Week 5 Video Transcript

0:00

okay so week five we're going to start talking about data science Here's the uh learning outcomes

0:06

and uh so let's start with uh data science with

0:12

generative AI So what happened was that this large language models they evolved

0:18

uh into a more elaborate solution Um and

0:24

what happened is that LLMs gained access to computer systems and to uh actions as

0:32

we talked about last week in the React uh framework So

0:39

um LLMs are they they kind of incorporated a more agentic style and

0:46

one thing that was very transformative was uh the Paul framework Paul

0:54

is um is is prompt assisted language and

0:59

uh a programming assisted language and it's um and it's uh it's the thing is

1:06

that the LLM can now have a coding uh

1:12

uh can execute programming language and I mean code to um help its reasoning So

1:22

we saw that one of the the bad re the bad things that LLMs had was uh

1:27

reasoning for example mathematics right because uh what the LLM's doing is

1:33

predicting the next token but it's uh in with math that's that's

1:40

kind of hard to do right um it's not in the uh underneath the hood it's not

1:46

actually doing math so Paul uh was very

1:51

transformative and what it it is is the LLM can

1:57

actually evoke computer systems in particular a

2:04

programming language uh and in in the case of most LLMs is

2:10

Python which is a programming language uh so that you can code stuff and this

2:16

is usually this this uh This ability to evoke and

2:22

to act on computer systems is controlled by what we call the orchestration

2:28

library So you you talk with the LLM the LLM talks back um and the orchestration

2:36

libraries will get the your um your prompt and will send this to when needed

2:44

to what the Python interpreter So you will run code and then the Python

2:50

interpreter will send information back and so on So I'm just going to give you

2:56

an example Uh let's say that we want to to solve a math example a math problem

3:03

with uh the uh chat dpt and you ask

3:08

here's the question Roger has five tennis balls He buys two more cans of

3:13

tennis balls Each ken has three uh tennis balls How many um how many uh

3:23

balls tennis ball does he have now so if you run this question in a pure um

3:31

in a pure prompt uh and just the question and stop there

3:36

uh you know the the LLM might just get the answer wrong right and in this in

3:43

this case it got wrong Um but so what you can do is knowing that now

3:51

the LLM has access to the Python interpreter you can help the LLM by

3:58

oneshot example or uh or um chain of thought reasoning

4:05

uh on how to do the logic Right so the

4:11

here you would prompt with a oneshot example like this This is a question

4:16

Roger has five tennis ball answer Roger started with five tennis balls And you

4:23

can see that this is a this has a hash This hash means that this is a comment

4:28

in Python Tennis balls equal to five So that's

4:34

what you would create in a in a coding assignment uh then two cans of tennis balls each is

4:42

bought balls equal to 2 \* 3 right so

4:47

you're creating these variables hash tennis balls the answer is uh tennis

4:54

balls plus bought balls and then you can come up with a new question so here what

5:00

you're doing is 5 + 6 the answer will be um 11 but you're telling the LLM

5:07

uh with Python language on how to solve the problem Okay

5:14

and then you come with a new question So the bakers at the Beverly Hills Bakery

5:19

baked 200 loaves of bread on Monday morning They sold 93 loaves in the

5:24

morning and 39 loaves in the afternoon A grocery store returned six unsold loaves

5:30

How many loaves did you did did they have left so it's you can see that it's a very uh similar question and when you

5:39

run this then exactly the um

5:45

the the LLM will answer you following the same the same reasoning right and

5:53

you can see that the reasoning is following the reasoning is made in Python and is

6:00

following what you instructed the baker start with 200 loaves So it's creating

6:06

for you the this this is the answer right so it's creating for you variables

6:13

the comments in Python and they're it's running this code that is is um that the LLM is

6:22

writing in Python Now I'm going to go and give you a practical example of this

6:28

open here and um I did exactly what I showed you This is a oneshot uh prompt

6:37

and here I gave this in terms of its variables

6:42

and I asked the question and you can see that the answer is this sure here's a structured answer in the same

6:48

step-by-step styles style using code style comments and variables

6:53

right so the baker baked 200 loaves so it's kind of reasoning and it is um running the python program

7:04

to have an answer Okay So that that's what it uh what Python

7:12

what uh Paul does Now it is important to give sometimes

7:19

oneshot inferences um

7:24

because it uh in in in more complex problems the AI can kind of struggle But

7:31

for this ones this kinds of questions I would say that you don't even so the technology is so robust right now that

7:39

you don't even need to to call uh to call this right So let's

7:47

change the problem to

7:52

uh something like a different problem like this

8:02

It's just a very just a modification

8:08

And you can see that the the answer Oops

8:14

Oh so it it it found inconsistencies and because I forgot to um

8:22

uh to change here the tennis balls But uh

8:27

in in the revi he the the LLM even um revised my question to make to make

8:35

sense and it runs a Python code So when you see a Python here it means that this

8:41

is a Python code and then it's running the Python code and giving us um the

8:49

answer Obviously it's it's asking me which version is better Um

8:54

but the technology is very mature Okay And in doing that

9:02

um in doing in having Paul we can now do amazing

9:09

things which kinds of leverage data science and I I'll show you something

9:14

because because um

9:20

because we can have access to Python there are many many things that we can

9:27

do with files with uh uh figures and

9:32

everything So I'm going to what I'm going to do I'm going to import an Excel file just to show you that

9:40

uh the large language model per se doesn't have this capabilities

9:48

right um of reading uh uh Excel files or

9:54

PDFs The LLM is just a probabilistic machine that's outputting

10:00

the pro the most probable token But with the orchestration library we

10:07

can we can run code and by running code we can import things We can uh read PDFs

10:16

we can um read data and then construct data analysis

10:22

uh by using our own natural language English or whatever language you speak

10:28

But we now the LLM actually interprets what you asks for in what you ask for in

10:36

your prompt and translate it quote unquote to to um

10:44

Python and you can have the you don't need to know how to code but if you if

10:50

you know concepts of data science enough you can come up with amazing tools uh

10:56

because Chad GPT for example will program for you Okay

11:03

So here's here's uh here's an example I'm going to import this spreadsheet and

11:10

uh I'm going to ask the LLM ask chat GPT to summarize

11:18

uh this the columns and try to extract knowledge from from this spreadsheet So

11:25

here is a uh spreadsheet

11:30

with uh my finite mathematics

11:36

course grade Please analyze analyze this

11:44

uh document and all its columns Um oops spreadsheet

11:53

Um please analyze and summarize

12:01

sum summarize so I'll I'll I'll go about what I'm do

12:07

why I'm doing this but let's see how it goes so here here here it starts when

12:13

when you give a prompt and the LLM knows it has to call Python because the LLM

12:21

per se doesn't have this capab ability it will call it will evoke Python

12:26

and if you you can you can kind of collapse this button here to do not show

12:32

the code but it's very good to see what what the LLM is doing

12:38

and uh it's actually importing the file it's creating uh like all the

12:47

going through all the columns and here it comes after evoke invoking Python

12:54

uh the LLM can read each column and then

13:01

can make reasonings from that from this document Right

13:07

so it it comes and says "Oh your spreadsheet contains great information for students That's the username That's

13:13

the first name That's the uh the numeric score student

13:18

received on the final exam It's taking it from the columns all the informations

13:24

and um there's a column named final grade override symbol

13:31

um and it's exactly that is a column that kinds of reflects the the the grade

13:37

used in records So if you had 98 uh 68.5

13:44

the override will will put you in with 69 kinds of as a rounding column but

13:50

this is what is used for the final grade and there's an end of line indicator and even if you want like you can uh have a

13:58

um you can have a visualization of the table here and you can even open

14:06

uh bigger Okay you can you can do that Um but now we have a very powerful tool

14:13

tool We can do we can do data science with this thing Um

14:19

just um to be clear the language for data science you you

14:25

can do data science in any programming language Um but the the the most common

14:33

and powerful language today um is Python Python is the the the

14:39

language of of data science and knowing Python is very important um even if

14:46

you're just you're not in the uh AI it

14:52

uh or computer science but knowing the basics of Python will

14:58

will open you doors to interpret what a large language model is doing for

15:04

example and help uh and help

15:09

um will and this will help you do more informed decisions and even guide the

15:16

LLM through a certain analysis Right so um this is what Python did You can

15:23

collapse this right so you can every time it uh it shows analyze it's because

15:28

it analyzed the document with a Python uh call Let me show you

15:35

another um another

15:41

example Now I'm going to uh to send to

15:48

the LLM Uh just one second

15:55

I'm going to uh send to the LLM a PDF So please

16:03

Oops This PDF is not working Oh it's my my PDF It's it's blocked It's

16:11

um Okay So I'm going to send another one I'm going to send this PDF and I'll say

16:17

"Can you please analyze and summarize

16:25

the this PDF in page 8

16:31

So you can see that um

16:36

you can see that the the it the LLM called the code interpreter

16:45

is actually uh it's actually extracting still that's why the analyze didn't appear yet But if you this is a test but

16:52

if you go to page eight which is exactly this this page is intentionally left

16:58

blank Okay Uh so but you can see that now you have much much more power

17:06

right so let's go back to the presentation

17:12

and uh here's the code interpreter here this is one other example that is very

17:17

good so if you if you put this puzzle in for chatpt and only chatpt to to solve

17:26

you will see that it won't work uh so I need Um here is the prompt I need you s

17:33

to think step by step how to solve this problem with a dictionary Um

17:39

so so the original problem would be the uh I need to find a word that fits the pattern letter E letter letter T The

17:49

words cannot include any of these letters here Okay And uh the word is a standard word in

17:56

the English uh language So if I just did that like just without this first uh

18:03

attachment and the the first part of the prompt the the LLM will fail miserably because

18:11

it's just that the u it's just just uh estimating the next token once I

18:19

input a file and I told it to analyze step by step

18:27

with the dictionary It means that the LLM will because of my prompt saying

18:33

that I have to solve this problem with the dictionary will evoke Python code

18:40

that will read a dictionary of words of the of the English language with with

18:46

thousands of words um and it will and the co and we'll code

18:52

in Python permutations of letters and we'll find um

19:00

we'll we'll find the answer So having code interpreter and you can if if you

19:07

have some task that chat DPT is trying to do and it's not um evoking Python or

19:14

or or the code interpreter as we call you can always ask it like please evoke

19:21

Python or the C code interpreter to kind of guide your answers So in the end of

19:27

the day um you can you can see the code

19:32

interpreter this Python uh orchestration

19:37

in action uh you can view it as this super powerful solution where you can

19:44

create files you can process files you can transform and and extract

19:49

information from files and um and then after that so when you extract

19:56

information this information is kind of being populated in the prompt So if we go back

20:02

here the information about my this this Excel file here is now

20:09

populated in the in the prompt And then we can ask questions about it Okay So

20:16

we're going to um uh we're we're going to continue this uh just in a sec Uh but

20:24

the now you have this powerful solution So

20:30

what you can do like what what what things this this huge

20:37

solution that has an LLM and a code interpreter can do well it can take

20:44

documents and and can treat it as objects So every time you have a task of

20:49

renaming of deleting moving copying downloading uploading archiving restoring print even printing these are

20:57

things that this this are operations you might perform to a file uh with code

21:05

like with the code interpreter helping you And on the other side the and chatpt

21:12

couldn't do that by itself Now uh on the other side you can do document uh you

21:18

can do what chat GPT does that it's very good so can analyze summarize read

21:24

interpret extract information search translate translate so that's um things

21:32

that just chatt now can do plus all of this kinds of more um um

21:41

mechanical things okay like like like with files but not only files

21:49

um we we we can use the code interpreter to code and

21:56

again um just to make it clear uh Python is very good to

22:04

do this kinds of stuff Okay So let's go back to the code interpreter and let me

22:10

start a new chat just just to so that we can start um

22:16

so that we can start fresh new and I'm going to again to give this a CSV file

22:26

and and I asked I I made the same question

22:34

Okay So it's running Python code and it's coming with the with the summarization of this documents Why

22:42

every time we're doing some data analysis with ChachiPT or with other LLM

22:49

and uh and uh the

22:56

I I always ask Chat PT to come up with a summary so that it can generate this

23:02

code going going column by column and making

23:09

the the overview view of the columns or like be

23:15

in the prompt be here in the context window because I'm going to continue my

23:22

conversation with it and whatever is loaded here um it's kind of in the rag

23:29

uh framework this will be appended to whatever prompt is next and that's why

23:36

chat tpt remembers things it it at some point you will reach

23:42

uh like you you start forgetting older things um and and that's why it's very

23:48

important to maybe sometimes create files so that you can come back to

23:54

whatever conversation you were having So for example before before going on uh

24:00

and doing data science with this I will I will ask chatpt to save a um a text

24:08

file with the columns overview just in case um after several interactions this

24:16

kinds of gets lost and I really want this overview to always be available and

24:22

and to so that I can always come back to this overview if if needed So first of

24:28

all what I'm going to do is I'm going to I'm going to come up and say uh can you

24:34

please save this column

24:39

overview into a plain

24:46

text file so that I can download and save it in my computer

24:56

And again uh the LLM is not capable of saving things and producing things So

25:03

it's call it's calling a Python code that will save the output

25:09

and um you can now download this So if I click here I can start the download and

25:15

then I can kind of save this and at any point when I want

25:22

to save a conver you can save the conversation per se right as well you

25:27

can save like this will be in your history in chat GPT like as a chat Uh

25:33

they they they keep saving all the chats uh but if you uh you can kind of um uh

25:44

save all the steps if you if you want So now

25:50

what I what I want to do is uh continue with his offer Uh chatb says "Would you

25:57

like a statistical summary of student performance mean uh mean minimum max standard deviation

26:05

or a visual analysis?" So you can see that now um things that I was I could

26:12

use Excel to calculate or Python or uh other software such as SAS such as um um

26:20

WKA that you've you've covered in um in the first week uh second week I can now

26:29

come and ask ChatgPT to do and I don't need to know the programming language

26:34

anymore obviously And I'm saying this again it is important to have some

26:40

programming skills uh so that you don't drift too much from reality And just

26:46

remember that this machines this LLMs they want to ple they are made to output

26:52

something Uh it's a probabilistic machine and it's usually uh trained to

26:57

please us I mean to engage us and sometimes they just hallucinate right

27:05

so knowing this um um knowing the basics of Python

27:12

for example is very important So one of the things I want to I want to first start is uh I want you to calculate the

27:23

mean of the final

27:30

uh course average of the course average column Okay So and you will see that

27:40

again chatpt cannot do that um by itself So it's it's calling Python

27:46

code and here's just uh chatt is will kind of do a a feedback you a user

27:52

feedback here with me kind of asking what type of response that I prefer and

27:59

um it's it's it's actually the same so I don't care Well it it ran the Python

28:06

code and the average was 61.31 So now you can see that this is

28:12

calculated with the programming language without having to kind of guess that the next token is six the next token is one

28:19

the next token is three and the next next token is is one Now um taking a

28:26

look at the uh the table I notice that there are some missing

28:32

values and I want to know what what the LLM like how how the LLM managed the

28:40

missing values because this is was a discussion that we had that missing

28:45

values are important for data analytics So you can ask

28:51

so how did you tackle

28:57

missing values on the course average

29:05

um well before asking how it tackled show me

29:12

uh are there are there any missing values

29:21

on the course average column

29:28

let's see what it has So again it's running in Python code that has this uh

29:35

uh um it's uh this function this

29:41

function that it calls and tries to so yes there are eight missing values

29:47

in the course average column Okay So can you display

29:55

these eight values with the respective columns

30:02

so that I can assess it

30:08

So again it will create a table and to create a table we'll call well it's evoking Python

30:15

Um so here we go So see you can see all this eight people we have uh student

30:21

names here um here are the eight entries with missing values in the course including their usernames names let me

30:28

know if you want to input or clean this values right so this is something

30:34

important that now that I know that this is happening I can ask chat dpt

30:40

to uh to recalculate the mean without this value So for example

30:47

recalculate the mean of this uh of the

30:55

course column without these values So I'm just going

31:01

to remove um this values Uh and so now I

31:07

discovered that yes so what the what the uh the the LLM was doing what ChachiPT

31:14

was doing is was calculating uh the mean already kind of disregarding these

31:20

values I could have asked directly but here it is So you can do statistical

31:26

analysis um uh on several documents but just be

31:32

careful Uh having knowledge of programming is very good for this uh and

31:39

trying to have all the information into the conversation kind of asking the

31:46

LLM to summarize the documents at first uh is very important because you're uh

31:53

you're pulling uh like you're you're really pulling the information

31:59

uh and then to to the context uh window and then you're doing the rag So you're

32:06

pushing all of the in all of the history together with your prompt Um you can do

32:14

you can do lots of of uh um um

32:20

a very kind of intricate analysis with the programming language here Um and you

32:30

can also generate visualization So now

32:36

that you know the data set can you come

32:43

up with uh two different uh interesting interesting

32:52

visualizations uh for me

32:57

So uh again it will run Python and you can just uh collapse this if you're not

33:02

interested in seeing it but I always kind of take a look um on what is happening here

33:09

but um so here it it came up it came up with um

33:18

with two different types uh of of graph right so the first is the the

33:25

distribution right uh on the course average So this is a histogram

33:30

uh of the course average and you can see uh that it generated automatically for

33:36

me and you could ask different things on um

33:42

modifying right so let me collapse this and the other one is uh the final exam

33:50

score versus the course average so in this document I'm not showing all the

33:56

other tests But you can see that the final exam is a good predictor of the uh

34:03

of the course average in because in this course this is the um the highest weight

34:11

Okay Um but you can see that this is a good predictor So you can come up with

34:17

different analysis and you can uh edit this and even work with the suggestions

34:25

Why why why did I I could have asked like come up with the histogram but it's

34:32

very one of the things that is being modified um uh that LLM is is modifying

34:40

uh in data science is the way that we interact with data and actually doing the flipped interaction Maybe the AI can

34:49

have ideas that I'm not seeing it Right so um this um

34:57

you can go and say can you trace this uh graph for me can you come up with the

35:04

visualization a specific thing but you you also can ask the the AI to come up with something

35:12

interesting Okay So this is this is uh this is very very um it's interactive

35:19

It's it's very new but you can you can edit this grass for example

35:26

um so uh I want to change so in the

35:34

in the histogram of the course averages

35:42

Uh could you let me ask could you change

35:48

the width the width

35:54

with the bins in the histogram averages I want the

36:03

histogram to have only

36:08

10 uh only six bins I don't something like

36:14

that and you can see that because this is in the context um the the LLM can go here you it will

36:25

it will see whatever it was created in the past it will see the Python code it

36:30

created and will modify the Python code to kind of get this Okay

36:35

So um you can add labels trend lines you can

36:43

then you can download this images and everything So here you go Uh let's edit

36:48

the image first I'm going to collapse this Um and there

36:53

you go So it created one two three four five six bins And um

37:01

uh you can you can so it's asking you do you like to overlay a mean median lines

37:06

yes please overlay mean

37:12

median lines to this plot

37:17

and see how it goes And you can see the the like the Python code If you are um

37:28

if you are comfortable with Python you can do you can get this code You can

37:34

copy and paste this code on your own on your own um uh Python interpreter and

37:41

and um so on So let's say that I want to save this I can ask if uh uh if uh we

37:48

want to to save Um but

37:53

another thing that we can do that is very impressive is that the LLM

38:01

um because of the orchestration library can evoke Python to create files So I

38:08

have to show this these graphs in the

38:15

uh uh those graphs to the academic dean

38:21

at the university Uh can you help me prepare

38:28

an Excel file with all these

38:36

uh figures on it the Oh not Excel sorry Uh PowerPoint

38:42

PowerPoint The images

38:49

um should go uh one

38:56

uh should should go on a single page And

39:02

please insert a caption on the right hand side of the figure

39:11

with a brief description

39:17

And so it's so that's the beauty of things this computer system that is um

39:23

uh that is a the Python interpreter can generate things and it will generate this is a kind of

39:31

verbose code but it will try to generate uh for you the PPT the PowerPoint and um

39:43

and it will uh there's an error here I have to check

39:50

So it's saying that it cannot import a a library It tried again So it it kind of

39:56

fi it fixed its its own code And then you can download this Let's see how was

40:03

uh what's the there you go So it obviously it's not

40:10

perfect right um but this saves a lot of time into the phase of analytics The

40:19

phase of analytics it really saves a lot of time

40:25

when we have uh the like the the code interpreter to work for us Okay

40:33

Now it is very important to be careful uh to be very careful with uh hallucinations

40:39

Um use the use the the LLM use chat PT for

40:47

example to analyze your data but be aware of the data Open the data in your

40:53

Excel as well Keep an eye on it Uh see if it makes sense and you know test the

41:00

LLM always try to come up with questions that could you know that could um

41:08

that could uh kind of challenge the LLM to to see if the data analysis is making

41:17

sense Okay continue talking about this capabilities uh from from our uh Excel file

41:28

we can ask the AI to generate a linear regression for predicting the final

41:33

average on the course So uh

41:39

you you experienced uh using data analytics software such as

41:45

WKA uh or NIme or SAS to build machine learning models

41:51

prediction models in this case a regression model

41:57

and uh no tools low tools uh low uh no code low

42:05

code tools are very useful uh if you're you're not comfortable with with uh

42:11

programming languages but now you can pro you you can uh English is a is a

42:17

programming language because an LLM can actually translate this to a

42:22

programming language So look at this capability I asked for the AI to

42:29

generate a prediction model It's a simple linear regression but it could be a neural network And of course knowing

42:38

how to code will uh will make you more

42:44

confident in accepting this Uh and it the AI makes mistakes So mistakes it

42:51

will happen Um and but but in the end of the day if you have the basics you can

42:58

be more confident But you can see here that uh the the chatbt to even comment

43:05

on and this hash is a comment in um in Python code So drops rows with missing

43:13

values in either column Um so I'm just dropping but you know this could this

43:20

this is okay because only eight students from more than 120 dropped Um but

43:28

sometimes if if you don't have uh values you have missing values you could substitute by the mean the mode and we

43:35

talked about this so you could ask if you know uh Python you would see this

43:41

and say oh this is what the AI is doing or if either interpreting the comment

43:46

and ask the AI to to modify its code Um

43:52

but in the end of the day what the AI is doing is fitting a linear regression getting the coefficients the slope the

43:58

intercept and R squared which is a measure of goodness So here here you

44:04

here you have so the linear regression model has a R 2 of 82 So it's very good

44:10

it's a good prediction model and has a slope of 1.36 and an intercept of 2530

44:18

So if you go back to week two these are the parameters and this is the measure of goodness of the model and you can

44:27

um you can uh the AI asked would you like a visualization of this regression

44:32

line on the scatter plot yes And it even plots the line here and you can see that

44:38

this uh linear regression is very good Um and so it's a good predictor right

44:44

there are outliers For example this one uh where you would predict a course

44:50

average around 40 If the final exam is around 10 you would prever u predict an

44:58

average on the course around 40 but the person had a seven uh 65

45:04

So it might be that the other uh this person went very well uh on the other

45:10

tests So another another capability again uh if you want to use the AI to save things

45:18

um and although you don't know although it might be that you don't know Python

45:26

but you could save this model in a file so that if you want to use it again uh

45:32

so uh so that it will predict uh uh if you want to predict the course

45:39

average from a new student you could upload again to to the LLM

45:47

right so you don't need to run any kind of uh interpreter in your computer because you don't know how to do that

45:53

You're more low code no code profile You can ask the AI to save

46:00

this file for you uh and run um and and and the AI will will run a

46:08

Python code It will kind of com compress this code into what we call a a PKL is a

46:16

a a pack a model package and um you can download So you can

46:22

download this Oh I expired my session Let me renew

46:28

this Um so I will download the model

46:35

and

46:51

Um I think I lost every so I I kind of I I was uh I don't have my login anymore

47:00

Let me see here if I could

47:11

If I just could um

47:19

I'll pause and then I'll I'll make this work here

47:30

The file I logged in again I saved the file and let's start a new chat Like

47:36

this chat here doesn't have any kind of memory or context And what I'm going to

47:41

do is I'm going to here is a model

47:46

um a machine learning model I created

47:52

from data I want now to predict the output from a

48:00

new unseen data And I'm going to input that the student

48:09

the student uh student A had a final exam grade

48:19

off um of 45 points

48:27

what would be the students

48:33

uh predicted course average

48:38

and then I'm going to um what I'm going to do is I am going to

48:46

um where is it

48:52

and then I am going to

48:58

upload the the model and just to give context to the machine learning because

49:04

this uh to the to the AI this AI it hasn't uh seen the data that was trained

49:10

on and anything So I'm going also I'm going to also add the the grades

49:21

and And so let's see how it goes

49:32

So here's uh the AI is analyzing

49:42

Based on the model a student with a final exam grade of 45 points is predicted to have a course average of

49:48

approximately 86.43 Now could you graph the

49:58

regression line with this exact point

50:04

uh which is 45 on the x- axis and 86

50:10

43 uh could you graph this is it

50:16

highlighted highlighted

50:21

highlighted let's see

50:29

and again um for example if you understand Python you can check

50:34

everything that's being made again the um the AI can and probably probably will

50:43

have some uh problems so here Um

50:49

so here is the the the graph Okay So you could create a uh a machine learning

50:57

algorithm You can do all the pre-processing steps uh with the AI uh

51:04

with a large language model by just giving it instructions But be careful

51:09

again this is not uh the LLMs They are probabilistic machines and they can kind of uh incur

51:18

in errors incur in hallucinations Sometimes you you ask a question and it

51:25

will just predict something even if there's no way to predict So the all the

51:31

knowledge that we created on week two three and four is essential to come to

51:38

this point so that you can know how to talk with the AI how to

51:47

um how to instruct the AI and know what you're doing Right this is a very simple

51:53

example Well where you have just two columns and one is predicting the other

51:58

You don't even have to normalize because they have uh very similar um um they're

52:04

not that different in scale But when you're giving an instruction uh you

52:11

can make you can ask questions So would it be good to normalize and we saw

52:19

normalization in week two or three So would it be good to normalize the

52:25

columns in this uh the numeric

52:32

columns in this data set Let's see what it says

52:40

So here's a quick quick breakdown So normalization is helpful and it's it's very good

52:47

to kind of um to kind of

52:52

um talk to the AI right so uh but you could ask for a

52:59

normalization then again check the data set check if the normalization is is

53:04

doing what is the right thing for me to do was to also when I started um

53:12

upload the uh the the the plan right the the summary with all the columns

53:21

um the overview here the columns overview

53:26

because again this AI didn't haven't seen any context on that So this would

53:33

for sure make my AI uh less prone to

53:39

errors um at some point you're having conversations and having conversations

53:44

with your AI and um you created a model you used the model you normalized um and

53:53

now you know like what means for example to create a a multi-layer perceptron if

54:00

you are using images right uh if you're using um you could even uh ask the AI to

54:07

to train a multi-layer perceptron here to do this regression It would be a big

54:12

hammer for a a very small problem But that's fine You can do that But at some

54:18

point this all this contacts as being you're you're at some point the initial

54:25

um um the initial uh information will kind

54:31

of get lost Although this this context window windows is is really getting

54:38

really really really uh uh evolving to very very big things But um you can you

54:48

can again just let's remember our original you can say let's remember

54:55

oh remember the our goatee original I'm sorry I could just uh oops gone let's

55:03

remember our original data set uh and our goal our goal that is

55:12

predicting and so you can go and remember So what I used to say is that the LLM is your

55:19

intern your assistant So imagine this uh it knows how to

55:26

program uh but you you cannot trust them Okay

55:32

this is very important because we know that they they they are kind of uh

55:37

designed to please They are designed to do predictions no matter what we have to

55:45

be aware that they can make mistakes and that the memory is not so good So

55:50

breaking up into little pieces breaking up uh and saving those pieces and

55:58

sometimes feeding it back and going back to what what the um

56:06

what the the LLM was doing right so it

56:11

is very important to when you're when you're working with a code interpreter to think that it's a an AI right so um

56:20

really break it in step by step as you were um talking to your intern

56:27

uh and because if you if you kind of spot an error you can go back to that

56:33

specific uh step Okay Um provide all the necessary context and information to

56:41

accomplish a task This is very important The more information you give the better

56:47

Provide detailed feedback on what to improve

56:53

Uh do not expect that to be perfect or error-free And again this models that

56:59

you will create in Python and you can run in in inside again inside the uh the

57:06

GPT they might be they might not be flawless and again knowing the basics of Python

57:14

and what you're doing in terms of data science is very very uh important

57:20

Uh and um again you have to be you you have to be

57:27

very specific about your goals about your constraints about your needs and um

57:33

break down as much as possible Okay

57:38

Um now um the first thing we should do is to ask

57:45

questions about documents or ask the LLM to summarize all the columns or can you

57:51

tell me what you learn from this data set because we want the LLM to run the

57:58

Python code where it's extracting the text from the the the the spreadsheet or

58:04

the PDF and it's putting into the context window It's put it into the

58:09

conversation Um and be careful with the hallucination

58:14

So go go go really step by step and it's very easy

58:20

to work with small documents such as the ones that I just uploaded Right working

58:25

with small documents is fast You can create several different models So just as a curiosity let's create a let's

58:33

train So let me go back to to the to the other

58:39

conversation uh because we had more context in that

58:45

and um I just closed and uh open the other so that you can see that you can

58:51

upload the model So now could you create

58:56

could you train a multi-

59:01

layer perceptron

59:06

to accomplish the task of predicting

59:12

predicting the students final C the student's course average

59:21

But again if you if you do it like this your intern that knows how to program

59:26

will program any like it will program any multi-layer perceptron It won't take

59:32

um and and you can you can come up with a model that's working very nicely but

59:39

you you know you you need to um you know now that you have this concepts on these

59:46

models you can be more specific and know what you're doing For example this multi-layer perceptron how many layers

59:52

do you want to try um um you know uh is this um

1:00:01

can we try you could you could ask even the model to try to after training to

1:00:07

test for overfitting So these are concepts that now you have and you can bring from natural English from natural

1:00:16

language to uh to programming language So um let's see what le let's let's

1:00:23

leave it this way and let's see what um the AI will do So again

1:00:31

um um it's opened the code interpreter with the Python code It's calling uh as

1:00:38

scikitlearn which is a a package of that has the multi-layer perceptron uh

1:00:44

regressors models inside and you can see that it's trying with a

1:00:50

uh input layer of 10 neurons and an output layer of two layers with um 10

1:00:57

and five neurons right and the activation functions are the relu um I

1:01:04

mean you can if you know uh if you're if

1:01:09

you're more in depth with all these concepts you could go and make your models better

1:01:16

So as as you go and as you have machine learning as a course uh statistics as a

1:01:22

course um and and advanced analytics as a course you can you can um kind of go

1:01:30

more in depth with your model Okay Um so uh here you here you have the the result

1:01:39

So this is the multi-layer perceptor model has been successfully trained to predict students course So this is the

1:01:46

architecture your intern chose but you again you it's your intern and it can

1:01:53

make uh mistakes So it it is very important that you should be very very

1:01:58

critic uh you have uh our our squared uh score

1:02:04

So it's it's lower than the linear um regression So likely due to simplicity

1:02:10

of the task meaning that you have a model complexity that's bigger than your

1:02:16

task If you recall week three uh or week two model complexity can take you to

1:02:24

overfitting So it's likely that you are overfitting this model Okay

1:02:30

So uh it even already kind of created the the um the PKL model Okay So uh you

1:02:40

could ask and you could go with all the data science um that we learned all the process and stages uh of the of the of

1:02:49

the um of data science So you can you can ask um would so can you

1:02:59

change the number of neurons in the layers

1:03:07

I think you don't need to put this but I think less neurons

1:03:14

in the layers so making less parameters to train in each neuron uh would uh

1:03:22

would take um would would make the the the model

1:03:28

less complex and then uh less uh uh prone to to overfitting Right so what

1:03:37

we're going to do is can you change the number of neurons in the layers i think less neurons in the layers uh would

1:03:44

be less prone to overfitit

1:03:49

use Let's specify four and four

1:03:54

on the respective layers And there we go So you you can see now

1:04:00

that it changed the code and it's running the code and the result was oh

1:04:08

a smaller architecture is very very very So now we're not

1:04:14

overfitting We're not generalizing So because you have all this concepts you

1:04:21

can talk to the AI and and go through So again this models are have so we need to

1:04:29

go in between So let's say could we try

1:04:38

using a 10 and uh a eight and two

1:04:54

Let's see how it goes And and you can change the activation right so this is a

1:05:00

relu ReLUs are usually used for classification You can change this for linear act for a a sigmoid activation

1:05:10

Change the activation function to a rel to a sigmoid function

1:05:18

And anyways so you can you can go and you can really really kind of test uh

1:05:24

everything and um you can you can even say something like

1:05:31

this um from the original data set

1:05:38

um divide my uh data into 90% uh into 85%

1:05:49

of training and 15% of testing

1:05:56

uh eliminating

1:06:01

having eliminated or disregarded

1:06:06

the the the nans or the missing values the missing values right so it will generate

1:06:15

the data set for the it's a uh the you can see your train test split so 45

1:06:21

samples for training nine samples for testing Now you're ready to train and evaluate

1:06:27

models on this data set So so now you can go and evaluate different models on

1:06:33

for example the test set right um you can even do early stopping Uh if you

1:06:40

go back to the concepts of early stopping as you have a validation data set you can you can also do that or or

1:06:47

you can ask for cross validation You can you can uh um use this uh for

1:06:56

um this kinds of machine learning things Not only this but you can really um you

1:07:04

know you can you can uh because it reads Excel files and reads this kinds of

1:07:10

commaepparated values which are t tabular data This these guys are very

1:07:16

good into Excel So if you if you're if you have an Excel table and you want to

1:07:22

do a formula right so you can do create a another column to my data set in which

1:07:33

I uh show the

1:07:38

I don't know uh the the max value of the

1:07:44

course average average minus the

1:07:52

uh create another column to my data set or actually I'm going to I want to

1:07:58

create another column to my data set in which I show the max value of the course average minus the final points

1:08:08

in the final exam for every row

1:08:17

I will do this in Excel

1:08:22

Please go step by step

1:08:28

showing the formulas used Right so if you you don't need to to

1:08:35

work with the code interpreter uh like with with the chat dbt and and the poll

1:08:41

kind of framework You can actually if you're using an Excel you can come and

1:08:48

it will give you a good formula right because it again it can interface with this computer systems

1:08:55

um and you can even the the AI will even ask you if you want to download it the

1:09:02

Excel version with this formula prefilled right so

1:09:08

all right interesting tool um capability

1:09:14

of this things here of the LLMs is to cross reference documents and join

1:09:22

tables and and kind kind of do some very uh intricate analysis So I'm going to

1:09:28

upload this new file So I am uploading this new file

1:09:36

uh where grades from other uh uh assessments

1:09:44

can be found Okay Please again please go through

1:09:52

other assessments Please go through the

1:09:58

columns uh to make sense of the data

1:10:06

And so uh this is another table You can see that I have username last name first

1:10:11

name the section the quiz points uh and then I have midterm points and

1:10:18

everything Um I mean there's lots this is a a bigger

1:10:25

kind of document right so it did the analysis

1:10:33

and then let me collapse this uh so detailed grading components so we have

1:10:39

uh quizzes quiz one to six midterm one midterm two final grade points grade so

1:10:45

you can see that this column is repeated with the simple um Excel

1:10:53

together with the course average and override But I have new information here

1:10:59

right um I also the uh the the Python

1:11:04

code can also perceive um kind of notice when a column is

1:11:10

created by the user with a formula So average quizzes it's uh uh it's some

1:11:16

column that I created to average the the quizzes average in midterm average final

1:11:23

um average flex right it's a bonus component exactly and so on would you

1:11:29

like a breakdown of how how the course average is being calculated from this component or should I help you add a

1:11:35

column so what I want is actually I want

1:11:40

to uh join the

1:11:48

um let me the

1:11:53

let me copy the name of this file here just to be more specific I want to

1:12:00

join the fine no it's okay you don't need to be

1:12:06

so so I want to join this uh spread

1:12:11

spreadsheet uh with the

1:12:20

uh 570 to import spreadsheet

1:12:28

the first document I have uploaded

1:12:34

so that the override column can come

1:12:42

from so that the over overall column can come from the

1:12:49

first document first document So uh substituting

1:12:57

the override column from the second

1:13:03

document So what I'm going to do is just like so you can usually you have several

1:13:08

um you have uh uh you know like several uh several files do you want to mix them

1:13:16

together and you have like an identifier in one and identifier in the other This is

1:13:23

something that uh chatgp2 can do and it will do And if you look at the at the

1:13:29

python um um code you see that it's cleaning up the

1:13:35

columns and merging the the things and uh the session reset has removed

1:13:42

access for previously uploaded file Okay So um let me upload

1:13:50

So what I'm going to do is I'm going to upload the file

1:13:56

So let me go back to

1:14:02

I'm going to upload

1:14:09

the file

1:14:14

and Here

1:14:20

you go This is the previous This is the first

1:14:29

document

1:14:38

Okay So now now that GPT has both required files and it will do its magic

1:14:44

here Let me just collapse So there you go So you have um it's

1:14:51

offering you to download but you can ask so can you show me

1:14:58

a preview of the um merged

1:15:04

file and again it will display uh this

1:15:12

and here it is a preview So you have username last name first name sections D

1:15:20

and the last uh column the column override which is here it is here and it

1:15:27

came and you can check that but it came from the second file So this is a very

1:15:33

uh amazing kind of um um cap Now one of the most

1:15:40

important things that kind of is changing data science

1:15:46

is that LLMs can keep traceability and reproducibility

1:15:52

uh if you know how to prompt it at the right way So again uh you have lots of

1:15:59

uh steps and what you can do you can you can come up with the prompts saying like

1:16:05

let's create a traceability document to make sure that others can one know what

1:16:10

data was used two uh know the analysis um how the analysis was performed and

1:16:17

three threads to validity We want a guide for someone else to be able to

1:16:23

replicate and know the limitations of this analysis And again we're doing a

1:16:29

very simple analysis right we came up to with two little um graphs

1:16:35

and we did a mo a machine learning model for something and that's it So here it it goes it creates you have to go

1:16:43

through this right the traceability document So this is the primary data set what it contains then the supplementary

1:16:51

data set what we did as an analytical procedure we did we did data cleaning um

1:16:58

we standardized uh students identifiers um we the data integration we merged

1:17:06

data sets by username replacing the overwrite column um uh we we used um in the past we did

1:17:15

compute We computed mean and distribution of course averages We counted missing values We um came up

1:17:22

with the histogram of course averages uh and a scatter plot and you know um and

1:17:28

we tried to use predicting uh predictive modeling uh predictive

1:17:35

um models Here's our threats to validity Missing data eight students lacked course averages as roles were excluded

1:17:42

model over fitten uh overwrite values So it it comes in and that

1:17:49

so uh it it comes in and asks us if it

1:17:54

wants to export as a word I would say I want to export as a readme.md

1:18:03

file And readme files are very common in software engineering where where

1:18:08

whatever work you did you're going to disclaim in this readme And MD is a markdown um style It's just a way to um

1:18:19

and and the LLM can generate that easily It's is it's a it's a markdown where you

1:18:25

can kind of show formatted text And that's what it's doing in its um with the code interpreter

1:18:34

And um

1:18:41

it's taking a little bit while but here it is Okay I created this read me I'm not going to download it yet Uh now I am

1:18:49

going to ask chat GPT uh for uh another thing that is very let's ask chat GPT

1:18:57

this thing here is uh it's not work So now for each analysis and visualization

1:19:03

create a single Python script that could perform the analysis and produce the visualization

1:19:09

The script can assume that the files that were used are in the same directory as the script Save each Python file with

1:19:16

a prefix that identifies the analysis it is related to and output a table listing

1:19:22

the analysis the file name of the Python script and a one-s sentence description

1:19:28

So the thing is every time we were analyzing the data it was generating Python code So now I want the Python

1:19:35

script to run like if you have both documents just and and you can hand to a

1:19:41

data scientist for example and say look this is my minimum viable product I want

1:19:47

you to to do this and uh this is the list of scripts generated You have the

1:19:53

histogram of courses So this is the the the script in Python that do do this

1:19:58

Then you have the scatter you have training the MLP model and so on

1:20:04

So it's an it's it's generating the package with the summary

1:20:10

uh table and it it really it it all python analysis scripts have been

1:20:16

created and saved You can now download individually or review the uh the summary table It's okay And then you

1:20:23

have all the available um Python scripts scripts scripts And now what we want to

1:20:30

do is to to create a package Um so what we

1:20:36

can um what we can ask is create a zip

1:20:42

file a package in zip file with the

1:20:50

um with the script with the with the script

1:20:58

with the summary with the script summary

1:21:04

with the data sets um and the readme

1:21:12

file And so this process uh is really is is

1:21:18

really innovative in data science because usually in data science we have sprints and we have like a focus in

1:21:26

having the code running the task done and we don't document things we forget

1:21:33

about the steps and this is a very very nice um feature that AI is bringing to

1:21:40

data science to go and prototype

1:21:46

um the past assignments using um LLM using chat GPT for example or

1:21:54

Gemini or whatever um you think it's best and compare the results go through

1:22:02

this experience of making data science AI assisted

1:22:07

Um you can do the regression models the uh you know that you use K nearest um

1:22:16

KN&N um you know you you could do all the all the the classical I would say

1:22:24

machine learning things uh uh here in the code interpreter Okay

1:22:31

but again the the knowledge of what you're doing is ve so very important

1:22:38

Now just to wrap up this part of uh having the AI to work for us in terms of

1:22:45

data science I want to talk about custom GPTs or or agents that you can um that

1:22:54

you can draft yourself that you can program yourself So this this here is

1:22:59

the standard GPT It's GPT40

1:23:04

Uh I I I am subscribed Uh and this is what it is right but again this model

1:23:13

has a cuto off date for training Uh I think if I'm not wrong it's end of uh

1:23:19

2023 Um and it's not specific it's not

1:23:25

fine-tuned for anything that I need like in in terms of answers and responses and

1:23:33

um it doesn't have the behavior like that I want So you can create your own

1:23:40

chat bots your own AI assistant uh now I'm sure that you will you you

1:23:49

can do it because you have all the knowledge necessary to go through

1:23:55

kind of crafting your own AI So what I'm going to do is I'm going to show let me

1:24:01

take here this is um if you go this is the chat tpt subscription and again I'm

1:24:08

using open AI's platform but you could use uh whatever um large language model

1:24:14

platform you think it's best and I can if I go to GPT

1:24:22

this is a library of models that of assistance that people create and you

1:24:30

can you can kind of use this assistance Uh for example if I go to edu education

1:24:36

learn English with Emily So it's it's a person that created an assistant to uh

1:24:42

help you um learn English You have math GPT prepared So but this is not from

1:24:49

OpenAI This is from users Um but you can assess them It might be good It might be

1:24:56

it might something might be there for you Okay Now you can you can come up with your

1:25:04

own assistant and the way you do it is um you can go to create you can create a

1:25:12

new uh GPT and this you will have to have a subscription and

1:25:20

I will really really this is very introductory of what the GPT can do So

1:25:26

you will create you you can give a name to for your GPT uh a short description just to if it's

1:25:33

public it will appear in that library and you can give the instructions to

1:25:38

this GPT So what is this instruction this instruction is the kind of the

1:25:44

persona prompt pattern you are and and where you're going to put the guard rails

1:25:51

You're going to this is a prompt that your GPT will follow from the start Okay

1:25:58

And um you c you can have conversation starters

1:26:04

and we have knowledge and I'll I'll talk about this just in a bit I I'll I won't

1:26:10

go through all the process I will open a a

1:26:15

assistant which is my teaching assistant It's called math t uh uh math tai Uh

1:26:22

it's my TA with AI I call it math tai And um the I will show you how I

1:26:30

configured it configured it and you can kind of try it yourself But just to show

1:26:37

show you what it can do It can do things like uh

1:26:43

hi I am I am not getting the concept of circular

1:26:52

permutation in my finite mathematics class

1:26:58

Could you please explain it to me again

1:27:05

so this is a this is a realistic thing and the AI says great question Circular permutations are like regular

1:27:11

permutations but one key difference The objects are arranged in a circle not a line This changes

1:27:19

It gives us an example and um

1:27:27

and um and it it asks after that I can also

1:27:33

show you a lecture video timestamp where this is discussed So this AI is actually

1:27:40

more it's more an I it's a more more of an agent Why is that because it can have

1:27:46

actions Uh it's it's the goal I gave the AI a goal a goal to explain me something

1:27:53

and it's going through its own kind of reasoning and the steps and and and it's

1:28:01

even connecting with actions So let let me see let let me say yes Um

1:28:07

the timestamps oops timestamps would be very good

1:28:14

and um I have to allow the the connection

1:28:20

because so what you you will see that my my assistant is talking to an endpoint

1:28:27

website here It's a it's a website that I created that has all the list of my videos the course YouTube video videos

1:28:36

and it will look the the AI will look in the transcripts of the video if it finds

1:28:42

any word related with circular permutation And I don't know why it's taking longer

1:28:48

than it uh would I'll check if it's the the website is

1:28:54

online Uh oh Oh sorry Oops

1:29:02

It it it it went to this action where it knew that it had to go

1:29:10

and uh contact this website where it has the transcripts It found it read all the

1:29:17

transcripts and it found the videos uh so that you stu the student can click

1:29:22

and go to the video Okay Um would you like help with the union

1:29:28

related problem next or should we return to circular permutation it's just because I I changed the uh the the the

1:29:34

subject to kind of have more timestamps Um

1:29:39

let's go back to circular permutation

1:29:46

So after I said let's go back the the agent

1:29:52

kind of took a step It took a step of getting me through a final exam from the

1:29:58

winter of 2025 So just just in January in the final exam there was a question

1:30:05

question A6 and the question is in how many ways can 11 bouquets of flowers can

1:30:11

be arranged in a circle or displayed most expens I mean it went step by step

1:30:16

with the student to kind of give it give the final answer and it go it still goes

1:30:23

on in a very identic way saying would you like to try computing that numerically or with Python on to confirm

1:30:32

So let's uh run a Python code to confirm

1:30:42

and again it's calling the code interpreter or the Python interpreter here and uh the result is very like it's

1:30:51

numerical right because uh but it's it's exactly uh 9 10 factorial - 2 \* 9

1:30:59

factorial Okay And then the agent still goes on with the interaction and says

1:31:05

"Would you like to try a similar problem on your own and have me check it or is it another topic you want to explore?"

1:31:12

Here are some available commands Search transcript Find where uh where a a topic

1:31:17

appears in the course videos Email help draft a message to Professor Lemon for

1:31:24

assistance Or grade scheme show how your final grade is calculated

1:31:30

Let's go on and just say email So sure Could you quickly tell me what you'd

1:31:36

like to help with uh in your uh uh in your email to Professor Lemon so I am

1:31:44

struggling with circular permutations

1:31:51

Okay Uh and let's see what the AI will do So

1:31:58

here's the here's the draft Help with circular permutations Dear Professor Leon I hope you mess I'm trying to ask

1:32:05

for some help understanding the topic of per uh circular permutation

1:32:11

I've been receiving the material examples but I'm still finding it a bit confusing

1:32:17

especially when there are additional conditions like not sitting side by side This is something that I've I've asked

1:32:23

in the beginning Um I didn't record that but if possible I'd like to appreci uh I

1:32:28

really appreciate any additional guidance resources or clarification Thank you so much So would you like me

1:32:34

to send this now or do you want to add it uh any part of it

1:32:40

i did not program the agent to send the email but let's see

1:32:48

So this is very the the AI is hallucinating This is a very good example of hallucination

1:32:55

My agent I still I'm not done with programming my agent customizing it I

1:33:01

will have them the uh the agent send the

1:33:06

uh email directly to my inbox but I didn't do that yet And again Chachi PT

1:33:14

is just predicting the next token and it's predicting something very very like it's an illusion that it pred that it

1:33:21

sent the an email It's just because from its context all of this window

1:33:27

the the like the next word would be done your email request and then you know has

1:33:33

been sent but that's not true So this is a hallucination and this is the kind of things that you

1:33:39

have to worry about Uh for example creating a gu a guard rail I will be

1:33:45

doing this uh action uh of sending the email I will be programming that If I

1:33:50

were to publish this assistant right now I would I would need to have a guard

1:33:56

rail and say "Asistant you don't send emails." Because if you say something

1:34:02

like this to the student the student will think the email was sent but it wasn't It was it wasn't Okay

1:34:09

Now let's go inside math tai and see uh how how we can configure this thing

1:34:17

So here are my instructions It's my initial prompt and it's very it's it's a lengthy

1:34:24

one Um and I will in the future uh break

1:34:29

down this assistant in several modules assistants uh several different assistants that do different things But

1:34:36

this is just to kind of start start to you to be get interested in these things

1:34:42

And it's not difficult You can do this with low code knowledge So you're a friendly approachable university level

1:34:49

mathematics professor who explain concepts in a conversational and theatic tone You help first year psychology

1:34:55

students many of whom have limited backgrounds in mathematics uh to understand course material and

1:35:02

solve exercise effectively when a student asks So here here is here here

1:35:07

are the instructions how I want the persona mafi to behave When a student

1:35:12

asks about course content just as I did with the per circular permutation explain concepts clearly and informally

1:35:20

like a relaxed but thorough chat with the student So I don't want the the assistant to come and think it's it's

1:35:27

talking to an undergrad in math They're not Look up relevant exercises from past

1:35:33

exams and tests in your knowledge base indicating the source And it did that right it indicated the source Always ask

1:35:40

if the student wants to go through the solution step by step or if they prefer to try it themselves and have you review

1:35:46

and give feedback um and proceed based on their response

1:35:51

That's exactly what my assistant did After the explanation run search

1:35:57

transcript command using the topic to find related lecture video timestamps If

1:36:03

results are found share a clickable video link with the timestamps and say here's the part of lecture we talked

1:36:09

about this If results are not found say I couldn't find this videos Here's the the course

1:36:16

playlist content to help you out and then show the content When a student ask

1:36:22

about the course structure such as the grading scheme deadlines or exam dates answer only based on the course outline

1:36:29

found in your knowledge base Include direct quotations from the outline in your response to support your answer So

1:36:36

we can test this now And at the very end of the conversation oh when providing

1:36:43

numerical or formula based solutions automatically use Python to verify the accuracy of the answer before finalizing

1:36:50

the explanation again because that 10 factorial minus something 9 factorial Uh

1:36:55

it was chat GPT actually at that point doing next token prediction So having

1:37:01

Python to verify that is is helpful And the uh and you could

1:37:08

ask for it to run Python from scratch but because this is math we we we have I

1:37:14

want some symbolic math as well to come up and not only like the calculation of

1:37:19

the the like a calculator where you have the the end number like the final number

1:37:25

and uh at the end of every conversation ask is there anything else I can help with and list the available commands and

1:37:31

here's the available commands search transcript email and grade scheme So the

1:37:36

email is here right but it's not configured yet So this is very

1:37:41

interesting because the instructions for you to manipulate and to and to create

1:37:47

your own custo custom GPT is actually natural language Again it's natural uh

1:37:55

natural language uh then you can you can put some conversation starters just so the the uh

1:38:03

the so that the student can kind of you know uh know what it's uh what it can

1:38:08

asks uh what what the student can ask and then you have knowledge So again

1:38:15

this is the this is where uh the rag

1:38:20

framework comes in and where you're putting

1:38:26

uh um you're putting the

1:38:32

the the knowledge the documents from

1:38:38

so for example PDF uh and this will be

1:38:44

sitting sitting there as a database right and the orchestrator will read this

1:38:51

database We'll find the best document that kind of fits We'll find pieces of this

1:38:58

document and then we'll kind of put this whatever it is in the document into the context window and send the prompt You

1:39:06

don't see it but it sends your prompt together with something else that was here So I

1:39:14

what I uploaded here is a content map of the course So week one we we talked about permutation week two something

1:39:20

like that Here we have the solution to fi different final exams

1:39:27

Um and this is PDFs with all the solution of the exercises and I also

1:39:33

uploaded the outline the outline for the summer course So after I uploaded this

1:39:38

knowledge base my my assistant will assess this So

1:39:45

let's ask a a question So

1:39:51

So here is the qu So So here is the question Uh I missed an online quiz What

1:39:57

should I do let's see No worries Since only the best five of out of six quizzes

1:40:04

count towards your final grade you don't need to do anything for your first miss quiz This is built-in flexibility in the

1:40:10

course design So and I remember that I asked to kind of site uh the

1:40:16

documentation to kind of um support students do not need to request academic

1:40:22

consideration for the first missed quiz Academic consideration requests will be denied for the first missed quiz So

1:40:28

you're all good for now If you end up missing more than one quiz then you'll want to submit an academic consideration

1:40:34

request for additional ones Would you like me through uh to walk you through

1:40:39

how to do that just in case and if there's anything else so that's uh

1:40:44

that's exactly what the AI is doing It's kind of following my instructions

1:40:51

Now the actions is here

1:40:56

Um and uh the the model that I'm using is

1:41:02

uh it like you can change the user will change whatever So this is the

1:41:09

foundation model of the GPT and here's the capability So the I can access a web

1:41:15

search So my agent can access the a web search So it can uh I know that it's

1:41:22

totally um not its role but I could ask who is the

1:41:29

uh 2025 Roland Garose champion

1:41:36

So it had a could date It looked

1:41:42

to the um it looked to its knowledge It doesn't

1:41:49

have anything to do with the course but I didn't put any guard rails I didn't put any guard rails that it it could not

1:41:57

um answer other things I if I want this guard rail I just come here and say here

1:42:04

are your guard rails Don't talk about uh things that are not mathematics Right

1:42:10

and it but it but its ability it's it's um it can evoke a web search action and

1:42:17

go and it it's really Carlos Okaras and that's uh that's a can do it canvas

1:42:24

means that you can generate an image so generate a a

1:42:31

probability tree so that you can use as example to

1:42:40

explain conditional probability

1:42:45

Right so now this is math and it's using the canvas which is like it will open a

1:42:52

window with a generated thing Um then you have the engine of image

1:42:59

generation and the code interpreter meaning and data analysis meaning that you want it to evoked evoke Python So

1:43:07

these are the capabilities This capabilities are actions that are

1:43:12

provided by OpenAI and they're there are standard but then there are actions that

1:43:18

you can go and you can set up yourself Now this

1:43:24

creating these actions are way more programmatic and and diff it's uh you

1:43:31

need to know coding So the the image is taking forever but it will take um it's

1:43:36

creating something but um I have two actions here One is to check the weather

1:43:44

The other one is to go and find my transcripts So if you click here I

1:43:49

created a action where I I don't need to

1:43:55

log in anything and I have a schema here that is pro programming and coding but

1:44:03

the the action is name is search transcript and what is uh what is here

1:44:11

is uh the description is list of matching timestamps and video info And

1:44:18

this will be returned to the GPT when I access this website So this is the

1:44:26

website um that when I ask this when when the AI

1:44:34

uh assesses this website the website returns give it give back or

1:44:40

communicates back to the to the GPT that

1:44:46

the a list of the timestamps It's it gives it it a list back Okay Now

1:44:53

this is not I mean this is programming you have to know a little bit of JSON of

1:44:59

uh REST APIs uh but it's it's really uh learnable and

1:45:06

we're this is not the goal of the course uh this is not an introduction anymore

1:45:12

right but um you can create all of these different actions and I will create one

1:45:19

uh I won't program it with you But I will create one that the name will be

1:45:27

um uh you know like let me incorporate a bl a blank template but the the name the

1:45:34

title will be send email Okay And what is going to happen is that

1:45:44

I'm going to have my uh Gmail for example connected to my GPT and whenever

1:45:51

a student writes an email the GPT will send that to a draft email to my to my

1:45:58

uh Gmail and we'll send it to myself Right so this is what I'm going to

1:46:04

create Um again this is more programmatic but it's not it's not that

1:46:10

difficult It's very learnable and you can even use chatt to help you create

1:46:18

these actions for your agent So let me give you an example because uh although

1:46:24

I have knowledge of this while doing this uh my um while doing math tai uh I

1:46:32

had to ask chat GPT for something and I'm going to show you how it went

1:46:39

So chat GPT let me open

1:46:45

um great So connect GPT as

1:46:56

um

1:47:01

let me see if this one

1:47:08

So this is I was designing the instructions and I I even asked Chachi

1:47:14

to help me get better Um it came with some uh uh you know like go uh and help me

1:47:23

out So I always like I always ask questions

1:47:29

about that Uh where is it oh my god Let me find it

1:47:37

here Let me search for API

1:47:55

Maybe this one Let me see So

1:48:00

I mean the I will find it here I will pause the the recording and find it But

1:48:10

um

1:48:28

yeah So uh this is this is well I mean the if you

1:48:35

ask for help look I want to uh create an action to you know to find timestamps

1:48:44

the chatpt will come up with the code obviously that knowing the code knowing

1:48:51

uh about rest APIs about JSON about Python will help you get there But you

1:48:59

can even accomplish very good stuff like just just by uh you know interacting

1:49:06

with uh Chad GPT Okay So what uh just to

1:49:11

finish here the the things one thing that I use let me go back here that is

1:49:18

very interesting and uh it's what it's part of the react framework right the

1:49:24

action of the agent uh is zapier okay so

1:49:29

zapier uh and I I didn't uh put that here but zap year

1:49:36

enables you to do automations with other software So maybe you want to when you

1:49:43

receive an email you want to send the content of the that particular email to

1:49:49

a Slack message for your posttops I don't know So what you're going to do is

1:49:55

you can have your interaction your action in get getting integrated to um

1:50:02

Gmail exactly what I'm going to do with my send email action and Zapier If we go

1:50:09

to the platform uh zap year is it's this automation

1:50:14

thing So when uh when you ask chat GPT

1:50:20

uh to send an email you know like the email will connect will interface with

1:50:27

Gmail and the content of that will pass through uh Gmail So this this is

1:50:36

everything that you need to create a custom GPT Now this is um the creation

1:50:42

of this customs GPT They are um

1:50:49

very easy very understandable But there's another more professional way It

1:50:55

doesn't doesn't let you to fine-tune for example Right now let me walk you

1:51:02

through what we call the OpenAI playground So OpenAI playground

1:51:10

Um let me log in Uh but OpenAI Playground is

1:51:20

is a a whole more professional in terms of

1:51:25

getting the models um low code It's low code as well like

1:51:31

you you if you know code it will be better but it's low code but you can create

1:51:37

assistance and if you want to create an assistant you have again like the same

1:51:42

model you have the the knowledge base you can

1:51:47

put the code interpreter there So uh continuing you can this is the

1:51:55

knowledge base This is the code interpreter you can come and put your files on You can also

1:52:03

work with the concept of the um of the

1:52:11

um where is it i forgot Well I'll oh yeah

1:52:18

I mean you can it's it's very similar but then you have a fun functions right

1:52:23

and functions is exactly the actions but here um it's you can you can use uh

1:52:31

actually describe what your function does or or your action and we will generate a definition for you So this is

1:52:37

very great um to use And here you can also kind of tweak the

1:52:44

parameters that we discussed in week four of temperature and

1:52:49

top P and how you want uh you know like your response format you can do that Um

1:52:57

here uh you know um a JSON is just a structured ob like it's it's a way to

1:53:04

structure the uh the text with uh like key and uh like uh key and and um value

1:53:14

pairs Um so it you have more options here in the

1:53:20

API playground It give you it give you it gives you more options Now just so

1:53:27

that you um know OpenAI program you have to have a

1:53:33

subscription or you have to pay for the the tokens So usually this models you

1:53:39

pay for each token that you send Uh so for example right now I have two 2408

1:53:46

tokens uh to use Um and you can play also like uh with

1:53:57

uh you can also play here with what we call vector stores This is a this is a

1:54:05

concept that is um you know it's it's a little bit more in depth but it's good for you to know and

1:54:12

you can call this I have different assistance here um but

1:54:19

you can create this assistance this assistance and deploy this assistance So

1:54:25

when you deploy if you deploy the assistance uh let me show you here If

1:54:31

you go to the math AI stuff the math tai

1:54:39

um if you if you open ma math tai

1:54:45

uh and you you you can actually when you deploy you will have a link you have a

1:54:51

link and you can share this link and or you can embed this link anywhere you

1:54:57

want in your website in your brightspace and whatever So and here's the same thing

1:55:04

Okay So here uh where am I and here's the same thing After that you

1:55:12

can export the assistant and uh you will

1:55:17

have a URL to access this assistant and then you can embed this assistant in

1:55:24

your for example um uh for the

1:55:32

um for the the general public Okay

1:55:41

into web scrap scraping Let me go back a little bit because I forgot to uh

1:55:46

mention that in OpenAI playground Um so you can access you can access uh both

1:55:54

playground and dashboard in platform.openai.com openai.com or if you

1:56:01

um if you want to access uh OpenAI playground

1:56:07

um you will see here a dashboard link right

1:56:12

so in the playground is really you playing with your assistants and you can generate your assistant uh you can

1:56:20

select which model you want right so um for example this was one of

1:56:26

the first GP GPTs uh that were available Now you can you can upload files you can

1:56:33

upload f uh you can come up with a actions as we discussed Uh there are

1:56:39

examples uh that you can use and also you can ask a GPT to help you out You

1:56:46

can enable code interpreter Well we we we all uh always kind of uh talked about

1:56:52

that uh if you want the response to be plain text or to be a JSON object JSON

1:56:59

is a structured way uh where your data is kind of uh you you

1:57:05

have pairs of informations like a dictionary and the data is um within

1:57:11

curly brackets So um the

1:57:20

this is playground is really playing but the dashboard is more elaborate and in

1:57:26

the dashboard you can actually do evaluate uh you can evaluate tasks

1:57:34

um you can uh sorry you can have task evaluations for uh your your model and

1:57:41

you will see that you can um upload files for example that have

1:57:47

your desired answer and see how the the

1:57:53

um the GPT is the LLM is doing

1:57:58

You can try uh your your models and come up with um some of the metrics that we

1:58:07

discussed So there's lots of options again this starts to be a little bit more programmatic you know um

1:58:16

and so on um so let me go and and tell

1:58:22

you one very interesting thing about fine-tuning so fine-tuning as we recall

1:58:28

is not about rag which is just com just kind of going towards documents and

1:58:35

putting that into the completion but it's actually by retra training your

1:58:40

foundation model kind of modifying the weights and um making your your your job

1:58:49

your task better right uh we talked uh week four

1:58:56

about what we called task finetuning

1:59:01

Um and but we also have now conversational fine-tuning and we we'll

1:59:08

talk a little bit bit more about that but as we discussed you can create a fine-tuning job to come up with your

1:59:15

modified foot foundation model your fine-tuned foundation model so it's a

1:59:20

supervised learning uh you can select the base model so for example I want to select chat dpt mini

1:59:30

uh here and I'm going going to add that this is the finetune

1:59:36

uh version 01 It's my finetuning uh this is you can leave it random

1:59:43

So see it is just the controls the reproducibility of the job So you can

1:59:48

leave it random or just assign a number and then you will upload a training data and the training data is again if you go

1:59:57

back to week four it's a pair pairs of um

2:00:04

these are pairs of uh um supervised

2:00:10

learning pairs like the prompt and the completion right um and the the file the

2:00:18

extension is JSONL It's it's JSON is a I will talk a little bit more about JSON

2:00:24

but JSON is a structured kind of uh document And in in particular JSON L is

2:00:31

each line of this document is a JSON is a structured uh kind of um

2:00:38

um kind of line And um and you can use validation if you want or you you you

2:00:45

know you you you can go directly without validation

2:00:50

And the batch size is the number of like the the number of

2:00:57

examples you want to train in each app epoch and uh well so and the number of epoch

2:01:04

So you can leave it this all auto you you can learn more about fine-tuning but here you will have to upload a file Let

2:01:12

me show you what the JSON L looks like So let me go here

2:01:18

Um let me first go oop sorry here

2:01:23

Um this one this is a JSON L uh format So each of

2:01:32

these lines is a JSON A JSON is again a structured kind of text where you you

2:01:38

you put as a dictionary keys and um value So for

2:01:45

example here this first JSON is what's the prompt that my LLM will receive So

2:01:52

it's an example each line is one training example So here we have just

2:01:57

two We know already from week four that we have if we want to fine-tune our our

2:02:03

model correctly we should go and use a lot and a lot and a lot of data Um and

2:02:12

um but here's just an example So prompt is

2:02:17

uh to the prompt will be translate the following text uh English sentence into

2:02:22

French Hello how are you and then comma So you're going to when you you see a

2:02:29

comma it means that this first key value pair that prompt and and whatever you

2:02:34

want to be the prompt is done And then you're going to give uh another

2:02:40

uh key value which is the completion and then the value of the completion of what you want the completion to be

2:02:48

Um and and you can you know like you can train

2:02:53

the uh the LLM in multitasks right so in this here I'm I'm I I have one example

2:03:01

of translation and you can see that I'm directly just

2:03:06

translating I'm not telling the LLM to say oh hi of course I will help you uh

2:03:12

the translation is so if I have several examples of this completion this more

2:03:17

simple direct completion the LLM will will learn

2:03:24

will adjust its weights to to act like this so it will be very very um if you

2:03:30

do a a very good fine-tuning it will be very rare to have a verbose kind of

2:03:36

response okay but I also can try I can um I can and also try to fine-tune into

2:03:45

several um um tasks So I can have lots of data in

2:03:50

the same JSONL and here goes summarize the following text Artificial

2:03:56

intelligence is a field of blah blah blah I'm just kind of um this is just dot dot dot because it's too it can be a

2:04:02

big prompt There's no problem uh given that it's within the limits But then the

2:04:09

completion is AI is a field of computer science that focuses on creating intelligent machines So you can see that

2:04:15

if I give the AI the simple oneline examples and I train train this AI um

2:04:23

like fine-tune and and go uh you know uh doing the supervised learning on

2:04:30

the target the completion target this this this LLM will respond will answer

2:04:37

me with this kind of style okay so

2:04:43

fine fine-tuning is really more about guardrailing and making this tile and the and the task or how to answer a task

2:04:51

um and and giving examples right because sometimes even though the LLM is

2:04:58

answering the the right way it's kind of for example with sentiment analysis it's

2:05:04

getting the answer wrong So having examples that have the the content like

2:05:09

the what is the sentiment plus the way the the uh the LLM should should uh act

2:05:18

Okay Now uh this is what we call the task finetuning There are other there is

2:05:25

another type of fine-tuning that we did not um cover in week four and I'm going

2:05:30

to really it's it's just because it's a little bit more um

2:05:35

intricate but it's something called um it's something called conversational

2:05:40

finetuning So here's an example of a conversational fine-tuning and the um let me see if I can uh find

2:05:49

the um

2:05:54

here is so so the format for conversational fine-tunings is this this

2:05:59

this whole line here it's it's broken in three but but you can see that it's one

2:06:05

in piece of information we call it this one line because it's in between this curly brack brackets This is one single

2:06:13

piece of information or what we call one line But the thing is that you have the

2:06:18

message So here is your first you this is your key This the message is contained in this brackets here

2:06:26

right so this is a conversation and what I'm this is I will have examples of a

2:06:32

structured conversation and first of all I will I will say what the role if it's the system or if it's

2:06:38

the user and the content of the system message then if it's the user what's the

2:06:45

content of the user message and then um what's if the if it's the assistant

2:06:51

what's the constant the content of the assistant response So let me show you this in practice

2:06:58

So let's say that you want to um you want a LLM to be sarcastic You can try

2:07:06

to use the persona prompting or instructions and say from now on you're going to act with a as a sarcastic

2:07:13

person And that's fine and it will give you a result but it might be that it

2:07:18

won't be the sarcasm you want because again the LLM was trained in whatever

2:07:24

data was found out there You want like Marv uh a chatbot to be sarcastic but um

2:07:32

you want to remove you know um you want to give data that makes a sarcasm but

2:07:38

without like for example um any kinds of um

2:07:44

violent wordings and everything So you want to give examples of what you want

2:07:50

So you can come up with like 500 uh lines and here is what we are going

2:07:56

to do So this is the conversation the role of this um the the first thing

2:08:03

is I'm going to tell the like the fine-tuning task that the system

2:08:10

is is this is a factual chatbot that is also sarcastic So if I ask the LLM after

2:08:17

training to be sarcastic um or if I use exactly this wording it

2:08:24

will take advantage of this training Okay Now

2:08:30

if given that the system is sarcastic the role of the user when I'm the user

2:08:36

prompts me with the content named what is the capital of France the assistant

2:08:42

now the role of the assistant will give me a answer of Paris as if everyone

2:08:47

doesn't know that already okay and you can come up with all these examples in a

2:08:53

conversational way and really to try to make your uh

2:09:01

fine-tuning um in and you can use this here in uh I

2:09:11

closed I closed the open AI playground

2:09:17

or the what we saw it's platform that has both of the playground and the

2:09:22

dashboard but what you see is you can create the job then you will you create

2:09:29

and you will start the job and after that you will have your model it will be cheap GPT mini uh FO um fine fine 01 FO1

2:09:42

because it's the fine-tuned version 01 and that's a is it that's a new model okay

2:09:48

all right so now let's go back and let's talk about web scrape what happens with

2:09:54

big documents right for example you pages of pages on policies and you want

2:10:00

that to configure the knowledge base of your agent So the the the way we should

2:10:07

go with it is creating maps or breaking it down So

2:10:15

first of all you could break down the document into individual pieces right

2:10:20

like uh let's say that you have pages of pages of a policy of an insurance policy

2:10:27

where the first part talks about a type of infraction and the other part talks

2:10:33

about another So you would break down this into individual pieces you could do that yourself um

2:10:41

and and and put this individual individual pieces to the AI because

2:10:46

there is a limit of space uh in the context window and also in the knowledge

2:10:52

base right um uh actually like but the knowledge base could be very very big

2:10:58

but when the AI is kind of assessing the knowledge base kind of reading from a PDF or reading from an Excel file that

2:11:05

is too too too large um like millions of rows it won't be able to put all of that

2:11:12

content into the context window and you will miss information So

2:11:18

one way is to tell it's one one good way to do it is tell the agent to open each

2:11:24

page manually for example if you're working with a PDF and summarize what is

2:11:31

on each page and slide so that you create a map you you're building an

2:11:38

index and page by page you're what is

2:11:44

going to the context window is so it it it will open the the file it will access

2:11:52

access only the first page which is enough then it will create a summary and say page one says this

2:11:59

and what is going to stay on your context window in your history in your memory is that right like open this page

2:12:08

and the summary So when you ask questions the AI will kind of go to the map and

2:12:15

see where it should look for the answer right so this is something that is uh

2:12:20

it's a very um it's kind of a prompt engineering way to bypass all of the

2:12:27

things So and be aware that outputs are also limited So the the the window where

2:12:34

you output the the number of tokens is also limited and

2:12:41

it is so very important that you know that and

2:12:47

usually we we uh you will understand that it's uh limited because the the AI

2:12:55

is kind of writing you a response an answer and then it's kind of breaking into the middle or sometimes even the AI

2:13:02

I tell you look this is my limit and the output is there and so you can say

2:13:09

proceed with what what you're doing so that the and you can even remember the AI what

2:13:16

the AI was doing but proceed with the next uh with like with the next output

2:13:22

window proceed and continue doing what you're doing because

2:13:27

um now it's a new output window and it it can't continue so you will have to break down um the AI will will kind of

2:13:34

force you to break down the answer if the answer is too long Okay

2:13:42

so this is very important very uh interesting uh to when you're working with larger

2:13:48

documents This also happens when you're scraping So it is very important to talk

2:13:56

about web scraping or document scraping um when we're talking about agents and

2:14:04

large language models and um agentic agentic AI and react talk about

2:14:12

web scraping So web scraping is the process of automatically

2:14:18

retrieving or copying information from the web uh say that you have all the

2:14:24

regulations for insurance and policies for insurance in the province of Ontario online and um

2:14:32

one thing you could do it is just print the PDF of each one of them It's tedious

2:14:38

but using coding you could do that Um and then try to um

2:14:46

uh try to upload this to your rag like to your files to your knowledge base But

2:14:54

the thing is that if this if this policies are very very very very very

2:14:59

large it is difficult to deal with it the AI won't have enough

2:15:07

memory to kind of um read all the file

2:15:13

or read several files um and do what you want So let's say that you're prompting the AI to see if a

2:15:22

uh insure policy is within is complying with the

2:15:29

provincial guidelines And so you would have your assistant to

2:15:34

look in all of these web pages amongst all of these regulations

2:15:41

and try to say if it's compliant or not compliant but you know first of all you

2:15:46

would have to download PDFs you would have to um upload this PDFs but in the

2:15:52

end it might not work because you need because of the memory like um um chat

2:16:00

GPT if I'm not wrong it can accept 128,000 token tokens as inputs

2:16:08

and um and some documents go way way uh more

2:16:15

than that So first of all um the the the

2:16:20

thing to do is to do a web scraping So web scraping extracts the full text of

2:16:27

that automatically So you don't need to download the PDFs Um and what you can do is that what what

2:16:36

is done the state-of-the-art is to break this text that you just scraped into

2:16:44

chunks Um so you break down into little chunks that make sense So paragraphs uh

2:16:52

or or sections and this will be uh so you're breaking one

2:17:00

for example one policy in several different several little pieces uh of

2:17:06

that policy and um what what we use a

2:17:12

lot is the lang chain or llama index to

2:17:18

vectorize this chunks Meaning that when the when the user prompt

2:17:24

for uh this lookup in the policies

2:17:30

uh these tools there are programmatic tools they will find the the the vac the

2:17:37

vectors or the the like the the the the piece this chunk that has to do that has

2:17:45

more similarity with your prompt or that has to do with your prompt So

2:17:52

you can do this is a little bit more advanced but it's um it's you can

2:17:58

implement this using the open AI um

2:18:03

uh uh playground together with Zapier for example or together with lang

2:18:10

um framework and um

2:18:16

so for example you could say a c a customer was told their claim was denied because their car is red Is this

2:18:22

compliant right and then um what you could do for example is you have your

2:18:30

your your prompt is in a chatbot and your chatbot for example calls uh action

2:18:37

in Zapier that will look at these little chunks of documents and you will have lots of chunks of documents there

2:18:45

um and it will it will it will result in the more the most important chunk of of

2:18:53

of document and that is going to be put in a rag like in the in augment

2:18:59

augmentation of the context window Okay Um so the the web um

2:19:09

the the the the web kind of um

2:19:14

um let me see if I want to say something

2:19:20

more No that's fine So there are several

2:19:25

um ways to do the scraping You can ask chatt to come up with a python code for

2:19:32

you and you can just use that um and give give an excel file to the chat to

2:19:39

chatpt so it can read the rows of the all the URLs you want to scrape and then

2:19:45

save it in a plain text document or something like that Um but you can also

2:19:50

use their thirdparty um

2:19:55

um tools right so for example browse AI is

2:20:00

one of them um but if you are already using Zapier

2:20:06

to kind of do your automations Zapier has a web scraping embedded into into it

2:20:17

Okay So now I want to discuss a little bit more about the impacts and challenges of

2:20:23

generative AI In the past week uh we defined generative AI We saw several

2:20:30

details on how it works and what applications it can tackle Um we saw some challenges such as

2:20:37

hallucinations such as uh outofdate information used in the

2:20:44

training set of generative AI But now I want to I want to extend this a little

2:20:50

bit to to the to the scene of um

2:20:56

businesses um and operations inside businesses So

2:21:02

first of all um we can talk about competitive advantage

2:21:08

So right now uh businesses are all uh they all know that they will have to

2:21:17

steer its um capabilities

2:21:22

to to incorporate AI um into its operations and make use of

2:21:31

AI to be more efficient reliable and so on Um the thing is that today you have

2:21:39

foundation models um that are open source some of them are

2:21:45

uh proprietary uh and but if you want for example to uh

2:21:51

to um to pre-train your own model or if you

2:21:57

want to fine full fine-tune this this models uh or even if you're doing a path path

2:22:05

So that is kind of a finetuning that uh demands less computational burden

2:22:12

It's you will still have to have large amounts of data

2:22:18

and so businesses that have this this highly large amount of data

2:22:27

they are in a competitive advantage Another competitive advantage I would

2:22:34

say is having uh skilled um workers in the domain of AI

2:22:43

Um there are some businesses that have to build this from scratch because they're so distant

2:22:51

um from this technological like um

2:23:00

breaking point Um but the and there there are other

2:23:07

companies and businesses that are closer to that They're more taxsavvy

2:23:12

um and they can manage to kind of create

2:23:17

this capabilities and incorporate AI quick and in a in a faster way But those

2:23:24

that um those that struggle in having this uh this workforce that is uh

2:23:34

specialized um in in data at least um

2:23:40

will be in a competitive disadvantage

2:23:45

Um the but we have to talk again about

2:23:50

limitations of AI we know that this businesses should incorporate AI in its

2:23:57

um operations Uh and but we we saw that the system may come

2:24:04

up with sentences that looks plausible It's not necessarily a factbased answer to a question So we

2:24:12

need to be aware that even if this business is incorporate AI

2:24:17

they need to to be uh uh they need to be aware that

2:24:28

you you are investing but you still can have an inability to solve all of the of

2:24:36

the uh problems and you have to create the culture of having a human question the answers

2:24:45

of the AI right um there are limitations as well that

2:24:52

can uh can be a problem to small businesses uh or businesses that have

2:25:00

constraints um financial constraints for example

2:25:07

because they these models they are expensive to

2:25:12

train even to fine-tune although finetuning is a little bit better

2:25:18

um they they are not they cannot be trained on a continuous basis So you you

2:25:26

will have to train and teach your workforce

2:25:34

on prompting engineering capabilities for example or rag strategies to make

2:25:41

your models um better right um

2:25:48

and obviously yeah again we we always like land on the data quality even if

2:25:57

you have lots of amount of data you have the capability you are aware of everything and all the limitations

2:26:04

it might be that you have this data and this data is not of a good good quality So you have biases in inside of the data

2:26:13

you have factual errors inside of that data and and you your your large

2:26:20

language model your generative AI will kind of reproduce all this this errors

2:26:26

Okay Um because this the systems are so having a

2:26:36

uh uh having this awareness of the importance of data Everything starts

2:26:42

with the data If you have good quality data your models will be much better Um

2:26:50

and you you need to have this capability the skills of

2:26:56

getting your data training your um preparing your data

2:27:01

assuring qual quality for your data um and the best way possible

2:27:09

Another thing that we it's it's good to discuss is the is that uh this the models are huge deep learning models So

2:27:17

there are neural networks immense neural networks that has layers upon layers and upon layers and billions or trillions of

2:27:24

parameters So the decisions the actions or even the generation

2:27:31

that these models can come up with are they are um they have a lack of

2:27:38

transparency um um it's it's it's difficult to know

2:27:43

how the model came about even if you use strategies to like chain of uh reasoning

2:27:51

chain of thought Um but uh sometimes you have lack of

2:27:58

transparency Sometimes um because the large language model is

2:28:04

predicting the next token it cannot site the sources you know like oh um this

2:28:11

this paragraph is extracted from here It's it's it's not because it's just one

2:28:18

token at a time Sometimes nowadays they um with the APIs large language models

2:28:26

can go and do a a consultation to a data base to a wiki page or whatever Um

2:28:34

but still the that that paragraph won't be extracted in in in whole by uh by a

2:28:42

website the LLM can point to an a website that um it used for uh for more

2:28:51

information or for context Um and there's another problem that uh

2:28:58

is plagiarism Um so LLMs can

2:29:03

copy styles uh or uh for example writers

2:29:09

several uh book writer styles and everything So um there there are

2:29:16

problems that you have to be aware um of plagiarism

2:29:25

Um so what what we see and as now we understand generative AI better of what

2:29:33

it is what it's doing and how it accomplishes it uh we can see that the

2:29:38

potential impact of large language models or more in complete automation

2:29:46

um it might mean this as a human displacement

2:29:52

in some specific activities right but in

2:29:57

general we we I think we can see that AI

2:30:03

is actually supporting human task performancing right so um

2:30:10

for the automation tasks for for automation things that are very repetitive and everything um I think

2:30:17

that uh some displacement can uh can be

2:30:24

um can happen um

2:30:29

in in um but in the end AI is supporting and

2:30:36

helping humans to perform better Okay

2:30:41

So here are some uh here I I took this from a course but I have all the

2:30:47

citations after and references This are several studies uh that showed that uh

2:30:54

for example large language models uh are very creative and can help humans

2:31:01

uh to do creative tasks right uh they they even compared the

2:31:08

creativity of large language models with creativity of humans right so for example Torrance tests of creativity um

2:31:16

tested Chat TPT versus 2,718 students and the results are that 99

2:31:24

that uh the Chat TPT was in the 99th percentile of originality and fluency Um

2:31:32

other tests were made right so um the results is that AI generated two to

2:31:38

three times more ideas than humans and it scored better uh in this ideas in the

2:31:45

innovative sense of this ideas in 91% of humans

2:31:50

and um uh there there was another uh study that

2:31:56

uh that compared AI generated hus these are uh Japanese short uh poems poems and

2:32:05

um and compared this poems with Japanese

2:32:11

poetry masters and um

2:32:16

and and and the AI scored very similar to the the poetry masters

2:32:23

and uh and what what actually happened what actually scored the best was having

2:32:30

an AI and a human working together collabor elaborating and then it showed

2:32:36

to have it it it um it had the the highest score

2:32:44

Um there are other studies for example a

2:32:50

study in science says that uh the the

2:32:56

humans with AI assistance experienced a very significant reduction in the time

2:33:01

to complete uh tasks that they were uh assigned for and AI helped increase

2:33:08

their average output quality um and participants with poor initial

2:33:15

writing scores benefited the most So um

2:33:20

they are they were supposed to have some writing tasks and we can see that AI is

2:33:25

coming um it's coming in handy for those

2:33:30

who lack the initial skills for a a task

2:33:37

And there was a there was a um a study as well uh regarding software development

2:33:45

uh with GitHub copilot which is a assistance assistant that um helps you while you're

2:33:52

coding giving you uh generating ideas generating snippets of code to help you

2:33:58

um and the completion times reduced uh obviously in the group that used AI to

2:34:05

kind of code a task And um they they they completed the task 55%

2:34:13

faster than the other groups and uh and and but again the developers with less

2:34:19

programming experience uh experienced the greatest lift in productivity

2:34:26

So one thing that we notice and that we can uh expect from AI is that the the

2:34:33

basic threshold of productivity we we're going to raise the bar of this minimum

2:34:40

threshold in pro uh in productivity And finally this is the last study There

2:34:47

was a uh there was a a a paper published from Harvard Warson MBCG study about

2:34:55

consulting tasks So consulting consultants that had access to AI system

2:35:02

systems finished more tasks on average completed those tasks faster and

2:35:09

received higher quality scores for their tasks So this is uh a very brief discussion on

2:35:18

all the impacts challenges and and the um

2:35:24

and actually the expectations that we have for AI in the future within the

2:35:30

business's uh environment Let's try to discuss uh MLOps So what is

2:35:37

MLOps so MLOps

2:35:43

um is a is is the application of DevOps

2:35:50

practices and principles to machine learning workflow So um if you don't

2:35:56

know about DevOps it's a it's a framework in in in the um

2:36:03

a set of practices principles and mechanisms

2:36:10

to that guide the development and operations of

2:36:16

software So the um MLOps is actually

2:36:23

this this applied to the machine learning workflow Um so what are the

2:36:29

goals the goals is to uh to do faster experimentation and model development

2:36:36

to have faster deployment of updated models into production and also to

2:36:45

try to guarantee quality assurance So I have some notes here I'll try to

2:36:50

navigate through them Um but first of all the first step of the

2:36:58

uh life cycle of uh machine learning is to have data is to acquire data clean

2:37:05

the data and make sure that you have a um and make sure you have quality in

2:37:14

your attributes So MLOps comes in handy here um because

2:37:21

it draws a lineage uh of source versions of your of your

2:37:27

data source versions Uh you can see that you can get really

2:37:32

overwhelmed by um different tasks and experimentations

2:37:38

while cleaning your data um and even kind of having different uh

2:37:45

versions of your data Um and that could can can really get

2:37:51

overwhelming As a second step we have feature extraction So you have to

2:37:58

extract and engineer new features You have to select what kind of algorithm you are using

2:38:04

You have to to tune hyperparameters And again this is all experimentation

2:38:10

experimentation and MLOps um uh can track metrics and different

2:38:18

experiments can um can release V um can can um track

2:38:25

versions and this is very important because this is a experimentation and even if your

2:38:33

model is not performing well and you're not you're not getting better you're

2:38:39

just um you're kind of still experimenting things are not

2:38:45

working that well this experimentations they will inform featur future

2:38:50

experimentations and what kinds of features you should change So having

2:38:55

this track is very important Um and uh this is is um is very it's

2:39:05

it's very important because you can create uh repositories

2:39:11

of um of your data set of of the the new

2:39:18

data sets you're creating after your your feature extraction or feature selection

2:39:24

Um and you can go back to that repository uh if some modifications

2:39:31

happens For example if you have two um uh two pe two people working on that

2:39:39

at the same time and one changes uh changes the data set or changes the

2:39:45

model code or some kind of hyperparameter you can go back to the versions you can

2:39:50

see exactly where the the changes were made So it's what we call CICD is the

2:39:56

control version Um uh it's a version control Okay Um then

2:40:05

uh this this what I just described is kind of the ML pipeline So you have this

2:40:11

checkpoints and you have this source this control um this version control and

2:40:18

this enables you to to release versions for validation right for staged testings

2:40:27

um as well So um

2:40:33

and dur during validations you can come up with different models model alpha uh

2:40:40

u versions version alpha beta for different kinds of models and

2:40:45

automatically track the metrics compare them and also assess fairness aspects of

2:40:52

the of the results in the model and we will talk about fairness in a bit Um and

2:40:58

you always can have the human in the loop right uh you can uh also talk about

2:41:04

differential privacy and important aspects So all of this organizes the the the

2:41:13

machine learning workflow Uh it automates wherever it can And then

2:41:20

finally when you deploy uh your machine learning uh solution you

2:41:27

can also you you uh you should also look to model drifts and data drifts So

2:41:34

drifts are con a concept that is very important in DevOps where is

2:41:41

small changes to your data or to your model or to your um assumptions of of of

2:41:51

your problem can make your your solution drift a uh off of what it should be So

2:42:00

you have to assess this drifts to make sure the solution is still very

2:42:06

good Um uh for example think about data If you if if you are uh trying to find

2:42:14

throatent uh credit card transactions and something happened that you just um

2:42:23

you just uh uh your your your your company just changed the like the

2:42:29

definition of what is a fraudulent uh transaction So your your

2:42:36

even if your data does not drift but your concept drift and and therefore the

2:42:42

the data has to be reassessed features will have to be uh reselected and this

2:42:50

this this model monitoring can trigger a new cycle

2:42:57

of the development and operation of this machine learning solution So there is a

2:43:02

very um it's a it's a very uh famous image that

2:43:08

kind of uh depicts the MLOp cycle So uh you you you start here with uh

2:43:16

with data analysis um and data preparation

2:43:25

um then the development of the model and the solution per se you can you can put

2:43:32

into this bucket as as well as developing uh applications on top of

2:43:37

your models um graphical user interfaces APIs and so

2:43:43

on and um then you can train if this is the first

2:43:49

cycle or retrain if it's not if it's the other cycles uh and then you can uh review that kind

2:43:58

of validate deploy Then um you you will have your

2:44:04

inferences done and then you will monitor for any drift If that happens

2:44:09

you trigger again So the MLOps are a again a set of practices

2:44:17

uh principles automations inside the industry where uh you have to engage in

2:44:24

this process you have your you have to have all your IT team and development

2:44:29

data scientists um working to

2:44:35

engage in this and you can automate do automations within all of this cycle

2:44:41

obviously having checkpoints of humans in the loop but this creates a very

2:44:50

uh a very good framework for having this three goals here faster experiment

2:44:55

mentation model development faster deployment of updated models into production and quality assurance

2:45:03

So now I want to discuss fairness and ethics Um it's it's actually part of

2:45:09

what we call a bigger area of study called responsible AI Right so

2:45:17

AI um AI can be simple if you use simple

2:45:22

models and AI can be very very intricate

2:45:28

and it's even called to be a black box when it's using

2:45:34

um more elaborate and complex models and uh there there there are lots of

2:45:42

questions some questions like whether This blackbox AI must be used in high

2:45:47

risks and um impacting areas such as medicine and infrastructure right uh

2:45:54

because for example with deep neural networks which is state-of-the-art for for for models and also for large

2:46:01

language models they are really bad black boxes and it's hard to interpret

2:46:07

like where the result is coming from or in which parts of the network the

2:46:13

learning is made Right um so the the

2:46:20

for example we know that the medical community is a little bit um uh unlikely

2:46:27

to accept uh uh to adopt

2:46:37

uh to adopt AI solutions because they are unlikely to

2:46:46

put their judgment aside right they they don't want a an AI to replace

2:46:53

their judgment fully So they want to be in the loop right at least they want to

2:46:59

be in the loop Um it it is a it and it is a very uh trending

2:47:07

uh uh preoccupation uh and for example the the US uh FDA

2:47:16

provided recommendations in what they call a software as a medical uh device

2:47:23

uh action plan to ensure that this algorithms comply with some um with some

2:47:33

characteristics of responsible AI uh and as this characteristics uh I will point

2:47:41

out to explainability and also to

2:47:47

fairness Okay And um

2:47:56

explainability is the it's like um if there it it's a

2:48:02

um set of techniques that uses um statistical tools even

2:48:09

other machine learning tools uh to to to

2:48:15

design and also during inference and deployment of this of AI systems

2:48:22

uh that you where you can visualize access uh and explain results right by trying

2:48:31

to interpret and uh the results and making the the AI more trustworthy

2:48:38

right so um it it tries to transform the black box

2:48:46

uh algorithm in uh not in a system not necessarily into a what we call a white

2:48:52

box but in in a system where if you question you have some interpretable

2:49:00

answer and um and sources and you know

2:49:05

where the reasoning is coming from Okay Um

2:49:11

so obviously that complying with explanability is made much more easier

2:49:16

when you're using algorithms that create human interpretable models such as what

2:49:22

we discussed uh in in previous weeks decision trees linear regression logistic regression it's easy and

2:49:30

they're not the the number of parameters are are reduced But for example with

2:49:38

deep learning these interpretations is uh they surpass human

2:49:44

capability for inter interpreting these things right so

2:49:51

and obviously nowadays we are inclined to use more of this deep representations because we we uh it's a trade-off we

2:50:00

have a a more complex model but we also uh we we can um be less specific in the

2:50:08

feature engineering and feature extraction these deep representations they're very good in uh taking several

2:50:17

features and just learning from them in a in a raw sense right as we've discussed

2:50:23

and um um that's why this is this AI

2:50:29

explanability uh field is trending a lot the other one

2:50:36

is fairness right so AR fairness is

2:50:42

uh it's a paradigm regarding the prevention and mitigation of bias

2:50:48

uh which can be reflected um like actually it can be ampl so AI

2:50:56

can reflect or amplify trends of society that uh that are

2:51:04

actually embedded in the learning data right so there uh this biases for

2:51:09

example are tru troubling in medicine because you can uh lead to unequal

2:51:17

uh unequal um uh um unequal diagnosis unequal access

2:51:26

to care for example and um

2:51:33

in in there's a work uh I'll get the references from uh Dr Kalentari from

2:51:40

York University She shows that AI systems diagnosing X-ray chest uh have a

2:51:48

lower performance on underserved populations such as Hispanic female

2:51:53

patients So and that's something that is in the data and the AI is reflecting

2:51:59

that bias Okay Um

2:52:05

so fairness has become um a a a worry

2:52:11

for not only uh protectioners of AI and

2:52:16

in data science but also by reg regulatory and and um um

2:52:25

um and and broader like uh policym

2:52:31

For example the the uh general data protection regulation of euro of the

2:52:36

European Union mandates algorithms transparency

2:52:41

and um they they published a report against the discrimination from biased

2:52:47

AI So uh it's it's a way to try to to bring accountability to the the parts

2:52:54

involved in this solutions in this AI based solution systems uh if biases and

2:53:01

um harm is made So uh

2:53:09

so sometimes uh uh

2:53:17

so sometimes uh achieving fair representation can

2:53:22

uh can be obtained in the data set like creating a data a data set where you assess for bias

2:53:29

um but the so the the explanability compliance is difficult to be achieved

2:53:36

in this large model So uh we want to have both right as much as we can Um

2:53:47

the uh this fusion between explainable and fairness is this area of responsible AI

2:53:55

right um and obviously that either the explanability and the fairness involve

2:54:03

ethics aspects right you have to have eth ethics aspects

2:54:10

either trying to build an explainable model and

2:54:16

uh having fairness uh compliance in your model Okay Uh so

2:54:25

this is this is trending these are trending topics and if you go back to the um

2:54:34

if you go back to the slide on DevOps you can see that during the the deploy

2:54:43

inference and review So so this is the the validation deployment and uh you

2:54:50

you're doing your inferences you're having results and you're monitoring You have also to monitor for these aspects

2:54:57

for fairness and for um explanability

2:55:03

So uh uh if you detect something that is

2:55:09

unethical or or something that is uh hindering this this uh concepts you you

2:55:18

would uh loop again and uh try to make it better The explanability portion

2:55:25

comes uh comes very much in the in the development of the model So choosing

2:55:31

easier models when not possible Uh trying to to make your part your your

2:55:41

model the the best that you can in terms of

2:55:46

explanability and we will discuss this in terms of generative AI in just a bit

2:55:52

So for generative AI we have three we have three problems

2:55:58

that are very big toxicity intellectual propert uh property and hallucinations

2:56:05

So this is these are the three challenges like I would say the greatest one So toxicity is something that the

2:56:13

LLM returns responses that can be potentially harmful or discriminatory

2:56:18

towards protected groups or protected attributes And it it can because it's

2:56:24

learning from the internet It's learning from language And one way to mitigate this right is to curate the training

2:56:33

data Again the training data is very important to not carry social bias

2:56:39

And um so having guard rails and and and

2:56:45

um assessing your data for biases is

2:56:51

very important There are lots of metrics to calculate bias Uh for example in this

2:56:58

X-ray classification it's a deep learning network It's not an LLM This it's just a deep learning network that

2:57:04

is learning from X-ray images and doing making a prediction of whether the person is uh sick or not and uh there is

2:57:13

a way to kind of assess if this data set has biases So this is the same for

2:57:20

generative AI The training data must be um if you're fine-tuning for example

2:57:27

your prompt completion pairs should be uh should have no no biases

2:57:34

So fine-tune guard rails like the human in the loop or the RHF

2:57:40

um fine-tuning to kind of make this content uh better And by having humans

2:57:48

in the loop you're annotating the whatever is uh in in uh closer to what

2:57:56

human uh likes right and preferences So having a diverse group of human uh of

2:58:03

humans in the loop is very important So diversity of the workforce

2:58:10

in shaping in modeling the AI in developing the AI is important to

2:58:16

mitigate this biases the these existing uh the toxicity and the existing biases

2:58:23

uh even for future data right because this generated data will be used as well

2:58:29

in the future whatever contents are being generated will be used in the future to to train new models

2:58:36

The other thing is about hallucinations uh as we understand it's it's just an incorrect content although it seems very

2:58:43

nice and um first of all how to mitigate that well uh and and and hallucinations

2:58:51

can be very very irresponsible and and have biases into it So first of

2:58:58

all educate users of how generative AI works That's what we're doing here add disclaimers when possible that look I'm

2:59:05

generating an answer but this is highly um uh like you ha you have to know that

2:59:12

this is just a a uh uh probabilistic machine Um other things that are coming

2:59:21

up with the APIs and rag and and other types of technologies is to augment the

2:59:29

LLMs with the with databases and with verified databases So not just web pages

2:59:37

but uh scientific papers um

2:59:42

and things like that And uh um and

2:59:47

obviously the more you define your uh use case right the more you're able to

2:59:53

like streamline if you're having an elucination or not So uh the the third

3:00:00

problem So one two three is that um

3:00:05

intellectual property right so there there's there are some some properties

3:00:11

first of all that uh make sure that people are um

3:00:19

not direct copying chunks of of of uh of

3:00:25

text that are equal to to uh uh books or to intellectual

3:00:32

property or the LLM learned from And uh the the way that things are

3:00:39

evolving is to um to have a mix of you know legal

3:00:45

mechanisms like uh like having copyright protection and not using uh this this

3:00:53

material for training Um but you we also have machine unlearning where this the

3:01:00

chunks of copyrighted uh of copyrighted um

3:01:07

material gets uh we just kind of eliminate it the the ability of the machine learning um to learn that we we

3:01:15

do an unlearning process and you can block right uh if you identify that this

3:01:22

is too similar for some of some materials you should just block the generation and this could be embedded in

3:01:30

the models per se The big tech companies can uh can implement and are

3:01:35

implementing some of this filtering mechanisms So they uh the generation is

3:01:41

compared to copyright uh documents and then um it's it's uh it's it it will it

3:01:49

will take an action of blocking or changing uh whatever generation is doing

3:01:55

So in the end of the day to responsibly build uh generative AI models in

3:02:01

specific it's to you know like you have to define your use case the the more

3:02:06

specific the better because you can detect hallucinations better Uh evaluate

3:02:12

which is what are the risks in terms of fairness uh and um ethics

3:02:18

uh in in in terms of explanability It's hard It's still very hard to have

3:02:24

explainable tools for for I mean you have already but it's large language

3:02:30

models are very non not explanatory um although they can output the

3:02:36

reasoning which is good right so um other machine learning methods

3:02:42

you can use several explainable AI tools we have several explainable AI tools

3:02:48

such as Sharley values and I'll leave is here as a chaplet values and so on Um

3:02:57

but for LLMs you you cannot you cannot in um um

3:03:05

you it's it's difficult to use this kinds of tools but you have the reasoning if you have prompt a good

3:03:12

prompting prompt engineering uh you can kind of go step by step and better make

3:03:18

sense of what the LLM is uh the reasoning of the LLM

3:03:24

So uh then uh evaluate the performance So evaluation is very important and

3:03:30

extensive evaluation to see if any uh toxicity intellectual pro property or

3:03:37

hallucinations come in and you you have to iterate this over the life cycle So

3:03:43

uh fairness and and and ethics and um

3:03:49

trying to do the most explainable model is something that you should iterate every time in your um ML life cycle or

3:03:58

your generative AI about some of the explainable AI tools These tools are also incorporated into MLOps

3:04:06

uh not only to uh accomplish more um

3:04:11

trustworthy models uh and solutions and kind of incorporate trust to the systems

3:04:18

but also to in the stage of monitoring the deployment and the results of the model

3:04:25

uh and the solution that we can uh anticipate any kind of

3:04:30

drift from in performance right so any kind of uh decreased performance of the

3:04:37

model because of data or because of change in concepts or because of uh whatever reason it is So the uh again if

3:04:46

if you have more interpretable um you have models that are naturally more interpretive interpretable for

3:04:53

example linear regression decision trees you can read just read and and understand what you're doing and but you

3:05:00

you can um you can scale this to models that have

3:05:07

uh you know that can be deeper more intricate with billions of parameters

3:05:13

our deep representations but they lose interpretability Okay

3:05:18

So uh the area of explainable AI that helps us find biases

3:05:26

helps us find important relationships and helps helps us find this drifts

3:05:32

Um it it it it it derived explanations for machine

3:05:39

learning models that are that are machine learning models that are not

3:05:44

originally interpret interpretable Okay Uh there

3:05:50

you can have what we call blackbox approaches or white box uh approaches So

3:05:56

blackbox approaches is uh black box approaches are the the approaches where

3:06:02

you you just have the input and the output You don't know nothing about your

3:06:07

model And white box approach is you have the input you have the output So you have

3:06:13

the function that uh you're trying to interpret but you also have more details

3:06:20

about the model and how the model works And um not only you have these two

3:06:25

approaches you can have um explainable AI algorithms and tools that can applies

3:06:32

that can apply to all models You can apply to random forests to support vector machines to neural network to

3:06:38

LLMs or model specifics

3:06:44

Uh there are tools that can uh

3:06:49

uh can handle different data types as inputs Not all of them handle all of the data

3:06:57

types and uh they defer as explan the

3:07:02

the the explanation the scope So here is the uh like the algorithm per se of

3:07:08

these tools For example you have global um explainable algorithms where you you

3:07:15

you're going to explain the whole model um or you you you have a local

3:07:23

explanation algorithms where you you will explain individual predictions So if you remember a classification problem

3:07:31

we we talked about the decision boundary that this could be very very very

3:07:38

complex In this case is highly complex It does uh it you could fit maybe

3:07:46

several polomials here um high orderer and and everything to

3:07:54

classify between crosses and and red crosses and blue dots blue circles

3:08:01

Uh so it's very difficult to explain right it's very difficult to explain what the model is doing If you take a

3:08:08

look at this decision boundary right if it was uh it was uh a a more simple

3:08:15

decision boundary uh it would be easy to explain

3:08:21

Right now the the global explanations there are algorithms that try to make

3:08:27

sense of this but there are local algorithms that look at one prediction

3:08:33

So for example this person here or this data here is uh is classified as sick

3:08:40

For example this red cross and it just looks at that vicinity

3:08:46

of so if you think about xaxis being um

3:08:54

uh being age and yaxis being uh if you

3:08:59

smoke or not Uh you can think about that

3:09:05

uh vicinity here and you can see that um in that that vicinity any twix little

3:09:13

twix in age and and uh and uh I gave a

3:09:18

discrete example Let me change the example um blood pressure for example in

3:09:24

the y- axis any twick uh

3:09:29

can be explained as a little line right so if if you're in a if you zoom enough

3:09:35

you can approximate this uh locally this decision boundary by a linear uh kind of

3:09:43

boundary and then you can explain this with linear regression explain how much

3:09:48

of the age and how much of the blood pressure um um

3:09:54

kinds of influence the decision locally Okay And you have um

3:10:02

models that export uh data points export visual dashboards export feature

3:10:08

importance uh or even export surrogate models The surrogate models are this

3:10:13

local models that you can use locally For here you will have a model of this

3:10:18

line to use locally So I will just go really really really uh easy on two sha in the shallow

3:10:26

explanation of two different explainable AI tools Lime and shap So lime is again

3:10:32

is is exactly what I explained So let's say that we predicted that this guy will

3:10:38

have a stroke But you can see that it's very complex right so let's say that

3:10:43

here is age and here is hyp hypertension Th this model is very complex It's not

3:10:49

linear but in this vicinity here we can

3:10:55

see that if this is age and here is hypertension You can see

3:11:02

that having like the the age was much more important to classify

3:11:10

this guy as as um as having a stroke Right uh you can see that you can range

3:11:18

a lot in hypertension in both classes and you you're not changing classes but

3:11:25

but but going on the horizontal axis at some point you will get separated So you

3:11:32

can think about the first this is a uh this is a response from SHAP for example

3:11:38

when you run it in Python um or other other programming languages

3:11:43

it's saying that age uh uh has an importance

3:11:50

to uh to stroke So left is stroke zero here is the middle left stroke right is

3:11:57

no stroke So age has an importance uh a great importance that it's higher

3:12:03

than the hypertension uh to stroke right and the and the

3:12:09

others here for example the um um I don't know if the gender is female it it

3:12:16

it it is it's more important it's more towards um

3:12:23

no stroke but it the importance is very very very subtle so these are

3:12:28

explanations and you can we you can uh use this uh algorithms to monitor how

3:12:34

your LLM or LLM or whatever uh uh model you're using And this is model agnostic

3:12:42

Lime stands for linear interpretable uh

3:12:49

surrogate model um explanability right so um

3:12:57

you can you can use this to kind of try to understand the features uh a little

3:13:03

bit better But this is local right so I'm I'm telling that in this in this

3:13:09

kind of bulk and set of features this is what kinds this is what's making

3:13:17

a difference for this person Uh so that's lime

3:13:25

So shap stands for shapley values and it's based on game theory So I'm not

3:13:30

going to go into details of how it works but it it borrows from uh uh modeling of

3:13:38

um economics gain theory where for example

3:13:44

if you have earnings uh um if you have a team of people and you got to they win a

3:13:52

they win a prize uh how how can you distribute this this

3:13:58

prize fairly amongst the participants So uh how can you eval evaluate the

3:14:05

participation uh of each person right uh h how how

3:14:11

each of these persons contribute contribute and it's not a simple or

3:14:16

trivial task um for example let's say that I have Joanna

3:14:22

Louisa and David in the same group but Joanna and David they work together um

3:14:30

And so although Joanna participated a lot uh if there is no David there would

3:14:38

there will be no Jo Joanna so you wait less it's uh her participation and so on

3:14:44

So this is what chap values or this uh this algorithm is doing with a model

3:14:50

It's also model agnostic and um in the end of the day it will

3:14:55

expose you the contribution of the of the feature uh and it's also locally analyzing this

3:15:04

locally for a particular prediction So let's say that here you have a prediction of of um

3:15:13

risk of of having I don't know um

3:15:18

a depression and uh you can see that this this feature here contributes to a positive

3:15:25

value of the result for uh way more than this other red ones and the blue ones

3:15:34

contributes uh in the opposite direction So you can interpret this graphics

3:15:39

locally but uh shap values have an important characteristics that the first

3:15:46

uh lime uh models uh lime explainable models that you can aggregate

3:15:54

this several local predictions uh to kind of come to global

3:16:00

explanations So the aggregations have some mathematical rules but I'm I'm

3:16:06

leaving uh kind of this very shallow introduction so that you can um if you

3:16:11

have interest you can go and study a little bit more of this explainable AI tool Uh so I want to talk right now

3:16:19

following our um

3:16:26

it's this last the last topic of synatic data and differential privacy Okay So

3:16:32

this is also trending and it is very um

3:16:42

important and and um machine learning and generative AI has all to do with

3:16:47

this Okay All right So let's start um

3:16:54

uh talking about synthetic data

3:17:01

Um so what is synthetic data synthetic data is uh data that is artificially

3:17:08

generated from real data using algorithms and modules that try to

3:17:14

replicate the statistical characteristics of the real data

3:17:20

And the reason we use synthetic data is that data

3:17:26

uh sometimes is complex to come by right in several domains uh or uh is sensitive

3:17:33

and and carry confidential information like um medical records financial

3:17:40

records Um and uh so what we we we can say that the

3:17:48

synthetic data is actually um revolutionizing the simulations because

3:17:53

it's enabling the creation of this uh difficult data sets by having accurate

3:18:01

approximations of the real data So that would allow us to explore

3:18:07

complex scenarios make better uh informed decision um for for several

3:18:14

applications uh uh and uh also to simulations So for

3:18:20

example let's say that you have uh you have to make uh uh control systems and

3:18:27

um uh solutions and machine learning uh

3:18:32

u models statistical models whatever you're doing with data uh for autonomous

3:18:37

cars So let's say that you you want to make this the autopilot

3:18:44

uh um you want to train this autopilot to make the best decisions but you

3:18:50

cannot uh collect real world data in some situations You cannot crash the car and

3:18:57

um simulate all this uh very strange simulations right so uh strange um

3:19:04

situations So simulation is uh is also something that is

3:19:10

benefiting from uh artificial data because we can generate this data right

3:19:16

so uh you can even use generative AI to

3:19:21

generate this kinds of data where in the real world this world is so very

3:19:27

difficult to come by Okay Um

3:19:32

so we can also say that you know like synthetic data enhances predictive analytics because we know that the

3:19:39

models are better trained in plentiful volumes of data and so we can do data

3:19:44

augmentation We can generate um you know more data

3:19:50

uh and to to train models

3:19:55

and uh we can by augmenting this data we will train this models in in this data

3:20:03

um come come uh uh come by uh um with

3:20:09

solutions and then uh the ultimate goal would be to transfer algorithms and

3:20:16

models back to real world data right

3:20:21

um there are also data that is rare For example rare diseases Um it's it's you

3:20:28

know you don't have you have a limited uh amount of data

3:20:33

So we can use uh uh synthetic data to supplement this kinds of data sets Okay

3:20:42

the the other the other way that we know that synthetic data enhances um all this

3:20:49

prediction models and anal uh in general predictive analytics tasks is by

3:20:56

reducing bias and fairness issues So um it might be that you have data set

3:21:04

that's representing diverse populations and scenarios but you you know you have

3:21:10

um you have uh you know populations that are minority and you don't have uh

3:21:18

enough data so you have an imbalanced data This bias can easily uh be

3:21:24

reflected in your model So by generated generating synthetic data of the uh in

3:21:31

this in um um minority groups and trying to overcome the imbalance of this data

3:21:39

is one of the other advantages of synthetic data

3:21:45

Um so another another

3:21:51

important aspect of uh of of uh of

3:21:56

synthetic data So we talked about simulations data augmentation and and kind of removing bias but also

3:22:03

protecting data privacy right uh so

3:22:11

the um um data privacy right uh uh so

3:22:18

for data privacy you can use synthetic data uh to generate realistic data sets

3:22:25

without exposing sensitive information So you are generating data that again

3:22:33

maintains the statistical properties but it it it um it it uh it doesn't

3:22:42

expose sensitive information because this is just a synthetic data

3:22:49

Okay Um so the the ways there are um uh

3:22:56

we we we see that there are several algorithms that generates uh synthetic

3:23:02

data and and different ways of doing this uh with the advent of generative AI we

3:23:09

we see more and more algorithms that are you know based on generative generative

3:23:14

AI um and and uh they are state-of-the-art and replicating statistical characteristics of this real

3:23:21

data one example is what we call a generative adversarial networks or GANs

3:23:28

or variational autoenccoders right so uh both of them I'm not going to go

3:23:36

into details GANs or adversarial networks or generative uh networks They

3:23:42

are usually uh there are two networks usually working together One generates

3:23:48

and the other one uh classifies in in order to test the how how good the

3:23:57

generator was So they they do adversarial tasks and by getting both

3:24:02

the best that you can you have a very um a very good model for generating uh data

3:24:12

Okay So uh obviously that you have tradeoffs to you should uh and you know

3:24:18

like to select this algorith algorithms you should understand what are you doing

3:24:24

what each algorithm does um and you have also you know um um other methods like

3:24:32

that uses uh statistical methods such as gshian mixture models So if you're

3:24:39

interested in that you can um uh look up some great resources for that

3:24:47

Okay So um

3:24:53

talk now a little bit more about differential privacy What's differential privacy

3:24:58

uh uh it is a mathematical framework to ensure the privacy of

3:25:06

individuals So you use mathematics to release data

3:25:14

and you you with with using this mathematics you can guarantee

3:25:21

a level of privacy of the individuals and also like this level of privacy

3:25:28

quantifies it's a measurement is a measure of of of of um of quantifying

3:25:36

the risk associated ated with the data set um of leaking information

3:25:44

So the way that it works is that you add calibrated noise to the data set So you

3:25:52

can use it um and there are several models that can do

3:25:58

that right um and so you noise this you add this noise

3:26:06

in a way that it preserves the overall statistical properties the best that you can but protects

3:26:14

individual privacy So uh you have the benefits of um the benefit of kind of

3:26:21

having the real world data underlying information but providing some stronger

3:26:27

privacy guarantees right and um now there are challenges and

3:26:33

considerations like imple implementing differential privacy requires careful

3:26:39

consideration of the tradeoff between the privacy and the utility So the the level of noise you add to the data must

3:26:46

be balanced to maintain the statistical properties of data but still guaranteeing strong

3:26:54

uh guarantees Right so when uh we usually say that a

3:27:01

algorithm or a solution or um

3:27:06

a model is epsilon differential privacy differentially private It means that

3:27:13

this epsilon is a number that specify an an upper bound on privacy loss So the

3:27:21

risk of leaking information So

English (auto-generated)