin Bryan: what it can't do right now. But it's coming soon is a correct math

143
00:24:44.820 --> 00:24:52.900
Kevin Bryan: and and derivations. It's just not not the for this task. Sometimes it gets it right. Usually it gets it wrong. I'll discuss why.

144
00:24:53.140 --> 00:24:54.990
Kevin Bryan: And and more importantly.

145
00:24:55.220 --> 00:25:01.290
Kevin Bryan: when we think through. If you know this idea in in management called architectural Disruption, that comes from a Rebecca Henderson

146
00:25:01.450 --> 00:25:04.760
Kevin Bryan: basically there, when you have a big new innovation like this.

147
00:25:04.800 --> 00:25:21.220
Kevin Bryan: Replacing tasks we already do with kind of a cheaper, faster way is the first thing we wind up. Doing. But the really big long-term benefit of Lms is going to be allowing us to do tasks that I can't even think of, because we don't do them right now, and the reason we don't do them is because it was just not really feasible.

148
00:25:21.220 --> 00:25:33.640
Kevin Bryan: Given the cost in the assets of an Lm. So what those tests are, I have no idea. But i'm sure. Over the next few years we're going to see amazing use cases. So don't think this is the the limits of Lms like any technology. We're we're very early.

149
00:25:34.280 --> 00:25:34.950
Kevin Bryan: Okay.

150
00:25:36.230 --> 00:25:39.900
Kevin Bryan: All right. Let's go to how I can control an Lm.

151
00:25:39.940 --> 00:25:57.530
Kevin Bryan: So there's essentially 4 ways to control the output of an Lm. And remember, all these are doing the same thing. I'm going to take that distribution of words that are predicted to follow up for words I've already written, and just ship that posterior to the kind of words that I care about. So, for example.

152
00:25:58.650 --> 00:26:02.090
Kevin Bryan: let me hear a story. Tell me about the weather yesterday.

153
00:26:02.370 --> 00:26:04.410
Kevin Bryan: The sun dot dot

154
00:26:05.120 --> 00:26:10.830
Kevin Bryan: so previously we were going to get. The sun is the sunshine. The sun rises. Now

155
00:26:11.130 --> 00:26:14.350
Kevin Bryan: you're going to get the sun what the Sun

156
00:26:14.810 --> 00:26:25.600
Kevin Bryan: Stone, you know. The sun rose right? And why? Because following things like the weather yesterday, like in the English language

157
00:26:25.980 --> 00:26:32.870
Kevin Bryan: words that follow that kind of rough part of the think of, like the linguistic space tend to be in the past dense.

158
00:26:32.920 --> 00:26:41.000
Kevin Bryan: So this just starting this little prompt, and giving this prompt to an Lm. Will be sufficient to ship the exterior. So you're going to get something like the sun was here

159
00:26:41.150 --> 00:26:44.500
Kevin Bryan: Now it's not a lot of control, but it's really really easy.

160

00:26:45.520 --> 00:26:46.650
Kevin Bryan: Second method.

161
00:26:46.800 --> 00:26:50.500
Kevin Bryan: We sometimes call this like 0 shot context.

162
00:26:50.750 --> 00:27:01.080
Kevin Bryan: and the idea here is we're going to give a bunch of context to the large language model about the kind of things we're interested in. So, for example.

163
00:27:02.200 --> 00:27:07.540
Kevin Bryan: I tell the the context of our conversation are stories written by Ernest Hemingway.

164
00:27:08.150 --> 00:27:15.000
Kevin Bryan: and then I say, the Sun, now an Lm. To complete this is probably going to say also.

165
00:27:15.070 --> 00:27:19.200
Kevin Bryan: which is where the sun rises, all sun also rises, is the Hemingway book.

166
00:27:19.230 --> 00:27:23.500
Kevin Bryan: and there's just no way it's going to choose the word. Also

167
00:27:23.520 --> 00:27:36.490
Kevin Bryan: it's a very unusual way to follow up the sun unless you're literally talking about books by Hemingway. But this context is enough to shift the posterior so far that that the Ln. Is going to say also in this context.

168
00:27:36.820 --> 00:27:44.420
Kevin Bryan: Now we can give it even more context. Sometimes we call these fu shot prompting.

169
00:27:44.440 --> 00:27:53.350
Kevin Bryan: So the idea here is whatever I want to think the Lm. To do. I'm going to give it a bunch of examples of how I would like it to have responded. So, for example, 150,

170
00:27:53.610 --> 00:28:01.560

Kevin Bryan: you can say something like example or user the Moon answer orbits 100,000 miles from Earth. These numbers are totally wrong. By the way.

171
00:28:02.420 --> 00:28:12.040
Kevin Bryan: I'm. I'm. Not Gpt. Nor am I in a stronger example. Mars answer orbits. 50 million miles from an example. Pluto answer orbits a 1 billionmiles from Earth.

172
00:28:13.570 --> 00:28:15.490
Kevin Bryan: User the sun

173
00:28:15.980 --> 00:28:16.980
Kevin Bryan: Answer.

174
00:28:18.350 --> 00:28:19.950
Kevin Bryan: You give it this context.

175
00:28:20.020 --> 00:28:23.550
Kevin Bryan: It is going to give you that the sun orbits 93 million miles from her.

176
00:28:23.930 --> 00:28:31.950
Kevin Bryan: Now again, that's a fairly unusual way to follow up to worth the sun. But with this kind of few shot context, it's just.

177
00:28:32.090 --> 00:28:48.410
Kevin Bryan: I. I am 100% certain in Lm: Any decent Lm. Is going to follow up the sun with exactly the sentence in exactly this way, if you give it this context. So this is a really common thing like an industry. These kind of a few few shot context

178
00:28:48.410 --> 00:28:52.330
Kevin Bryan: prompting is, is it's probably actually the most common thing industry does to control

179
00:28:52.430 --> 00:28:54.140
Kevin Bryan: controlled their large language models.

180
00:28:55.270 --> 00:29:06.180
Kevin Bryan: Finally, method 4. Probably the one that seems the most natural is okay. I'm going to. I train these lms on like the entire Internet, say, or, like all of Wikipedia, what happens to you

181
00:29:07.220 --> 00:29:24.740
Kevin Bryan: if I want it to respond, say, in a more academic tone. Why, Don't, I just train the model on like academic documents right then, the way the model is going to speak that that posterior distribution of next words is going to look like the kind of you know, clunky academic text that we expect.

182
00:29:24.980 --> 00:29:33.400
Kevin Bryan: So, for example, I could I could recrain, or or just a train an additional way. There's some really nice ways to do this that have come out over the last few weeks

183
00:29:33.550 --> 00:29:35.820
Kevin Bryan: on Lm.

184
00:29:35.950 --> 00:29:36.550
Kevin Bryan: Up

185
00:29:37.170 --> 00:29:40.600
Kevin Bryan: on, say, a set of astronomy, academic papers.

186
00:29:40.660 --> 00:29:58.870
Kevin Bryan: And then, if I say the sun, you might get something like the sun emits radiation or something like this. Right? Those types of words are much more likely to appear in an academic paper about the sun than they are in all writing, or all human speech about the sun. So this type of fine tuning

187
00:29:59.450 --> 00:30:10.340
Kevin Bryan: it doesn't require you to give a whole bunch of context. So if I've got use cases that look really really different, but I would like to say, answer them like an academic tone. Fine-tuning can be quite useful

188
00:30:10.690 --> 00:30:15.130
Kevin Bryan: like 1 One thing. You often see, this in industry is like.

189
00:30:15.210 --> 00:30:19.320
Kevin Bryan: I would like you to answer, using, say, internal corporate documents.

190
00:30:20.080 --> 00:30:27.170
Kevin Bryan: They have. They use words in a very unusual way for the English language, but are very common inside a given corporation

191
00:30:27.960 --> 00:30:36.130
Kevin Bryan: so sure. Fine-tune the model I probably it's just probably too hard to give all the context I need on each question, and it's probably too expensive.

192
00:30:36.780 --> 00:30:42.490
Markus Brunnermeier: So these are 4 ways I can try to control. So, Kevin: yeah, can I? Just as one question by

193
00:30:42.610 --> 00:30:43.910
Markus Brunnermeier: epic, genie.

194
00:30:43.920 --> 00:30:55.810
Markus Brunnermeier: She would like to know whether you know you can ask Lms directly how to better prompt, more effectively the L. Is. It is still some, you know, Loop. Essentially.

195
00:30:56.430 --> 00:30:58.270
Kevin Bryan: Lems

196
00:30:58.550 --> 00:31:01.740
Kevin Bryan: do not understand your question.

197
00:31:01.900 --> 00:31:11.360
Kevin Bryan: Lms. Literally are predicting the next word on the basis of your context, your question and the following words: so it is not possible

198
00:31:11.450 --> 00:31:13.820
Kevin Bryan: for saying l them trained on

199
00:31:13.910 --> 00:31:26.120
Kevin Bryan: writing from 2,021 to understand what was done to program the model. Now we can give you a really convincing, sounding answer to your question, but it will not know the answer.

200
00:31:26.160 --> 00:31:29.570
Kevin Bryan: On the other hand, if, for example, I trained an Lm.

201
00:31:29.670 --> 00:31:40.030

Kevin Bryan: On a bunch of documents that we use to train that particular Lm. Then I asked the question that i'm sure can give you the answer, but I certainly go. Ask Gpt. How?

202
00:31:40.150 --> 00:31:46.690
Kevin Bryan: Ask Chat Gbt. What G. P. 4 does it literally doesn't know, because it was trained before Gb. 4 existed.

203
00:31:46.760 --> 00:31:48.920
Kevin Bryan: It doesn't know what gpt for is.

204
00:31:49.210 --> 00:31:58.420
Kevin Bryan: but it's going to make up something very convincing to you about what's happening under the head. So this is not a good way to use it now. 11 years asking for a hallucination in this case.

205
00:31:59.230 --> 00:32:01.500
Kevin Bryan: Okay. So for these methods of control.

206
00:32:01.900 --> 00:32:05.240
Kevin Bryan: the first method is like pure just change the prompt.

207
00:32:05.800 --> 00:32:10.990
Kevin Bryan: It's super cheap. It's often too imprecise for kind of research. Use cases.

208
00:32:11.450 --> 00:32:12.730
Kevin Bryan: Fine-tuning

209
00:32:12.780 --> 00:32:18.010
Kevin Bryan: generally works quite well, but it's it's incredibly expensive.

210
00:32:18.030 --> 00:32:24.210
Kevin Bryan: You would have a tough time. Fine-tuning a model for you know less than 4 figures, and and

211
00:32:24.400 --> 00:32:28.390
Kevin Bryan: you would get bad results if you tried to do it yourself, and at that that budget.

212
00:32:28.620 --> 00:32:37.350

Kevin Bryan: So these ones in the middle, these contextual prompts, especially if you shot contextual prompting, where a few shot means give a few examples that are kind of drawn for the distribution of stuff.

213
00:32:37.380 --> 00:32:45.830
Kevin Bryan: You would like the L. And take a look at that turns out to work really well as a method of control alongside some other tricks, and I like to show you

214
00:32:46.400 --> 00:32:47.180
Kevin Bryan: so.

215
00:32:47.590 --> 00:32:54.270
Kevin Bryan: Oh, just a quick note since we we talked about in our initial questions. How much does this cost?

216
00:32:54.530 --> 00:32:55.490
Kevin Bryan: So.

217
00:32:56.520 --> 00:33:02.470
Kevin Bryan: Gbt: You can think of? Think of Gpt as taking the entire Internet for all practical purposes that that's not that far off

218
00:33:02.650 --> 00:33:12.920
Kevin Bryan: the memory that it can hold on any given question, like when you're on Chat Gbp, and having a conversation, the memory being held is about 25 higher words at most.

219
00:33:13.270 --> 00:33:15.470
Kevin Bryan: You cannot pay at any price.

220
00:33:15.650 --> 00:33:24.080
Kevin Bryan: Open AI on the the chat, you key with it, called Gpt. 3.5. You cannot give it more information on a given question.

221
00:33:24.150 --> 00:33:35.450
Kevin Bryan: including all the context of prior questions. You get a conversation of more than 2,500 words. That's the most you can give it. Gpd: 4 has a bigger memory. But this memory constraint is really serious. So if you have, like 2

222
00:33:36.290 --> 00:33:47.870
Kevin Bryan: 10,000 pages of documents. You can't just give 10,000 pages of documents to an Lm. And ask them to read it. That's just not how they

work. They have very limited memory. So you need to use some tricks to be able to get around this.

223
00:33:47.950 --> 00:33:56.760
Kevin Bryan: But for the fine-tuning that the memory limit is also is it mostly for prompting of fine-tuning. I don't know there's no memory limit in the sense that

224
00:33:57.960 --> 00:34:12.070
Kevin Bryan: Think of the memory as what I'm using to shift this initial distribution to some posterior. Yes, the fine-tuning just changes that initial distribution, that i'm going to apply apply bas to so fine-tuning in some sense for like

225
00:34:14.380 --> 00:34:17.800
Kevin Bryan: allows me to have like a bigger memory for certain things

226
00:34:19.350 --> 00:34:30.070
Kevin Bryan: so like. For for example, imagine I had an ln there's only train on English documents, I could say, explain some German grammar and some German words, but i'm very quickly going to run out of memory.

227
00:34:30.139 --> 00:34:38.730
Kevin Bryan: or I could just retrain them but on-demand documents. And then then it's going to understand German. So for that use case obviously fine-tuning your only method.

228
00:34:38.980 --> 00:34:52.080
Kevin Bryan: The other problem other than memory is that generally these these systems you can do some stuff locally, but your the quality is much worse than than the the the kind of best models which are open. AI: basically so gpt

229
00:34:52.290 --> 00:34:58.110
Kevin Bryan: If you use open AI, you're paying for token. A token is about 3 quarters of a word.

230
00:34:58.200 --> 00:35:04.840
Kevin Bryan: So it's not expensive $1. It gets you 500,000 tokens

231
00:35:05.040 --> 00:35:08.050
Kevin Bryan: of of responses and questions.

232
00:35:09.310 --> 00:35:14.230

Kevin Bryan: But once you go up to Gpt 4, that's a 15 to 20 X increase in what you pay.

233
00:35:14.340 --> 00:35:18.450
Kevin Bryan: Once you start asking questions of your own fine-tune model through of an AI

234
00:35:18.800 --> 00:35:20.000
Kevin Bryan: it's even more expensive.

235
00:35:20.610 --> 00:35:30.850
Kevin Bryan: So you know a 20 X increase in your research budget. It better be a lot better for you to use that rather than just use some of the other prompting tricks I mentioned here today.

236
00:35:30.860 --> 00:35:35.680
Kevin Bryan: The other thing you can pay for is, if you don't already pay for Gpt. 4 in like the chat.

237
00:35:35.830 --> 00:35:41.740
Kevin Bryan: This you can't really use for research, because it limits you to 25 questions every 3 h, and you can't use an Api anyway.

238
00:35:41.750 --> 00:35:44.020
Kevin Bryan: but it's $20 a month.

239
00:35:44.070 --> 00:35:48.010
Kevin Bryan: I use it. When i'm coding, I would say every day.

240
00:35:48.350 --> 00:35:52.720
Kevin Bryan: So that's $20 very well spent, and you can just sign up for it and pay right now

241
00:35:53.250 --> 00:36:06.970
Markus Brunnermeier: it just explains. So if I use everybody uses the chat. So everybody is familiar with the chat. But the other method You say I did. I write a little Api program to log it on top of G. P. 4. Yeah, I'll show you exactly how the code works.

242
00:36:07.020 --> 00:36:15.180
Kevin Bryan: but the chat, the chat. You have very little control. There's a lot of parameters underlying these models that are just fixed in the chat

243
00:36:17.020 --> 00:36:22.180
Kevin Bryan: you use. If you code and say python, or whatever you want want to use to call their Api.

244
00:36:22.190 --> 00:36:26.850
Kevin Bryan: you have to pay for it at about, and $1 for 500,000 tokens.

245
00:36:28.220 --> 00:36:34.660
Kevin Bryan: It's not that expensive? My lifetime spending with with open AI is like 20 bucks, and I use it a lot.

246
00:36:36.280 --> 00:36:45.420
Kevin Bryan: But you do have to pay, so there's no like it's not like the kind of free Chat Tbt: you see on the Internet but it's really the only thing that's usable for the kind of research you want to do in some of these examples.

247
00:36:46.200 --> 00:36:47.580
Kevin Bryan: Okay, so.

248
00:36:47.800 --> 00:36:57.160
Kevin Bryan: And he didn't talk about the the Google bought, and all this you will come to that as well. So I won't. I won't come to it other than to say.

249
00:36:58.230 --> 00:37:02.110
Kevin Bryan: I know people at Deep Mine. Google has awesome internal stuff.

250
00:37:02.260 --> 00:37:12.050
Kevin Bryan: I don't know if you saw like we're talking in mid-may right now. Google Bar just demonstrated. I did a big demo yesterday and it's like embarrassingly far behind

251
00:37:13.170 --> 00:37:18.210
Kevin Bryan: what open. AI has released. And the same is true actually of what Meta, what Facebook has put out

252
00:37:18.290 --> 00:37:25.680
Kevin Bryan: with the only benefit being that their stuff got lead. So people are building open access Lms on top of it

253
00:37:25.850 --> 00:37:33.330

Kevin Bryan: right now. The only, like frontier Lm. Is open. AI's. Gpt. Everything else is way behind it.

254
00:37:33.350 --> 00:37:42.490
Kevin Bryan: and essentially any task you want to perform, that the anything you have access to, let's just say so, for all practices. Think of this talk as being about Gbt

255
00:37:42.500 --> 00:37:50.940
Kevin Bryan: in 2,025. Who knows? I'm trying to be kind of agnostic to the exact technology. But I can't imagine any use case I would use like barn, for, for example.

256
00:37:53.070 --> 00:37:56.210
Kevin Bryan: So let me show you in practice how I do some of these tests. So

257
00:37:56.430 --> 00:38:02.650
Kevin Bryan: this is from a a paper I was writing where we had to. I identify in patent text

258
00:38:02.840 --> 00:38:04.770
Kevin Bryan: why a given

259
00:38:04.840 --> 00:38:08.410
Kevin Bryan: scientific citation was used so in particular.

260
00:38:08.500 --> 00:38:14.110
Kevin Bryan: we are interested in. If you look at this paper by barack at all in the J. By L. Chem.

261
00:38:14.290 --> 00:38:28.800
Kevin Bryan: I want to know. Is this citation referring to like just general background knowledge? Is it like a use case for the invention? Is it like a tool or technique we use to make the invention? Why does this appear? And the reason we care, of course, is like. There's

262
00:38:28.840 --> 00:38:47.930
Kevin Bryan: cheap citations that Don't really mean anything, and there are citations that are super important to the advancement of science. And you know, in the year 2,023, we ought to be able to separate these things. And so we did a big survey. We have some ground truth that surveys are expensive. Every single answer we get to a question like this cost us 40 us dollars or 0. That's very expensive.

263
00:38:48.220 --> 00:38:53.510
Kevin Bryan: So can we get an Lm. To replicate some of this stuff, some of this

264
00:38:53.800 --> 00:38:56.030
Kevin Bryan: analysis that we are just asking the inventor about.

265
00:38:56.250 --> 00:39:13.310
Kevin Bryan: So you read this, and You see, Dot Dot, we visualized these cells on microscope stage as described in this article. They were then collected sequentially using this control of microscope. Okay, so you would read this, and you would say, that's got to be. That's got to be a tool they were using when they were developing the invention.

266
00:39:13.330 --> 00:39:15.140
Kevin Bryan: Okay. so

267
00:39:15.180 --> 00:39:18.630
Kevin Bryan: how would I try to get an L to understand this?

268
00:39:20.380 --> 00:39:40.040
Kevin Bryan: This is essentially all the code like in Python. I'm not joking. You can write this much code plus like importing the open AI library and loading your Api key, which is what charges you at. Open. AI, that's like 5 lines of code. And then this, that's everything. That's all you need to perform this task in Python.

269
00:39:40.040 --> 00:39:53.580
Kevin Bryan: And so what i'm going to do is i'm going to create one screen called instructions. I'll show you what this will look like, and then i'll give this other thing called patent, where I say, the article in question is the baroque one. The patent text is that text I just showed you on the previous slide.

270
00:39:53.730 --> 00:40:01.960
Kevin Bryan: Then i'm going to send this information in this structure to Gpt's Api. I'm. Using the chat. Gbt: Not Gp for just save money, but

271
00:40:01.980 --> 00:40:15.570
Kevin Bryan: and here I set temperature to 0. Temperature to 0 means essentially. I'm going to get deterministic answers. They're not going to be creative. It's not going to be like, you know what a snoop dog wrote like a rap song about. I don't know game theory

272

00:40:16.790 --> 00:40:33.900
Kevin Bryan: that the the implicit temperature that you're seeing on Chat Gbt: when you use that website is not 0. It's much higher, because it's trying to get creative answers. We're economics researchers. We don't generally want the creative Answer. I want the as determinist as possible, as truthful as possible. Answer. So

273
00:40:33.960 --> 00:40:35.640
Kevin Bryan: you take that temperature way down to 0.

274
00:40:35.820 --> 00:40:40.800
Kevin Bryan: What instructions am I going to pass? Well, there's a few things I can do right. So

275
00:40:41.450 --> 00:40:59.720
Kevin Bryan: the first one. This is just a guess on something that might work. I'll say, consider the sample from a patent one to a. Why, this state of paper was cited by this inventor potential reasons or general background, a tour technique, a use case. So I told the system, State, first background, knowledge, tool, technique, or example. Use, then state why you think the stage article

276
00:40:59.870 --> 00:41:02.150
Kevin Bryan: was cited by this adventure? Okay.

277
00:41:02.250 --> 00:41:09.870
Kevin Bryan: So this is just a guess. This will work just from some some playing around that I've had, but I don't know. So let's try a second one.

278
00:41:10.200 --> 00:41:28.060
Kevin Bryan: Here we go. The user wants to know why the State level is St. By this adventure that this looks very similar to you as a person. But in practice these these systems are a little bit of a black box, and minor changes in language can actually have pretty big differences in in how the system behaves. So

279
00:41:28.310 --> 00:41:37.950
Kevin Bryan: we'll try this one as well, and then let me try a few straight example. This is just I kind of chop this. Just so you'd be able to read it on this presentation on your computer.

280
00:41:37.950 --> 00:41:56.800
Kevin Bryan: But here I gave. Consider the sale from patent, and then I gave examples. User. Article in question is Drew's, and it's a patent text. And then I said, answer. This is background dollars, the article by

Joseph. First, the importance of Gpfr Drug User: the other questions Ferguson patent text dot dot response tool technique. This is the preparation of sales mystery to create a full convention.

281
00:41:56.800 --> 00:42:00.420
Kevin Bryan: And so in the full one I gave a bunch more examples that were based on ground.

282
00:42:00.530 --> 00:42:01.240
Okay.

283
00:42:01.810 --> 00:42:06.000
Kevin Bryan: And if it were me and I was performing this task.

284
00:42:06.350 --> 00:42:15.950
Kevin Bryan: I wouldn't like guess which one of these would work better. I would just use a trick that's well known to anyone who's used neural networks or or machine learning, which is called an ensemble Approach

285
00:42:15.970 --> 00:42:24.020
Kevin Bryan: the ensemble approach. Is I'm going to ask all of these, and then i'm going to have the Lm. Vote, and i'm just going to take the most common answer. So

286
00:42:24.060 --> 00:42:29.890
Kevin Bryan: this one, this particular, this particular text, I think, is not going to be very difficult

287
00:42:30.030 --> 00:42:32.910
Kevin Bryan: for an Ellen to understand. This is like a tool or technique.

288
00:42:32.920 --> 00:42:42.100
Kevin Bryan: But let's just see what it says, because I asked it for the explanation. So it's not even a black box. We'll we'll get some idea of why the Lm. Answered the way it did so. Here we go.

289
00:42:42.460 --> 00:42:57.710
Kevin Bryan: These are the 3 prompts I just gave you and the responses this number up here is how many tokens it cost me s049-54-5724 so totally cost me about 1,700 tokens, or point 3 us cents to do this ensemble approach.

290
00:42:57.910 --> 00:42:59.500

Kevin Bryan: And all 3

291
00:42:59.700 --> 00:43:02.720
Kevin Bryan: think this is a tool technique to our technique, to our technique.

292
00:43:02.970 --> 00:43:05.090
Kevin Bryan: right? And they give.

293
00:43:05.160 --> 00:43:24.540
Kevin Bryan: I would say, actually, all 3 of them give exactly correct explanations of why it's a tool or technique like exactly what a person would have given. And so you know, if if I say, did this for like a 1,000 different chunks of pads, where I want to understand why the reference is being used, I would guess most of the time these different prompting techniques are going to agree.

294
00:43:24.830 --> 00:43:26.510
Kevin Bryan: and the ones that Don't agree.

295
00:43:27.860 --> 00:43:35.800
Kevin Bryan: I can take the majority vote of the different prompts, or I can say, maybe I want a person to look at this on mechanical Turk and Ra to clean up the data.

296
00:43:35.860 --> 00:43:45.190
Markus Brunnermeier: Okay. So you don't know how quickly and also all straight to pity, to give them a probability estimate which category it fors in.

297
00:43:45.330 --> 00:44:00.060
Kevin Bryan: You need to be very careful. It's, or even you can ask it. How? How difficult is this to know? There are ways to write that prompt, that generally do a pretty good job. But I found it. It's quite finicky.

298
00:44:00.140 --> 00:44:02.130
Kevin Bryan: and

299
00:44:02.150 --> 00:44:05.430
Kevin Bryan: I mean I I see actually, I've seen some papers that.

300
00:44:05.570 --> 00:44:20.530
Kevin Bryan: like like economics papers, as you know, like what's the probability district? What's the the variance of this thing, given what

the example you've seen before, and it gives answers to look a lot like human subjects when you ask them to do the same task in in a lab.

301
00:44:20.570 --> 00:44:23.000
Kevin Bryan: so so we can do it.

302
00:44:23.170 --> 00:44:27.080
Kevin Bryan: This one I I feel like I've got total control.

303
00:44:27.090 --> 00:44:32.760
Kevin Bryan: whereas and and see it's not totally even, you see, like it didn't give a colon here. It gave a period.

304
00:44:32.780 --> 00:44:50.160
Kevin Bryan: and so the things I would do on the back end to clean this up even when the Lms. Not exactly what I wanted to do. To still kind of understand to a technique is what the how it responded. So I I would just throw this screen to a spreadsheet. I would probably like, say, you know, given this first answer.

305
00:44:50.160 --> 00:44:53.940
Kevin Bryan: how close do we think? Is it like close in in

306
00:44:54.790 --> 00:45:00.380
Kevin Bryan: It's a letter distribution to one of the 3 answers I've asked for that'll be sufficient to clean all this up.

307
00:45:00.990 --> 00:45:08.650
Markus Brunnermeier: but you can use the Llm again to go. Yeah, and I'll I'll I'll show you a great example when you use that.

308
00:45:09.650 --> 00:45:25.920
Kevin Bryan: Okay, I mean, how good is the any recognition. This comes from a friend of Duke. He he was at this, I think, at at the Nsf. And they gave him this document of Grants. They given the 1,900 eighty's on an old old fluffy disc.

309
00:45:26.100 --> 00:45:29.560
Kevin Bryan: and he opened it up in the file format. He didn't even know what it was.

310
00:45:29.970 --> 00:45:43.190
Kevin Bryan: No idea. He tried to open a file in a text editor, and you can see it's just tons of things that you can like. Can't even interpret

what the what the character is supposed to be like. How are you supposed to extract the data from this?

311
00:45:43.670 --> 00:45:53.880
Kevin Bryan: He used an Lm. To try to ask like, what is the structure of this data? Can you figure out what the main things are? And once he's done that he has the Lm. Extract those categories. And then he wound up with this.

312
00:45:54.710 --> 00:45:59.580
Kevin Bryan: Okay. and like that that came out of this. It's totally accurate.

313
00:46:00.350 --> 00:46:06.450
Kevin Bryan: So we're cleaning up really messy data super good. I'll give you one more example here. I took this from a.

314
00:46:06.690 --> 00:46:11.030
Kevin Bryan: You know Pdfs, especially economics. Pdfs are like a giant mess, especially when they

315
00:46:11.180 --> 00:46:29.930
Kevin Bryan: use latex and make a Pdf. There's all sorts of well-known problems copying and pasting such as stinks. I I don't want to pick on Marcus, but Marx does working papers in particular and possible got you at days from, so I in any case, I I took this paper of mine. It's got a little bit of math, and I

316
00:46:29.930 --> 00:46:44.600
Kevin Bryan: I, I. O. C. Art it! And then I put into a a little notepad, and you can see, Golly, this is a giant mess. So for one, the Latex code has a ligature between the F and the I, and a very common font we use

317
00:46:44.600 --> 00:47:00.350
Kevin Bryan: that if you look up here on consider question, Mark has no idea what this ligature is supposed to be right, and it happens a couple of other times as well completely gets wrong like here it says one minus pi, even though it's one minus P. Sub: I okay, so it's a mess

318
00:47:00.460 --> 00:47:01.210
Kevin Bryan: right?

319
00:47:01.320 --> 00:47:03.940
Kevin Bryan: And so in any case I wrote a little code

320
00:47:03.980 --> 00:47:16.460
Kevin Bryan: very simple. I would say no more than like 20 lines of code to to try to take textbooks like this and clean it up, but I ran it. This is what we get. I had. I also had to write it in in late that for me.

321
00:47:16.640 --> 00:47:18.380
Kevin Bryan: and this is perfect

322
00:47:19.080 --> 00:47:34.130
Kevin Bryan: like. What about what do you have to do? If to explain the pies now? No, no, I I I did not explain. I I didn't tell it what this that this should be, an fi I didn't say that the pi should be p sub i. But I did tell it. The context of this came from.

323
00:47:34.160 --> 00:47:35.320
Kevin Bryan: and the thing is

324
00:47:35.670 --> 00:47:40.040
Kevin Bryan: because it's an Lm. That understands next word. So it sees, you know. If

325
00:47:40.800 --> 00:47:49.770
Kevin Bryan: P. I are independent, right, it doesn't make any sense. If Pi are independent, and so it's going to think. Oh, it's

326
00:47:49.990 --> 00:47:54.280
Kevin Bryan: an Ocr mistake, so it goes. If piece of iron I mean this is perfect

327
00:47:54.390 --> 00:47:58.990
Kevin Bryan: in this. This is this is, you know, the original late Tech basically I'm. Recovering

328
00:47:59.300 --> 00:48:08.350
Kevin Bryan: so super useful for cleaning up messy text. You can see we Ocr stuff all the time. It's a giant mess. It's much easier to clean up this way than the other way.

329
00:48:08.830 --> 00:48:10.330
Kevin Bryan: How about Code?

330
00:48:10.800 --> 00:48:19.260

Kevin Bryan: I'll give you a really a really simple one, just because it's a fun one. Marcus has been this one before, so I I i'm just going to have the

331
00:48:19.620 --> 00:48:38.610
Kevin Bryan: this is Gp. 4. Give me the simplest python code for charged fine demand. I I gave it a a demand as a function of price on purpose, label, supply, demand, label, total surplus. Think, step by step this writing, thinking step by step, turns out to be a quite useful

332
00:48:38.610 --> 00:48:51.790
Kevin Bryan: for getting the correct response. Like it moves the posterior toward basically avoiding hallucinations. It's it's a nice trick for avoiding some hallucinations and then create the lay tech code. I type this in output. Here we go. Now.

333
00:48:52.720 --> 00:48:59.170
Kevin Bryan: Actually, the equations are all grabbed correctly. The only problem is, we're economists and since cournot we put price

334
00:48:59.190 --> 00:49:06.800
Kevin Bryan: on the y-axis quantity on the x-axis. That's not quite right. I just. I didn't bother to like redo all the code or anything. I just

335
00:49:07.180 --> 00:49:25.910
Kevin Bryan: typed into the lm that's pretty good. Remember that in this playing of mangraft he's on the Y access and queues on the X also make the app. He's go right to 0 on the plot. If you look over here, you see in the bottom left that little annoying spot here, and when you're making your graphics, always some stupid stuff like this happens. You're like oh, God! I got a

336
00:49:25.920 --> 00:49:33.630
Kevin Bryan: what! I what's the code? How do I clean that up? No, that's all a waste of your time. Look, I just type. I type one sentence in natural languages.

337
00:49:33.680 --> 00:49:46.160
Kevin Bryan: We drew the graph completely, correctly. Okay, and if I wanted to change the colors of stuff whatever I just like. There's no point trying to create these drafts by like typing ticks or whatever you do before the Lms are out right better

338
00:49:47.750 --> 00:49:49.760
Kevin Bryan: summarization. So

339
00:49:49.950 --> 00:49:55.840
Kevin Bryan: I told you that I I'd scanned a bunch of papers from kind of my own personal database.

340
00:49:55.950 --> 00:50:11.070
Kevin Bryan: and I use something called vector embedding. This is a little too difficult to discuss in the time we have. But you can think of this as 200 lines of python, and not 20, because it has to do a lot of stuff in the background. But basically I can now query

341
00:50:11.250 --> 00:50:12.950
Kevin Bryan: with an Lm.

342
00:50:12.960 --> 00:50:15.870
Kevin Bryan: Like literally tens of thousands of pages of text.

343
00:50:15.970 --> 00:50:28.360
Kevin Bryan: And so, you know, I've got this paper with with Mitch hoffen, and a mirror that we we did a a field experiment, and I just said, Give me the sample size, main methodology of the Brian at all start of hiring paper. I didn't even tell what the name of the paper was.

344
00:50:28.440 --> 00:50:32.580
Kevin Bryan: and that does a bunch of stuff in the background and try to figure out where this context might be.

345
00:50:32.770 --> 00:50:36.890
Kevin Bryan: Gives all that stuff to an L. The L. And just rewrite the answer. This is completely correct.

346
00:50:36.980 --> 00:50:39.080
Kevin Bryan: Okay, completely correct. The summary

347
00:50:39.480 --> 00:50:51.770
Kevin Bryan: and the cool thing again for people who are not native English speakers, since our scientific documents are in English, but you might be more comfortable in their language. So Don't worry, you know, for for for all the Germans here?

348
00:50:51.770 --> 00:51:00.900
Kevin Bryan: The the answer is, the answer is completely correct. If you, if you ask that question in German, you can get the yeah you have to give this, and and

349
00:51:01.050 --> 00:51:04.760
Kevin Bryan: there you go. There's still 26 firms in 1,877.

350
00:51:04.820 --> 00:51:07.850
Markus Brunnermeier: It seems like You' some high, some German in high school.

351
00:51:08.080 --> 00:51:11.160
Kevin Bryan: Yeah, that now that is the Spheric Marcus

352
00:51:12.070 --> 00:51:18.450
Kevin Bryan: and a problem. Okay. So so those are the main use cases. Now, practically

353
00:51:18.720 --> 00:51:24.330
Kevin Bryan: what do I want to do to make things as controlled and non hallucinatory as possible. So

354
00:51:24.570 --> 00:51:27.580
Kevin Bryan: multi-step process is best. Do not open up

355
00:51:27.910 --> 00:51:32.230
Kevin Bryan: the web browser and go to chat. Gbt. Get yourself access to the Api.

356
00:51:32.590 --> 00:51:39.470
Kevin Bryan: Then you want to feed in data generally with fu shot, prompting the examples you choose should be

357
00:51:39.490 --> 00:51:52.140
Kevin Bryan: drawn as widely as possible from the distribution of the things you want the Lm. To understand as possible. Honestly, you can think about this as like you have an an Ra. You need them to like, categorize some things.

358
00:51:52.360 --> 00:52:00.790
Kevin Bryan: What examples would you give them that would like Cover kind of all the broad cases that might come up, or as much of them as possible. That's what you want to give the L into. Okay.

359
00:52:01.050 --> 00:52:11.600

Kevin Bryan: play around with the prompts that ensemble approach. I gave you works really well. So you don't have to like, guess what the best prompt is. But just be aware, I I've seen computer science papers where

360
00:52:11.630 --> 00:52:13.650
Kevin Bryan: differences in prompts that are like

361
00:52:14.030 --> 00:52:19.900
Kevin Bryan: trivial lead, one prompt to match the ground to 60%. Another problem matches the ground for 99%.

362
00:52:20.140 --> 00:52:31.270
Kevin Bryan: So for this reason, since we don't actually know in the computer science literature how to control these things fully of what awful prompting looks like. I tend to think the best thing to do is use an ensemble approach.

363
00:52:31.310 --> 00:52:37.660
Kevin Bryan: So it's so cheap to query the the Api. Just write a few reasonable looking prompts.

364
00:52:37.710 --> 00:52:50.780
Kevin Bryan: Test them out a few times, and then just have the Lm. Vote essentially on on, but not to hit the the memory limit more easily in the assembly. Approach. No, because you're you're actually sending you're sending each prompt, totally independently.

365
00:52:50.880 --> 00:52:58.930
Kevin Bryan: I see. Okay. So i'm setting one prompt with this instructions: Independent? Answer: One problem. This is our independent answer, and then it's gonna look at the answers it gave and take the majority vote.

366
00:53:01.010 --> 00:53:02.420
Kevin Bryan: So

367
00:53:03.650 --> 00:53:05.780
Kevin Bryan: follow up questions.

368
00:53:05.880 --> 00:53:11.340
Kevin Bryan: and I don't mean in like a chat Gpc sense, but in your code are really useful. So, for example.

369
00:53:11.690 --> 00:53:14.900

Kevin Bryan: if I send a bunch of context. and I asked

370
00:53:15.120 --> 00:53:25.620
Kevin Bryan: what has been, give me the names of all the authors that appear here. It's gonna give me some names. Sometimes it will hallucinate, no matter how good you are prompting.

371
00:53:25.980 --> 00:53:29.920
Kevin Bryan: What I would do then is, I would send another question to the Lm.

372
00:53:30.220 --> 00:53:31.530
Kevin Bryan: Where I send

373
00:53:32.780 --> 00:53:39.270
Kevin Bryan: A. User tells me that the authors in this paragraph are X. Y. And Z.

374
00:53:39.560 --> 00:53:52.820
Kevin Bryan: This is the original paragraph. Are these users, in fact, an Xyz. Answer: Yes, no. Okay. And this one's a pretty easy task. Actually, it's much easier than extracting the authors.

375
00:53:52.960 --> 00:54:02.100
Kevin Bryan: And so, if it says no, maybe you want to look at that on your own and double check if it says yes, i'm feeling really confident and didn't didn't hallucinate.

376
00:54:02.290 --> 00:54:07.320
Kevin Bryan: And lastly, this this temperature of 0. When when you're using the Api.

377
00:54:07.450 --> 00:54:12.960
Kevin Bryan: almost all use cases that we care about in research, I would take the temperature down as low as you, as low as you can.

378
00:54:12.970 --> 00:54:23.910
Kevin Bryan: Like Creativity is generally not what we want in our our our our our as, or mechanical Turk, or whatever we're previously using. It's not. It's not. We want. We want people to code so

379
00:54:23.920 --> 00:54:36.550
Kevin Bryan: down to 0, sometimes, for like unusual coding, like. If you code and data, for example, where there's much less content on the

Internet, it can help to bump the temperature up a little bit. It'll help the the kind of

380
00:54:37.120 --> 00:54:47.660
Kevin Bryan: the Lm. Figure out some aspects of data code that are not that obvious. But if you code in like Python, for example, and you want to ask some python code. Me. This simple thing in python

381
00:54:47.690 --> 00:54:56.330
Kevin Bryan: definitely temperature equals 0. There's so much that the Lm. Has read about how python structure works from the Internet that it's going to get to get the answer correct.

382
00:54:56.730 --> 00:55:03.340
Markus Brunnermeier: So to to understand the randomness of this system. So it it doesn't always take the most likely posterior.

383
00:55:03.860 --> 00:55:17.880
Kevin Bryan: because otherwise you would always get the same if you ask twice the same thing, and so temperature of 0 would give you twice if you was twice the same, you could twice the same. Answer. I'm sure 0 in my experience always gives a deterministic cancer.

384
00:55:18.000 --> 00:55:25.510
Kevin Bryan: I understand from the the computer science literature. That is not necessarily the case is not a guarantee.

385
00:55:25.730 --> 00:55:28.900
Kevin Bryan: Now it doesn't mean it's always true.

386
00:55:28.990 --> 00:55:40.040
Kevin Bryan: It just means it's always deterministic. These are the same thing. But unsurprisingly deterministic and true are pretty heavily correlated for a lot of tasks. Right?

387
00:55:41.310 --> 00:55:45.030
Kevin Bryan: Okay, how do we handle limited memory. So one

388
00:55:45.040 --> 00:55:47.690
Kevin Bryan: break the task into different

389
00:55:47.740 --> 00:55:53.320
Kevin Bryan: Break your content into different chunks. So, for example, if I want to search

390
00:55:53.520 --> 00:55:55.740
Kevin Bryan: they a paper that I wrote.

391
00:55:56.000 --> 00:56:04.950
Kevin Bryan: or like some legal documents, hundreds of pages or particular thing like I would like to know, like in this giant legal document.

392
00:56:05.000 --> 00:56:09.150
Kevin Bryan: say a zoning bylaw. What's the maximum height of a house

393
00:56:09.320 --> 00:56:11.650
Kevin Bryan: in this city? Okay, bye bye.

394
00:56:11.800 --> 00:56:23.010
Kevin Bryan: So if I was a person, I I guess I can just control at this hopefully. They say height. Maybe they say, you know maximum elevation. Maybe I don't know what they're going to say, so You might not find it even with control, that

395
00:56:23.080 --> 00:56:25.730
Kevin Bryan: you can't send 100 pieces of text to an Lm.

396
00:56:26.160 --> 00:56:32.550
Kevin Bryan: So all you do is you break this text into chunks of a 100 words, 200 words overlapping them.

397
00:56:33.010 --> 00:56:41.110
Kevin Bryan: You embed them, meaning you're going to take these like chunks of text and put them in like 1,500 dimensional space

398
00:56:41.480 --> 00:56:52.640
Kevin Bryan: where some chunks of text, a similar meeting wind up in similar parts of space. If you use Burt or other similar inventions, you're already familiar with this. It's a idea that precedes Lms.

399
00:56:52.950 --> 00:57:01.120
Kevin Bryan: And then you're going to take your question like, what is the maximum height in the city? You're going to embed that also in the same space you're going to

400
00:57:01.520 --> 00:57:07.630

Kevin Bryan: through the dot product, the cosine similarity to find the chunks of text most likely to be about the height of a house.

401
00:57:07.770 --> 00:57:15.380
Kevin Bryan: and then you're just gonna send all that to the L. You're gonna say Lm: here's a bunch of context from a bylaw. What's the maximum height of a house in the city.

402
00:57:16.290 --> 00:57:29.420
Kevin Bryan: So now i'm only sending like a 1,000 words to the L. But it's a 1,000 words that we're very judiciously chosen on my end, and that the compute here to do all this stuff on your side is basically free. So it's not recording about.

403
00:57:29.470 --> 00:57:34.110
Kevin Bryan: So this is this kind of like a vector embeddings to find the right context

404
00:57:34.360 --> 00:57:40.240
Kevin Bryan: Generally, the way that we we get around limited memory the other way we do it is that we have a database that

405
00:57:40.470 --> 00:57:43.060
Kevin Bryan: we have the L. Right. The query

406
00:57:43.070 --> 00:57:52.040
Kevin Bryan: the Json query to query the database, rather than passing the whole database to the Lm. So these are, I think, not that

407
00:57:52.060 --> 00:58:07.410
Kevin Bryan: complicated. There are no packages which do this for you, or you have to program it on your own. So the the embeddings, like open AI has through their api and embedding. You can use. That's very expensive, you know. A few cents for a big Pdf, for example.

408
00:58:07.450 --> 00:58:19.660
Kevin Bryan: So I would just use that. I would certainly wouldn't try to code this myself if you're familiar with like or or similar kind of word embeddings or sentence embeddings. You can run that on your computer.

409
00:58:19.720 --> 00:58:24.990
Kevin Bryan: It's a little bit slow. I just wouldn't bother, because so Gp is an Api to do this.

410

00:58:25.290 --> 00:58:25.890
That's right.

411
00:58:26.170 --> 00:58:29.840
Kevin Bryan: And also this memory thing, maybe, is only a short run problem.

412
00:58:30.110 --> 00:58:33.030
Kevin Bryan: It's not totally clear

413
00:58:33.060 --> 00:58:38.300
Kevin Bryan: at a reasonable compute cost how much memory. We can give these models so like in 2,025.

414
00:58:38.500 --> 00:58:43.430
Kevin Bryan: Will this be deprecated or not? I I just don't know right. It's possible we'll solve memory limit.

415
00:58:44.520 --> 00:58:45.880
Kevin Bryan: Kind of do math.

416
00:58:46.150 --> 00:58:51.480
Kevin Bryan: No, the status goes bad. Do not do Matthews and Lm. It is not good enough.

417
00:58:51.610 --> 00:59:01.960
Kevin Bryan: That does not mean that you can't get an l and do that if you're clever. So I and here I mean, even if you have access to the beta of the wol from Alpha plugin to Gp. For it's still not gonna.

418
00:59:02.340 --> 00:59:10.500
Kevin Bryan: So what I would do if I really need to do that is, I would send the question to the Lm. I would ask it: Are there any things that need to be computed

419
00:59:10.790 --> 00:59:12.290
Kevin Bryan: when I ask you that

420
00:59:12.480 --> 00:59:16.740
Kevin Bryan: I would say, Write me python to Code to do that computation.

421
00:59:17.090 --> 00:59:28.890

Kevin Bryan: and then I would have my code on the back end. Take any response that says, right, run this python code, run that, and then pass the answer back to the Lm. Doing the compute on my side. So, for example, like.

422
00:59:29.370 --> 00:59:32.970
Kevin Bryan: think about how this works. Imagine I I multiply 2 8 digit numbers together.

423
00:59:33.250 --> 00:59:41.550
Kevin Bryan: Those 2 ages numbers have never been multiplied together on the Internet before that's not in the corpus of human language. Okay, so there's no way

424
00:59:41.590 --> 00:59:55.700
Kevin Bryan: that something just predicting the next letter or the next word is going to get the right answer. It is going to get the first, 5 6 7 digits correct, because things that look kind of like the first number of times kind of like the second number, have actually

425
00:59:55.780 --> 00:59:58.810
Kevin Bryan: appeared in the human corpus before.

426
00:59:58.860 --> 01:00:00.490
Kevin Bryan: but it's not going to give you the right answer.

427
01:00:00.510 --> 01:00:12.630
Kevin Bryan: Of course this is so easy to fix. You Just say, tell me the multiplication. You want to do that. I just pass it to whether it might be a programming languages to do it. So there's many ways to get around the math problem, but it but it it's still a bit of a problem

428
01:00:12.850 --> 01:00:21.090
Kevin Bryan: I mentioned already. I don't want to talk more about this. We don't know the way to optimal prompt. You can't like Go read the computer science literature and understand the right.

429
01:00:21.150 --> 01:00:35.590
Kevin Bryan: the right wording to use like it's all totally experimental at the right now, and Chat is only 5 months old. 5 months from now. There'll be all sorts of new tricks that people figured out that that are better. But the idea i'm mentioning here, like ensemble methods

430
01:00:35.630 --> 01:00:43.990

Kevin Bryan: of Lms like vector embedding to get around memory all that is, I am sure we're still going to be around a year from now, 2 years from now.

431
01:00:44.660 --> 01:00:45.850
Kevin Bryan: What's coming soon.

432
01:00:46.400 --> 01:00:53.900
Kevin Bryan: Then we can't do so. Ocr sucks. It's really awful having to scan all documents. They never scan really. Well.

433
01:00:53.930 --> 01:01:00.990
Kevin Bryan: there are demos of Lms reading images, reading the text off images. It really really high quality.

434
01:01:01.030 --> 01:01:10.320
Kevin Bryan: I suspect we will see Ln. Based. Ocr. In the next year there will be superior to High Tester Act, or whatever as you happen to use right now.

435
01:01:10.590 --> 01:01:15.810
Kevin Bryan: Code, instead of have asking me Lm. To write you some code. Then you

436
01:01:15.910 --> 01:01:32.290
Kevin Bryan: pass it to your programming language. It'll just be able to run this in line as part of the Lm. You already have something called code interpreter. That's a beta with an AI right? Now that can do this, and obviously searching the web to verify facts. You can do this now. So if you find the fact, what you can do is

437
01:01:33.130 --> 01:01:36.750
Kevin Bryan: use Google's Api to search Google, grab the top results.

438
01:01:37.020 --> 01:01:46.580
Kevin Bryan: Pass that back to the Elev and say is the fact that I wanted here confirmed by my web search. Okay, I mean, this is not hard. You can program it in. I don't know 20 lines of code.

439
01:01:46.620 --> 01:02:05.630
Kevin Bryan: but doing this automatically, so that you can kind of pull facts in real time when you're talking to an Lm. Being Gpt. Is a little bit able to do this, but not very well. Obviously this is the kind of next next step, and the computer scientists working at something called the Information Retrieval, or knowledge graphs

440
01:02:05.630 --> 01:02:10.970
Kevin Bryan: are combining the ideas they have in those fields with Lms to do some really exciting things. That's coming soon.

441
01:02:11.440 --> 01:02:23.530
Kevin Bryan: All right. I want to make sure we have time for questions, so i'm going to pause here. I have a couple of little demos. I'll show you if we have extra time, but otherwise let me give it to the audience. And and you, Marcus.

442
01:02:24.220 --> 01:02:36.170
Markus Brunnermeier: Yeah. So there's some questions. Perhaps you can cover them, and then come back to the applications. But most of the questions are, you know what you mentioned? All the advantages of the dilemma.

443
01:02:36.300 --> 01:02:43.110
Markus Brunnermeier: Is there some risks associated with that as well as on the ethical side? Is? Can you use it in a malicious way?

444
01:02:43.330 --> 01:02:45.700
Markus Brunnermeier: That's one by type of question.

445
01:02:45.900 --> 01:02:48.320
Markus Brunnermeier: And what do you see? The dangerous?

446
01:02:48.840 --> 01:02:51.330
Kevin Bryan: Yes, it's it's interesting.

447
01:02:52.380 --> 01:02:53.560
Kevin Bryan: There are.

448
01:02:53.710 --> 01:03:02.060
Kevin Bryan: There's like the long term Lm. Risk that you you know these kind of computer science. Folks sometimes worry about that come from some

449
01:03:02.910 --> 01:03:06.070
Kevin Bryan: effective altruism web words. And then there's the

450
01:03:06.120 --> 01:03:12.070
Kevin Bryan: that are hypothetical, and it's not obvious they'll ever come to pass. But then there's short-run worries that are

451
01:03:12.140 --> 01:03:15.180
Kevin Bryan: currently possible and super worrying. So, for example.

452
01:03:16.780 --> 01:03:19.750
Kevin Bryan: it is very very very cheap

453
01:03:19.920 --> 01:03:20.950
Kevin Bryan: to run

454
01:03:21.240 --> 01:03:28.270
Kevin Bryan: like, Think of like the emails that they would send to your your parents. Your grandparents try to steal money from them. They were not that personalized.

455
01:03:28.350 --> 01:03:36.420
Kevin Bryan: I can now run text messages where I respond, using knowledge about you that I pull off the Internet

456
01:03:36.530 --> 01:03:40.710
Kevin Bryan: basically for free without even needing like a call center in India to do it.

457
01:03:41.010 --> 01:03:47.020
Kevin Bryan: I mean it's so worrying like when I talk to my folks. For instance, we have a code word.

458
01:03:47.120 --> 01:03:54.140
Kevin Bryan: so that if someone pretends to be me. This word is not on the Internet, and if I say it, if they they, they're not sure. I said, ask me for this word.

459
01:03:54.490 --> 01:04:00.060
Kevin Bryan: I would say this kind of thing if you don't have it, especially with older folks you talk to. I would already be doing it.

460
01:04:02.850 --> 01:04:06.180
Kevin Bryan: you know, for research. I'm not sure what the what the

461
01:04:06.220 --> 01:04:13.790
Kevin Bryan: the maliciousness is that I would worry about. I mean it's generally like a super useful tool for for teaching and for

462
01:04:13.830 --> 01:04:17.330
Kevin Bryan: and for research through some of the some of the cases I showed you today.

463
01:04:18.860 --> 01:04:21.870
Kevin Bryan: But yeah security, it's a I mean

464
01:04:21.890 --> 01:04:27.410
Kevin Bryan: all all the things that cost like a dollar. To try to scam people

465
01:04:27.720 --> 01:04:30.900
Kevin Bryan: now cost is a penny that's that's obviously not going to be good.

466
01:04:31.070 --> 01:04:39.990
Kevin Bryan: and then we don't. Have. We Don't have good ways to stop that. The other question was related to this is, you know, check. Gpt. Was using all this data from the Internet

467
01:04:40.220 --> 01:04:51.050
Markus Brunnermeier: should we treat this as a pool of a global public good at this pool of knowledge, and does it need some international regulations? So in the music industry, for example, that try to protect existing music.

468
01:04:51.660 --> 01:04:58.030
Kevin Bryan: I mean it's it's a it's a bit weird, so like, you know. I'm an innovation economist. I study copyright as well.

469
01:04:59.360 --> 01:05:10.170
Kevin Bryan: All the people making songs today. Listen to other songs. That's how they learn to be a songwriter. It would be a little weird for someone to say, oh, you like the Beatles as a kid. And now you're now you're Drake.

470
01:05:10.190 --> 01:05:19.770
Kevin Bryan: you know. So Drake owes royalties to the Beatles. That's just not how we think about copyright, and similarly like if I if I wanted to like, learn German, and I like.

471
01:05:19.960 --> 01:05:21.810
Kevin Bryan: you know, red like.

472
01:05:22.210 --> 01:05:42.170
Kevin Bryan: I don't owe the money like it's kind of it's kind of a crazy question, I think. Now with with art. Maybe it's a little trickier because you're like. Go! This is to to allow you to like, Replicate your in the style. Take a person. But again, if I was teaching art class, and I want my students to make art the look like, say, a particular artist.

473
01:05:42.170 --> 01:05:47.270
Kevin Bryan: I would send them on the Internet to look at what that art looks like, and then, you know, no one would say that I owe the artist money.

474
01:05:48.300 --> 01:05:49.580
Kevin Bryan: I suspect

475
01:05:49.640 --> 01:05:56.700
Kevin Bryan: that if you are famous enough that people will use your I your name to create new content.

476
01:05:56.760 --> 01:06:08.440
Kevin Bryan: You're basically, in fine, yeah, that's only going to make you richer. The other thing I should also mention is that what we what we've learned is the initial training of these models uses tons of data.

477
01:06:09.290 --> 01:06:11.880
Kevin Bryan: but we can get 99% of the results

478
01:06:12.110 --> 01:06:18.100
Kevin Bryan: using much less kind of a much more sparse set of wings.

479
01:06:18.410 --> 01:06:19.380
Kevin Bryan: so

480
01:06:19.820 --> 01:06:24.690
Kevin Bryan: that tells you most of the content was actually not really relevant.

481
01:06:26.400 --> 01:06:35.740
Kevin Bryan: which probably isn't that surprising, and we're going to get a little better at learning what type of content matters and what doesn't. And then it won't be a problem that you don't want your stuff to be

482

01:06:35.930 --> 01:06:41.980
Kevin Bryan: ingested by an alm like Honestly, we don't need it. Let's use cases. So who cares? You know?

483
01:06:42.610 --> 01:06:52.890
Markus Brunnermeier: So so just in Wolf's would like to know what teaching related. I know you've developed some teaching software as well on, based on, but in particular is interested about. You know. How should we

484
01:06:52.930 --> 01:07:02.930
Markus Brunnermeier: assigned homeworks? So avoid cheating. Do you have some recommendations on this dimension to do? We have to go to oral examination now, and more paper submissions or

485
01:07:03.210 --> 01:07:14.900
Kevin Bryan: so. This is a little bit of a different track. What you should be aware of is that there it is not possible to know whether text is written by an Lm.

486
01:07:15.130 --> 01:07:17.580
Kevin Bryan: Even if a student doesn't change it at all.

487
01:07:17.690 --> 01:07:24.880
Kevin Bryan: it's impossible to know. And if they ask the Lm. For ideas and for how to write the essay, and they just type it in different words. Obviously, you'll never catch them.

488
01:07:25.010 --> 01:07:26.080
Kevin Bryan: So

489
01:07:26.170 --> 01:07:28.950
Kevin Bryan: I mean, we caught many people

490
01:07:29.070 --> 01:07:38.680
Kevin Bryan: cheating in December, using Gpt. At our University by having someone stand in the back of the room and look at their computers as they typed an in-person exam.

491
01:07:39.020 --> 01:07:40.100
Kevin Bryan: So

492
01:07:40.750 --> 01:07:45.220

Kevin Bryan: you know like, I would expect anything that can be done with an Lm.

493
01:07:45.730 --> 01:07:55.430
Kevin Bryan: The students will cheat because it's just it's just very easy. In fact, we're all economists here. We understand substitution. You saw this company called Cheg, which is what your students were previously using to cheat in your class.

494
01:07:55.490 --> 01:07:57.820
Kevin Bryan: Their market Catholic, fell in half.

495
01:07:58.300 --> 01:08:07.450
Kevin Bryan: because it turns out that it's just cheaper to it's cheaper to have Gpt Write your essay than someone in Kenya right, your essay, which is what people previously think.

496
01:08:07.800 --> 01:08:26.550
Kevin Bryan: So there's so there was a undergraduate student who wrote a senior ceases and a software called Gpt: Cr: yeah. Yeah. So it does. It doesn't work. A lot of it is a lot of administrators think this works. So if if you went to Gpt and you asked it to write messy on something, and you took the results and you paste it into like

497
01:08:26.620 --> 01:08:30.330
Kevin Bryan: Gpg. 0, it would say, 99% this comes from.

498
01:08:30.710 --> 01:08:35.189
Kevin Bryan: But if you ask me to spend 1 min to generate the same message.

499
01:08:35.210 --> 01:08:46.410
Kevin Bryan: I can get it to say 1% comes from Tbt and i'm sure the students are better at. They can read it also, and they know, like I mean literally take whatever you think.

500
01:08:46.520 --> 01:08:48.340
Kevin Bryan: Gbt: 0 is identifying

501
01:08:48.399 --> 01:08:58.109
Kevin Bryan: and paste that back into Chat Gbt: and the only instruction you type in is, we write this, so the content is the same, but it does not look like it comes from

502

01:08:58.620 --> 01:09:04.350
Kevin Bryan: an Lm. It's going to add a couple of spelling mistakes and a couple of things, and then can't find it anymore.

503
01:09:04.580 --> 01:09:18.340
Kevin Bryan: So so a lot a lot of companies, because of things like prompt and injection attacks and other security worries are trying to solve the problem of when this, When is this computer generated text? And when is it not? And we do not. We do not have an ability to do this.

504
01:09:18.439 --> 01:09:19.970
Kevin Bryan: So assume

505
01:09:20.040 --> 01:09:23.960
Kevin Bryan: that if you give an assigned people can cheat on the cost of cheating is now free.

506
01:09:24.060 --> 01:09:29.630
Kevin Bryan: and so so that's how much team I would expect to get. I would probably push

507
01:09:30.109 --> 01:09:40.399
Kevin Bryan: evaluation to stuff I do in person. or I would. What I do in my class is you're allowed to use Gpt. However, you want.

508
01:09:40.920 --> 01:09:44.180
Kevin Bryan: You just have to in the same way that you can talk to your classmates.

509
01:09:44.260 --> 01:09:46.140
Kevin Bryan: You just can't have them do your assignment.

510
01:09:46.189 --> 01:09:53.140
Kevin Bryan: So I ask you to if you use Gbt to state exactly where you'd be used for a spell check. You set state it when you hand in the assignment.

511
01:09:53.859 --> 01:09:57.220
Kevin Bryan: Now, just like you could have cheated by having your friend Ricky assignment.

512
01:09:57.360 --> 01:10:06.070

Kevin Bryan: But I don't tell you. You can't talk to your classmates. The the same kind of norm needs to develop, and we also need the same kind of trust that we had previously. But but that's basically all we can do.

513
01:10:08.070 --> 01:10:18.940
Markus Brunnermeier: So I know that you have developed them software for teaching. So they a lot of questions about the teaching teaching aspects. But let's do what you still have time for this examples you want to go into, or do you want to see a quick one.

514
01:10:19.170 --> 01:10:22.800
Kevin Bryan: Yes, let's let's let's do. Let's do this. Here

515
01:10:25.560 --> 01:10:26.960
Kevin Bryan: you share this with you

516
01:10:27.330 --> 01:10:27.930
all right.

517
01:10:28.460 --> 01:10:29.590
Kevin Bryan: so I get a

518
01:10:29.890 --> 01:10:43.190
Kevin Bryan: I got a I I need you to tell me, how do I open the Powerpoint back to the previous line? Okay. So so here's one like as an example of something we're using in 3 classes right now that I wrote here at Rottman.

519
01:10:43.270 --> 01:10:47.110
Kevin Bryan: Basically, think of this as Gpt.

520
01:10:47.200 --> 01:10:48.770
Kevin Bryan: For your class.

521
01:10:48.980 --> 01:10:55.310
Kevin Bryan: So all of our all of our transcripts, our handouts, our slides, our readings, our optional readings.

522
01:10:55.360 --> 01:10:59.900
Kevin Bryan: are ingested into an lm using

523
01:11:00.260 --> 01:11:06.030

Kevin Bryan: it's not fine-tuned because it's too expensive. But it's basically using this vector embedding idea I mentioned before, plus like

524
01:11:06.310 --> 01:11:12.730
Kevin Bryan: lots and lots and lots of back end code to make sure there's no hallucination and make sure nothing's.

525
01:11:12.760 --> 01:11:14.820
Kevin Bryan: No, that's a made up.

526
01:11:15.070 --> 01:11:23.300
Kevin Bryan: And so, if you, a student, asks one of these questions like, for instance, this my entrepreneurship class our I CEO's initial coin offerings a good way for a cryptostar for his money.

527
01:11:23.420 --> 01:11:27.250
Kevin Bryan: In my class I would have said something like.

528
01:11:27.530 --> 01:11:39.850
Kevin Bryan: you know, in 2,016, 2,017. These are super common. If you'll raise a lot of money with them, there's quite a lot of fraud. This led to a regulatory crackdown, more or less. It's dead these days. Even crypto firms are not raising money in this way. And so

529
01:11:40.080 --> 01:11:46.190
Kevin Bryan: the student asked this: you type this into the Internet that is not the answer you're going to get. You take this into this system.

530
01:11:46.240 --> 01:11:53.420
Kevin Bryan: I mean it says it. It just says exactly what I just said, so it's just pulling it straight from our

531
01:11:53.510 --> 01:12:00.550
Kevin Bryan: straight from our the context of our lectures. So it allows the students essentially to have a 24 over 7. Ta. They can query.

532
01:12:00.610 --> 01:12:08.920
Kevin Bryan: which is pretty useful. I also have a another thing I wrote, that lets you using those same documents

533
01:12:10.050 --> 01:12:19.180
Kevin Bryan: auto generate a question bank of like hundreds and hundreds of questions, and then you can just like chuck the ones you you don't

like, because some of them won't. Be very good questions You keep the ones you like.

534
01:12:19.190 --> 01:12:20.780
Kevin Bryan: And now you've got, you know.

535
01:12:22.120 --> 01:12:31.770
Kevin Bryan: 2 300 question bank that the students can just. I want to know more about this topic. I don't feel like I understand it pretty well. And the cool thing about this is it's not just multiple choice where

536
01:12:32.250 --> 01:12:35.110
Kevin Bryan: I give the answer, and you say you got it right or you got it wrong.

537
01:12:35.140 --> 01:12:43.850
Kevin Bryan: This is an Lm: so passing the L. A bunch of context about why, why it like that it generated about why it generated these off.

538
01:12:44.440 --> 01:12:47.510
Kevin Bryan: So then, when you give these answers that you say like a

539
01:12:47.780 --> 01:12:52.250
Kevin Bryan: you know, like a here's like what's what's a minimum vial product

540
01:12:52.910 --> 01:12:57.610
Kevin Bryan: should be an easy question, and this student goes launching a product with maximum functionality to gave feedback.

541
01:12:57.640 --> 01:13:00.260
Kevin Bryan: So obviously this is like the opposite of what you want to do

542
01:13:00.340 --> 01:13:06.250
Kevin Bryan: like an Mvp. Should be like kind of. They're both product in order to learn quicker about consumer demand for various features.

543
01:13:06.390 --> 01:13:24.780
Kevin Bryan: And so this not only tells you the answer is, be it also explains why your answer was wrong, and tries to give some explanation on. So the student can try to understand kind of what they're doing

wrong. So how is this different from Khan Academy? Scan lingo which uses open? And I to you know, teach like a tutor.

544
01:13:24.870 --> 01:13:34.930
Kevin Bryan: Yeah. So very, very similar to what kind of I mean. The underlying idea is the same as like Dualingo, Max or Khan Academy. I think my mine was earlier. I I built mine in March.

545
01:13:34.990 --> 01:13:43.700
Kevin Bryan: but the thing is like this: one is completely controllable for your content, and for the kind of stuff you want students to have access to.

546
01:13:43.730 --> 01:13:57.370
Kevin Bryan: So if I was teaching, like, say, Spanish, 101, or like, you know, intro to Alpha or something. I just send the students to kind of academy. There's they. They have teams of engineers building stuff, but if I wanted to like replace

547
01:13:57.590 --> 01:14:05.010
Kevin Bryan: a ta on spending a couple hours training, for, like one of my classes. especially, a class, is more qualitative.

548
01:14:05.130 --> 01:14:07.080
Kevin Bryan: like theory of the firm or something like this.

549
01:14:07.970 --> 01:14:17.460
Kevin Bryan: You can just do this here, and you basically have something that looks like Khan Academy. So this is how hard was this to build. I I spent 2 afternoon to make this, and you can actually

550
01:14:17.590 --> 01:14:37.580
Kevin Bryan: you just download it. It's it's done. You have. You can do whatever you want with it. So so this is. This is like not complicated stuff. but they will make a huge difference in teaching, and as we kind of get better lms like Once the mathematics ability gets better, say, in 2,024 to 2,025. And we can start using this to

551
01:14:37.600 --> 01:14:46.070
Kevin Bryan: to help students understand where they're making the mistakes and derivations, or in the equilibrium concept. I mean it's going to be a game changer, for how we can.

552
01:14:46.250 --> 01:14:54.950

Markus Brunnermeier: At this stage it's mostly for things which are very what based. So if you want to have figures in it, the math and it it will not work at this stage. This

553
01:14:54.960 --> 01:14:56.300
Markus Brunnermeier: we don't care. Yet.

554
01:14:57.180 --> 01:15:08.490
Kevin Bryan: in the sense like I mean, I showed you that I can use the Lm. To generate that client demand graph which was just created in Python. And of course I can then just pass that to this

555
01:15:08.490 --> 01:15:17.390
Kevin Bryan: this this is a web app using something called Flask. I just passed that figure to flask. I just need more stuff in the background, so the can't do all this itself right now.

556
01:15:17.460 --> 01:15:19.930
Kevin Bryan: But you, if that's the feature you want.

557
01:15:20.070 --> 01:15:23.960
Kevin Bryan: It would not take you a long time to be able to add that.

558
01:15:24.140 --> 01:15:25.580
Markus Brunnermeier: Add that functionality.

559
01:15:27.720 --> 01:15:44.320
Markus Brunnermeier: Very good, thanks a lot. I think we ran way over time today. Hope, but people are are still here, so I appreciate a lot, and we will do more on this line of teaching, because I think teaching will change dramatically as well. And we're trying to learn this, and Kevin is on the frontier

560
01:15:44.350 --> 01:15:53.120
Markus Brunnermeier: on this dimension, as well as other people. So things are changing, drinks are moving. And, as Kevin said, very fast, so very grateful that he, princess

561
01:15:53.140 --> 01:16:05.390
Markus Brunnermeier: up to the frontier, and how we can use it for economics. Of course there are a lot of the sources on the Internet but I think he is more focused on for economists and other social sciences how to make a practical use of that.

562

01:16:05.620 --> 01:16:08.710
Markus Brunnermeier: Thanks again, Kevin, and hope to see you soon again

563
01:16:08.950 --> 01:16:11.100
Markus Brunnermeier: and enjoy your time in Boston.

564
01:16:11.180 --> 01:16:13.360
Kevin Bryan: and I hope it was useful for everyone else. Thank you.

565
01:16:13.520 --> 01:16:16.720
Markus Brunnermeier: Thanks and thanks for everybody for coming this one.

566
01:16:16.880 --> 01:16:17.450
Bye.