**TRANSFORMER SENTIMENT**

2.coding (5 min)

-samen gpu tonen

-samen torch cuda tonen

-samen transformers installeren

-samen pipeline importeren

Uitleggen

3 transformer model (15 min)

classifier = pipeline("sentiment-analysis", device=0)

TD leg uit

-vocab

-layers – head

-size or dimension

Check again

hidden size or hidden dimension

**"dim": 768**

DistilBERT is a smaller version of the BERT (Bidirectional Encoder Representations from Transformers) model. The number **768 corresponds to the dimensionality of the hidden layers in DistilBERT**. In other words, when DistilBERT processes input data, it transforms the information into **vectors with 768** values in each vector. These vectors capture the semantic meaning of the input text and can be used for various natural language processing tasks like text classification, named entity recognition, and more.

(hidden\_dim": 3072)

The hidden dimension or hidden size in transformer models refers to the dimensionality of the hidden layers in the model. In many transformer-based models, including BERT and DistilBERT, the hidden size is a hyperparameter that determines the number of values in the vectors representing the hidden states at each layer.

For example, in DistilBERT, the hidden size is often 768. This means that the hidden layers in the model produce vectors with 768 values at each position in the input sequence. The hidden size is a critical parameter as it influences the capacity and expressiveness of the model, allowing it to capture and represent complex patterns in the input data. Different transformer models may have different hidden sizes based on their architecture and intended use cases.

That's correct! In DistilBERT, the hidden layer has a dimensionality of 3072. This means that the intermediate representations of the input data are transformed into vectors with 3072 values in the hidden layers of the model. The high dimensionality allows DistilBERT to capture complex patterns and relationships in the input data, enabling it to learn rich representations for a variety of natural language understanding tasks.

hidden\_dim": 3072

**vocab\_size": 30522**

The "vocab\_size" parameter in the context of a transformer model like DistilBERT refers to the size of the vocabulary that the model is trained on. In the case of DistilBERT, the "vocab\_size" is set to 30522, which means that the model has been trained on a vocabulary of 30,522 unique tokens or words. This vocabulary size determines the number of distinct tokens that the model can recognize and generate embeddings for during its training and inference processes.

("max\_position\_embeddings": 512,)

The "max\_position\_embeddings" parameter in the context of DistilBERT refers to the maximum sequence length that the model can handle. In the case of DistilBERT, this parameter is set to 512. It means that the model can process input sequences with a maximum length of 512 tokens.

This parameter is important because transformers, including DistilBERT, process input data in fixed-size chunks, and sequences longer than the specified maximum would need to be truncated or split to fit into the model's architecture. The "max\_position\_embeddings" value essentially defines the maximum number of positions or tokens that the model can attend to in a single pass.

TD leg uit

Heads

Layers

In a transformer model, such as BERT or DistilBERT, the terms "heads" and "layers" refer to important components of the model architecture:

1. **Heads:**
   * In the context of attention mechanisms, a "head" typically refers to an independently operating attention mechanism within the model.
   * Each attention head attends to different parts of the input sequence, capturing different aspects of the relationships between words or tokens.
   * The output of multiple attention heads is often concatenated or linearly combined to form the final output for a given layer.
2. **Layers:**
   * The transformer model consists of multiple layers. Each layer contains two main subcomponents: the self-attention mechanism and the feedforward neural network.
   * The self-attention mechanism allows the model to weigh the importance of different words in the input sequence with respect to each other.
   * The feedforward neural network processes the outputs of the self-attention mechanism, introducing non-linearities and capturing complex patterns.
   * The entire model processes the input data through multiple layers, with each layer refining the representation of the input sequence.

In summary, heads operate within individual layers, and multiple layers make up the entire transformer model. Each head contributes to capturing different aspects of the relationships in the input data, and the combination of multiple layers and heads enables the model to learn hierarchical and nuanced representations of the input.

Model Name: distilbert-base-uncased

Model Type: DistilBertModel

Model Configuration:

{

"\_name\_or\_path": "distilbert-base-uncased",

"activation": "gelu",

"architectures": [

"DistilBertForMaskedLM"

],

"attention\_dropout": 0.1,

**"dim": 768**,

"dropout": 0.1,

"hidden\_dim": 3072,

"initializer\_range": 0.02,

"max\_position\_embeddings": 512,

"model\_type": "distilbert",

**"n\_heads": 12,**

**"n\_layers": 6,**

"pad\_token\_id": 0,

"qa\_dropout": 0.1,

"seq\_classif\_dropout": 0.2,

"sinusoidal\_pos\_embds": false,

"tie\_weights\_": true,

"transformers\_version": "4.34.1",

**"vocab\_size": 30522**

}

Studenten genereren zelf de vergelijking v 5 modellen

Vergelijk BERT vs Roberta modellen

Vergelijk nederlands BERTje met Robbert

BERT – Distilbert

BERT meer layers en heads,

Roberta – Robbert – BERTje

Robbert vs BERTje

Robbert : vocab groter, nr of layers

TD: Desondanks wat zie je in resultaten,

BERTje kleiner maar betere performance

3.1 Classify examples (5 min)

Generate examples good – bad and in gradations

What do u see?

## 4.Data

STEP 1 (10 – 15 min)

Studenten autonoom

Analysis of data

write your code in code fields below for:

-general info  
-check the first 5 instances  
-size of data set  
-count sentiment label values  
-check instance 20  
-check only the text of instance 20  
-check only the label of instance 20  
-make histogram for the sentiment labels

STEP 2 alle stappen door teacher uitgelegd (15 min) / of zij gaan er door

Leg reden uit v selectie alleen 500 : maakt deel uit v bredere les gecombineerd met fine tuning

6.1 NIET

TRANSFORMER ZERO SHOT alles zelf

Voeg toe voor test voorbeeld

**Sommige neutrale sentimenten zullen werken**

**NU IS BEDOELING ALLE sentiment classes te gebruiken**

**De 500 pos 500 neg 500 neutr**

**2.1 data preparation**

**Vrij gelijk aan sentiment**

**2.2 predict iets moeilijker**

**Cosine similarity**

**Verschil transformer en sentence transformer**



**You**

what is the difference between a transformer or a sentence transformer

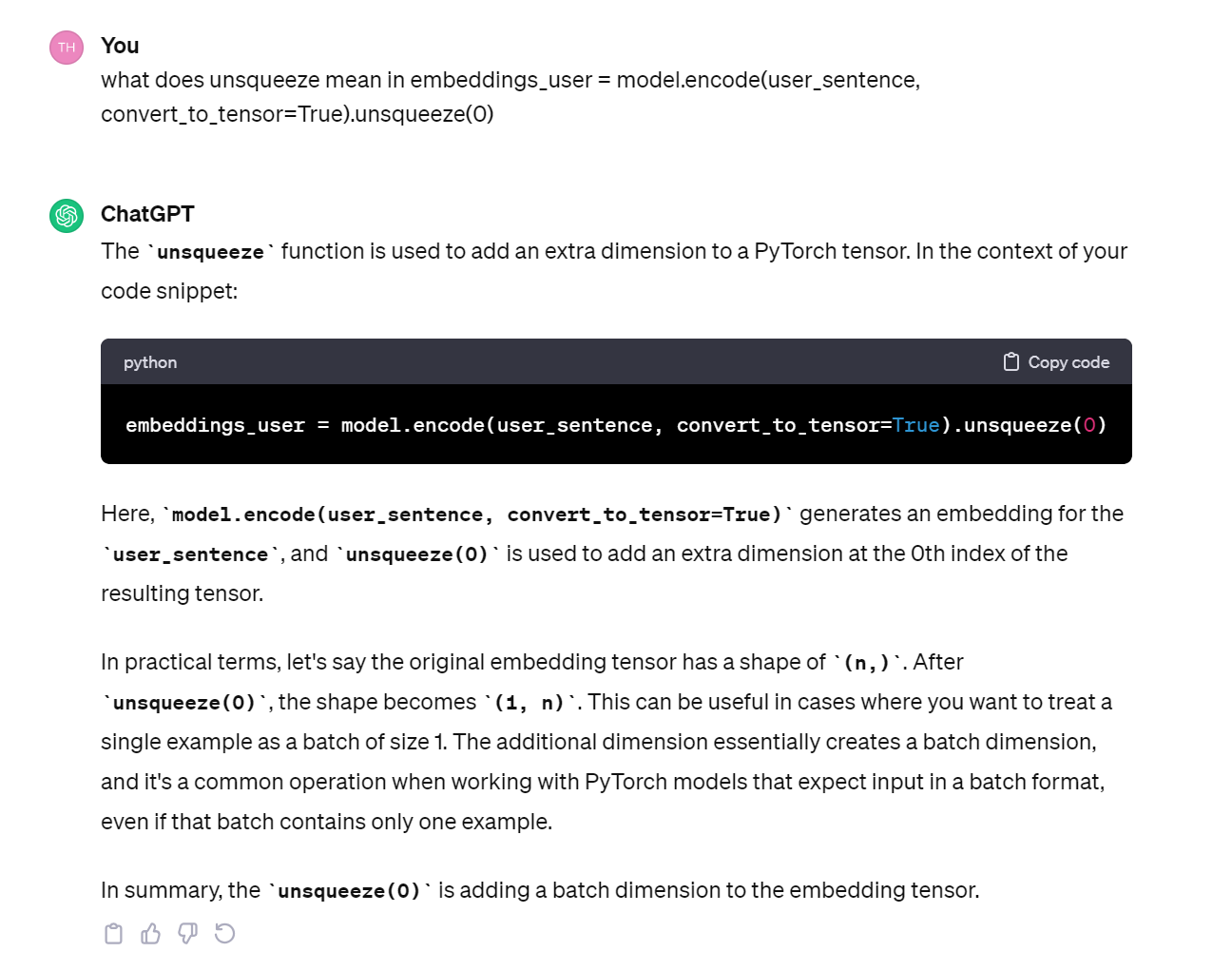
ChatGPT

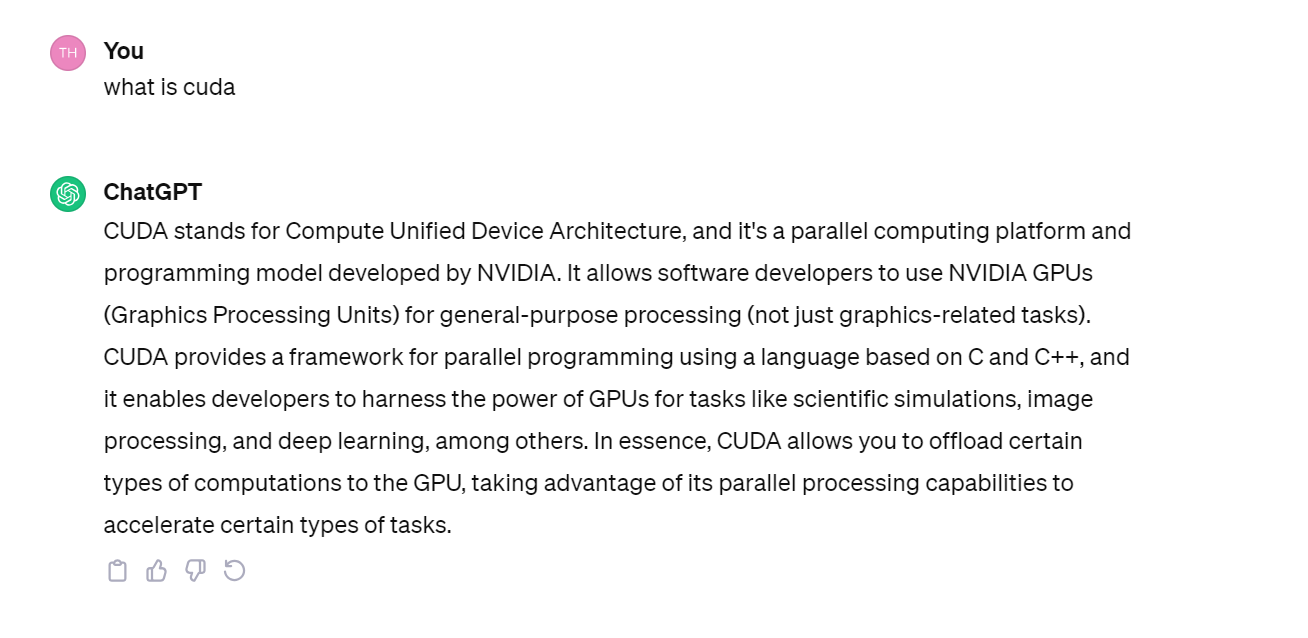
**ChatGPT**

Great question! A transformer is a type of neural network architecture that's been widely used in natural language processing tasks. It's known for its ability to handle sequential data, like sentences, in parallel, making it efficient for tasks like language translation and text generation.

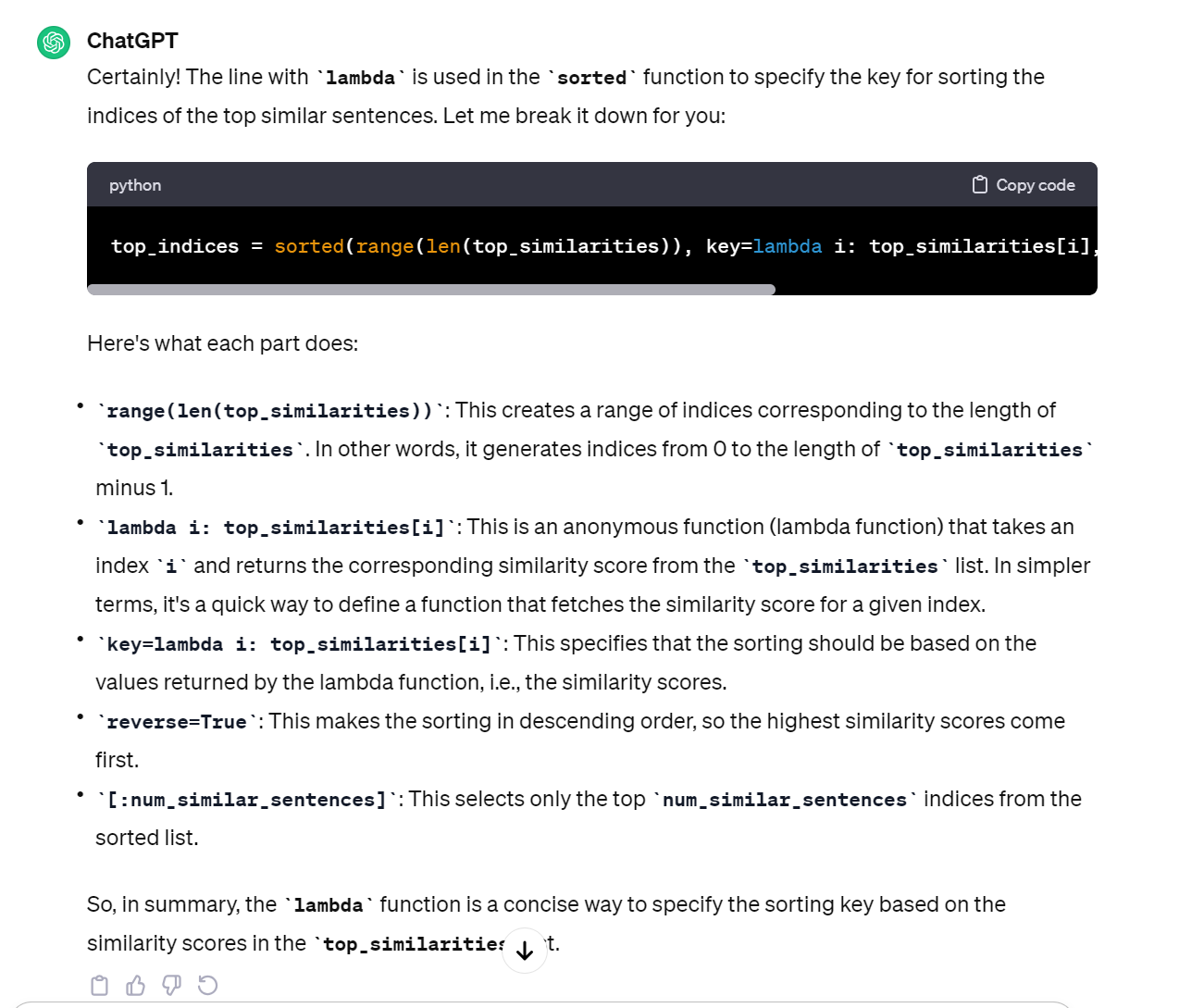
On the other hand, a sentence transformer specifically refers to a model that's designed to encode and understand entire sentences. While transformers can be used for various tasks, including sentence-level ones, a sentence transformer is usually fine-tuned or pretrained specifically for tasks like sentence similarity, semantic textual similarity, or other tasks that involve understanding and representing sentences as vectors in a meaningful way.

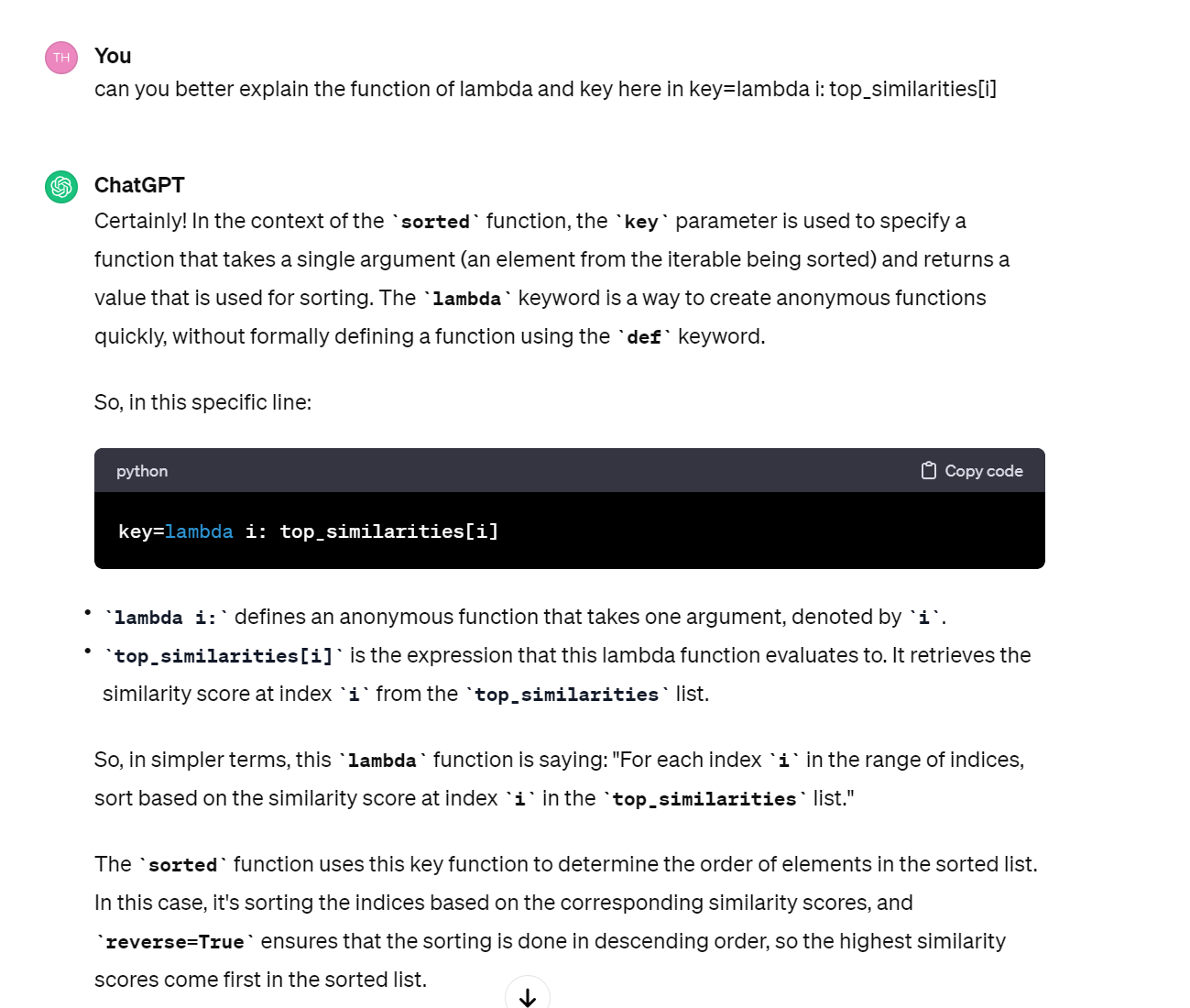
**So, in essence, all sentence transformers are transformers, but not all transformers are necessarily fine-tuned or designed specifically for sentence-level tasks.**

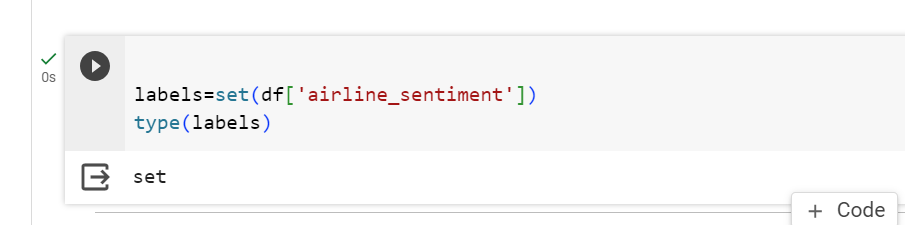


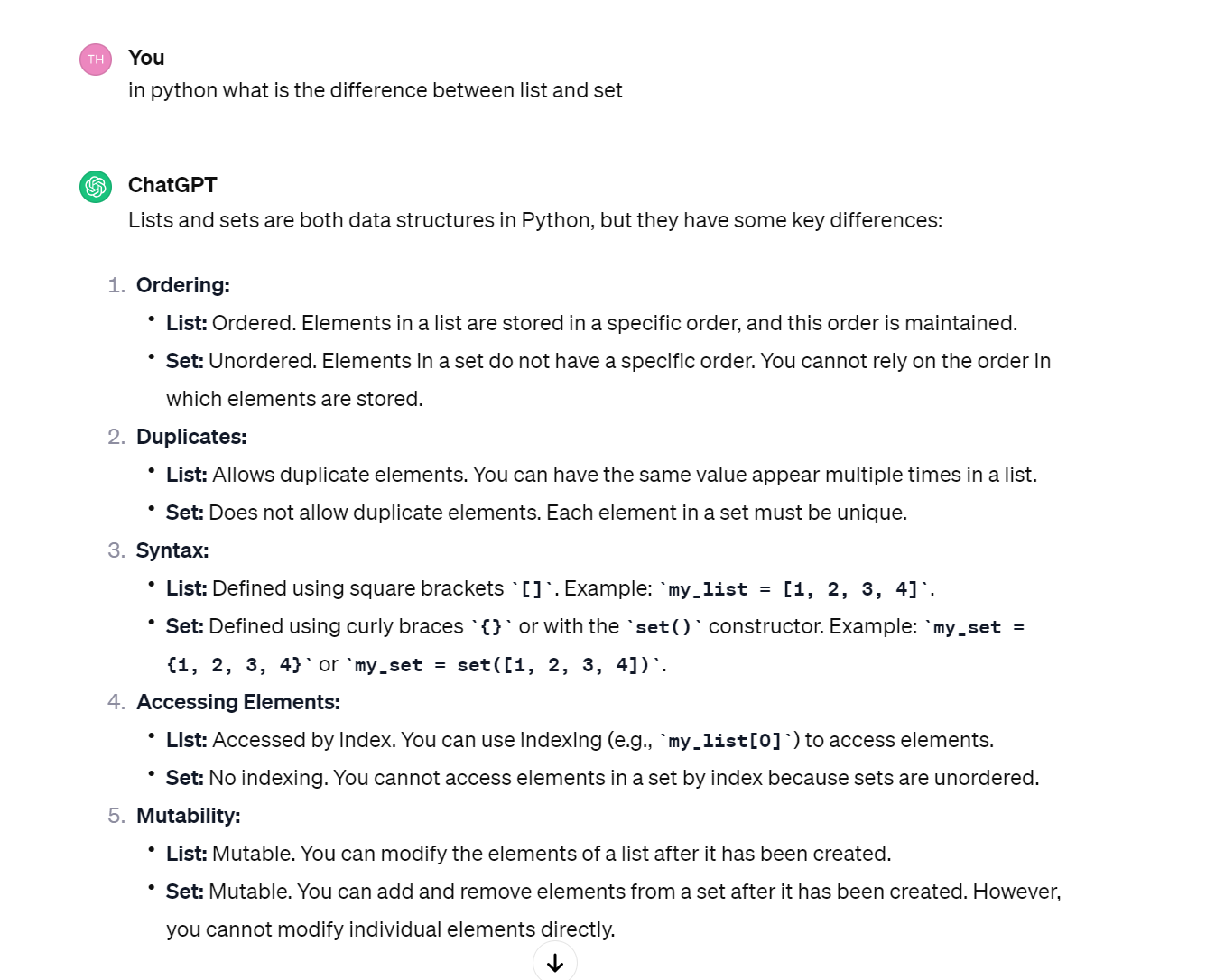


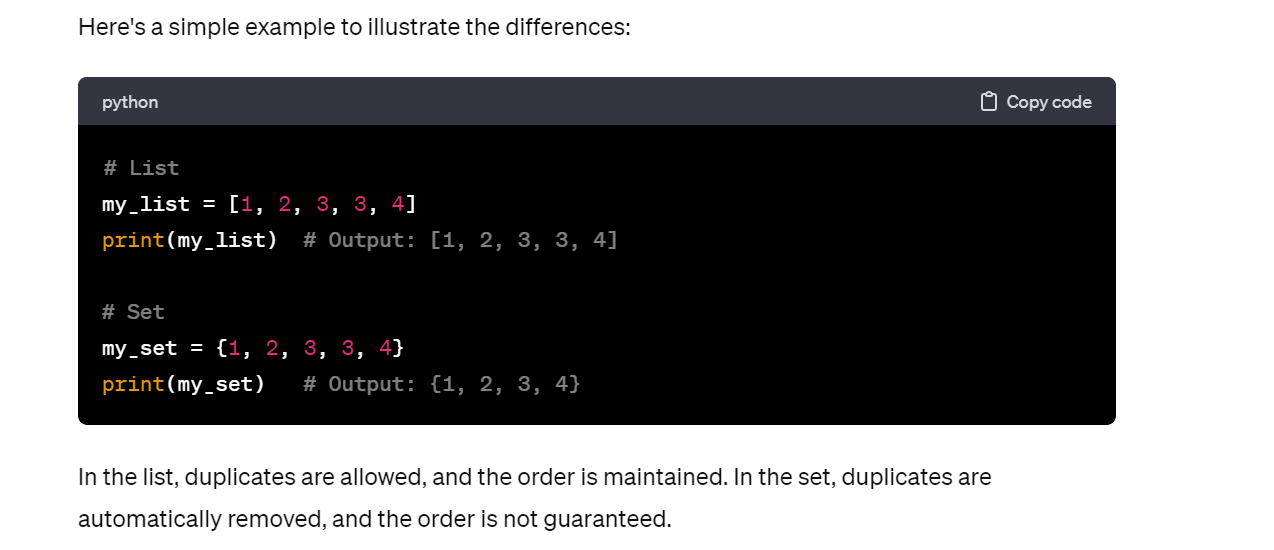
top\_indices = sorted(range(len(top\_similarities)), key=lambda i: top\_similarities[i], reverse=True)[:num\_similar\_sentences]











List ook gebruiken

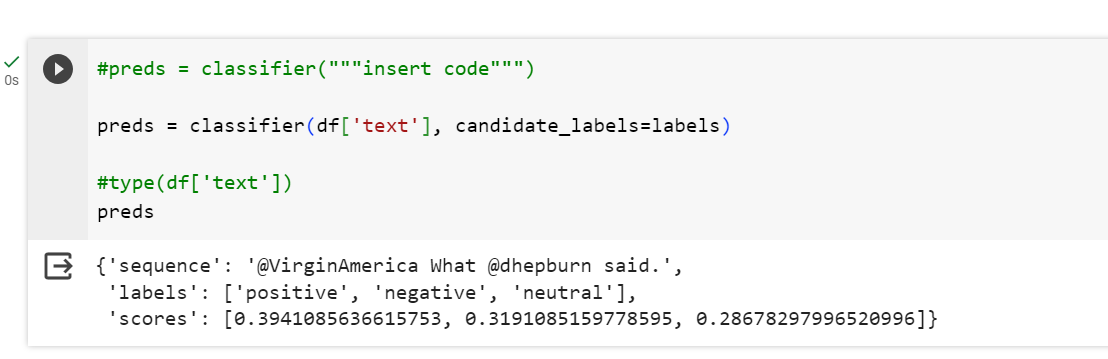
Labels = list(set(df[‘airline\_sentiment’]))

TD opmerkingen en vragen

What could be improved here for cos sim scripts:

Sentences split on punctuation

Hoewel, gelijkenis zal berekend w op basis van1ste 512 tokens

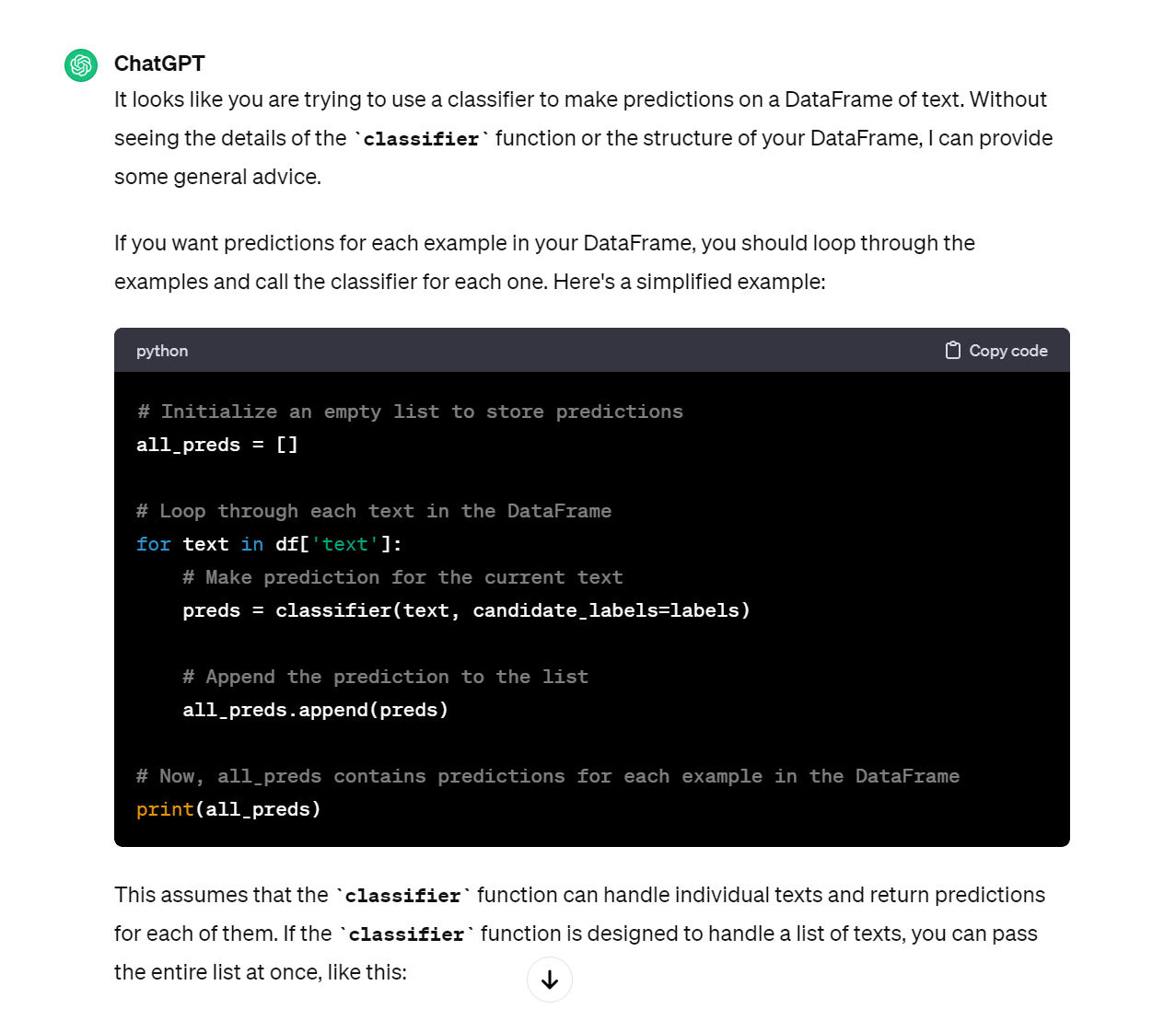


Only outputs 1 case not the complete rows, input should be list

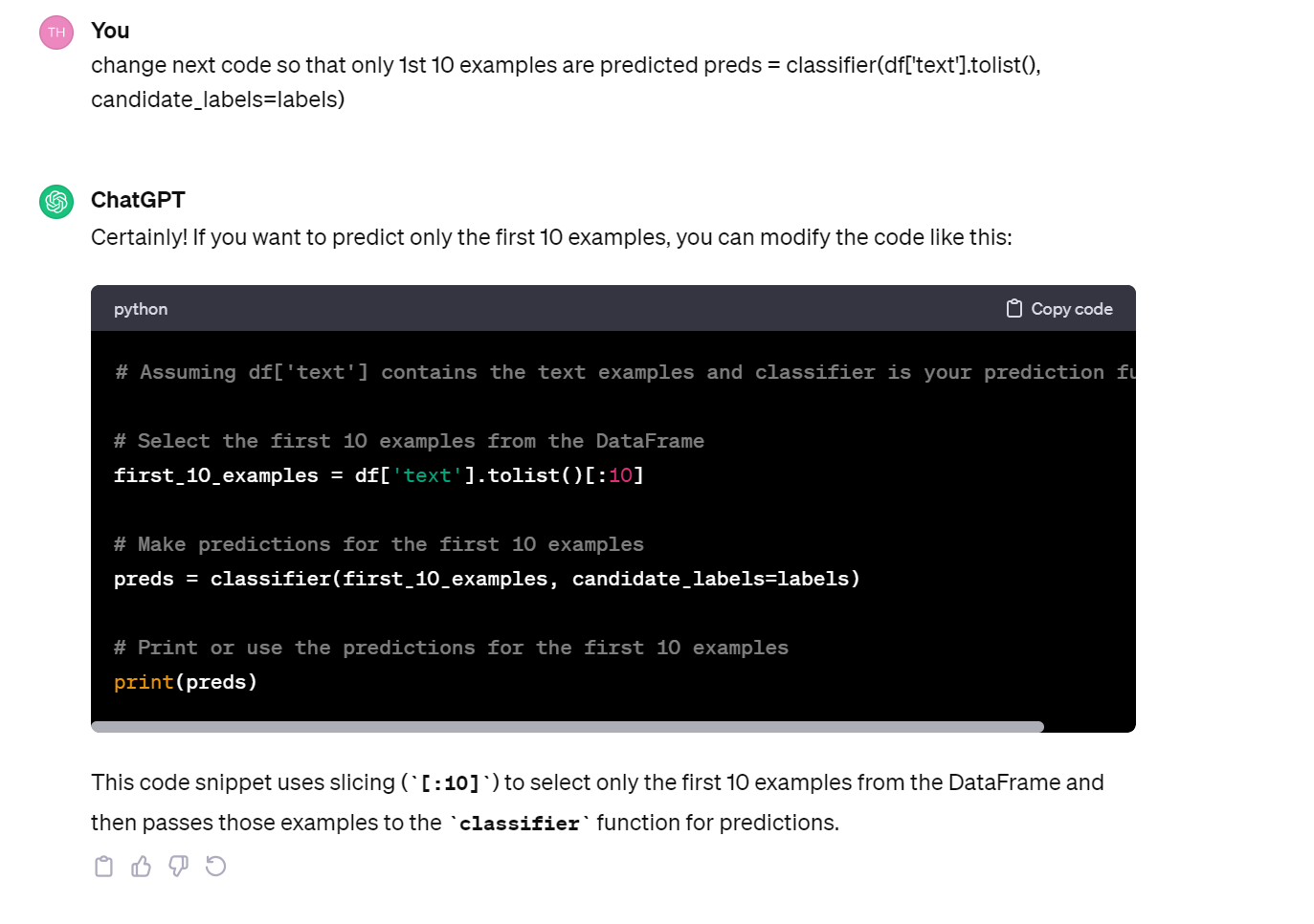
Df[‘text’] is geen lijst



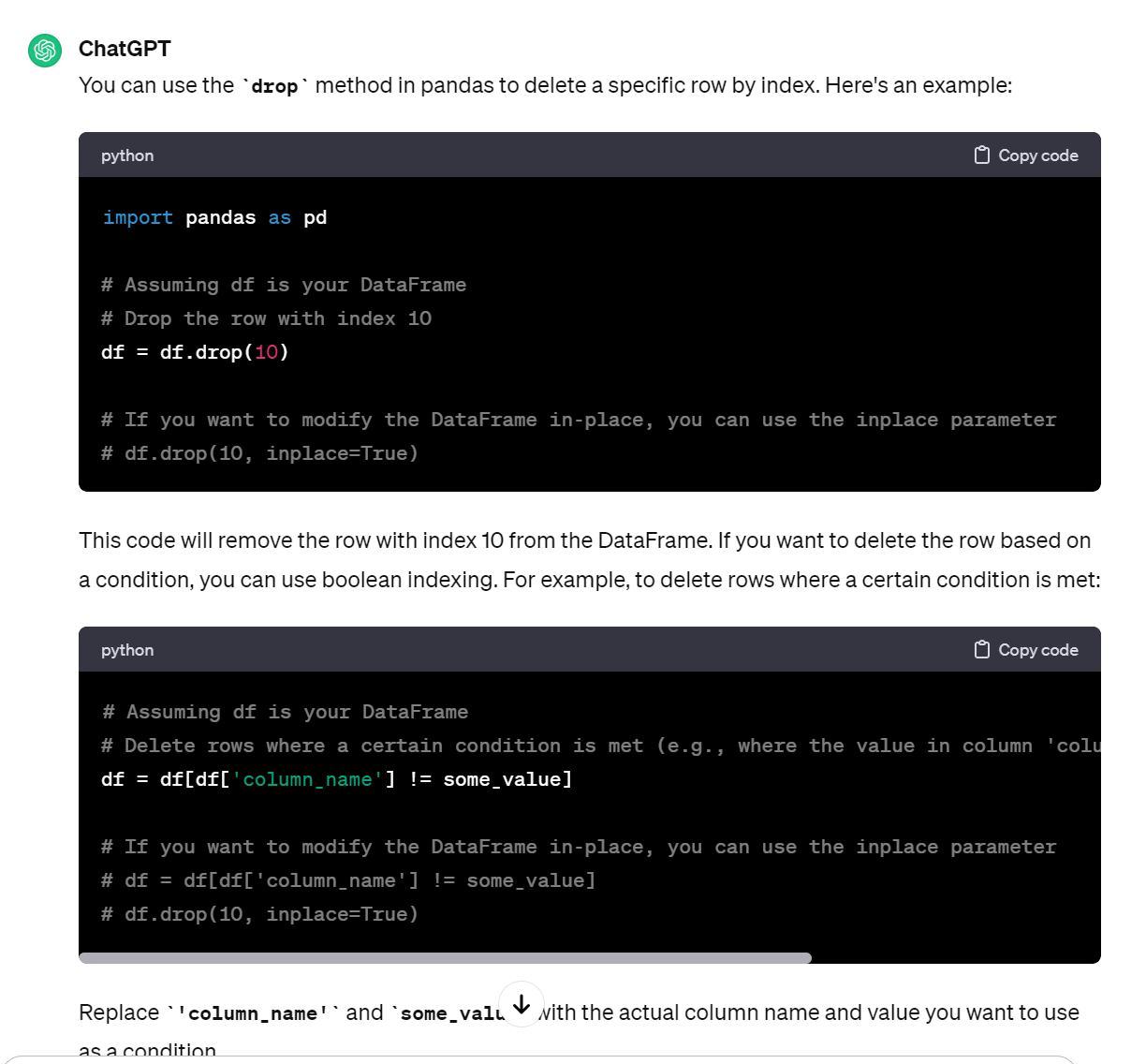
Input should be list

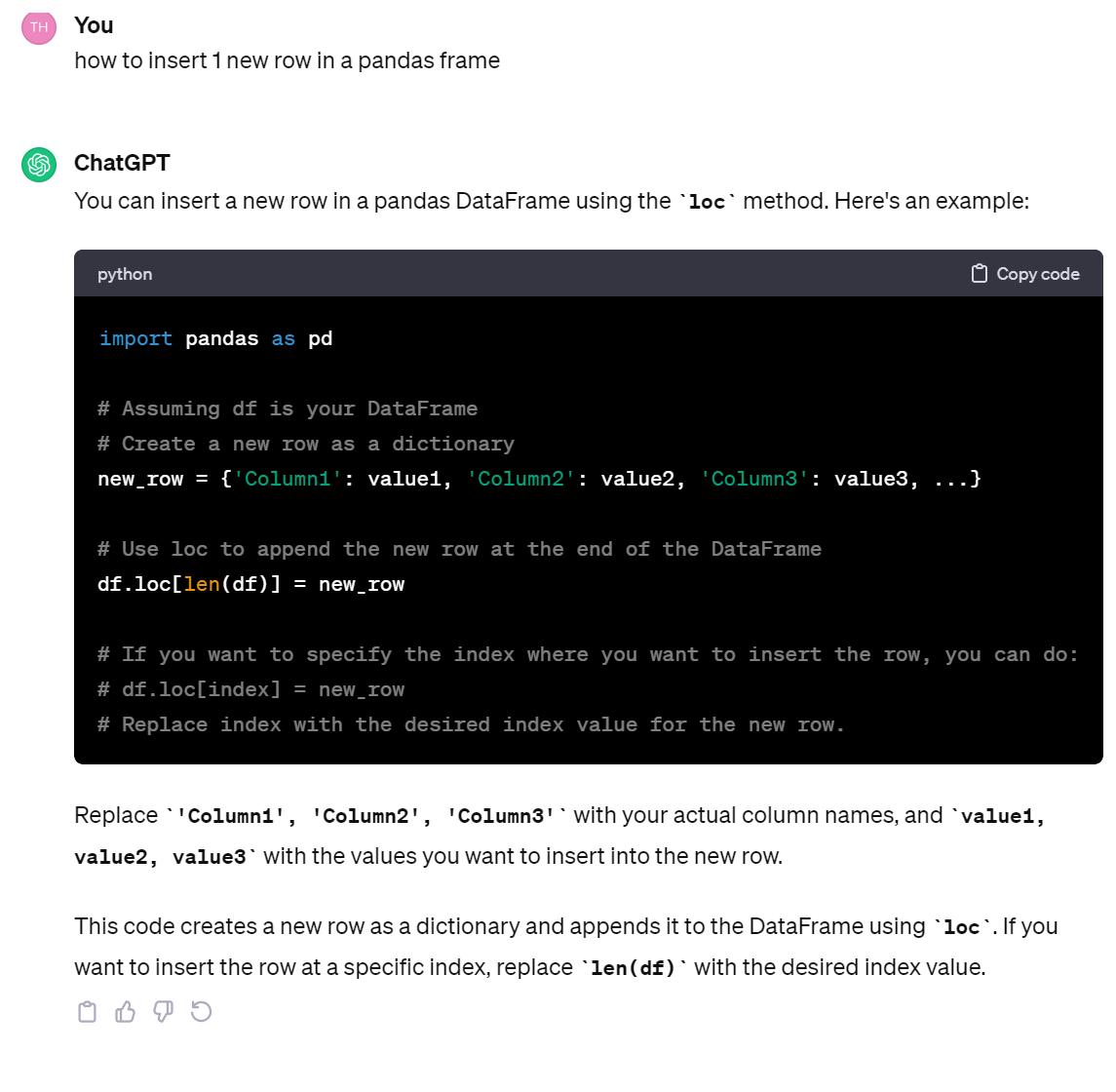






Prediction voor zero shot class duurt 3-4 min





PPT les

Wat weet je technisch gezien over chatgpt/

Wat vind je er VOORAL ongelooflijk aan toen je er 1ste keer mee werkte.

TD taalvaardigheid model?

Slide 2

Kaderen context ML

Specifieke toepassingen

Slide 3

TD : teveel info,

Studenten laten uitleggen ML,

Wat is jullie ervaring

Platforms

Vooraf a slide 4

Hoe klasseer je vormen van ML

Slide 4

Slide 5

Transformer semi-supervised

Vanuit jullie ervaring v machine learning, welke vormen

Feature based

Slide 6

Feature

Slide 7

Adj features v sentiment classification

Slide 9

Ken je een vb van reinforcement learning. Chatgpt

We hebben data nodig.

Hoe ga je bij je data te werk, hoe split je

Slide 10

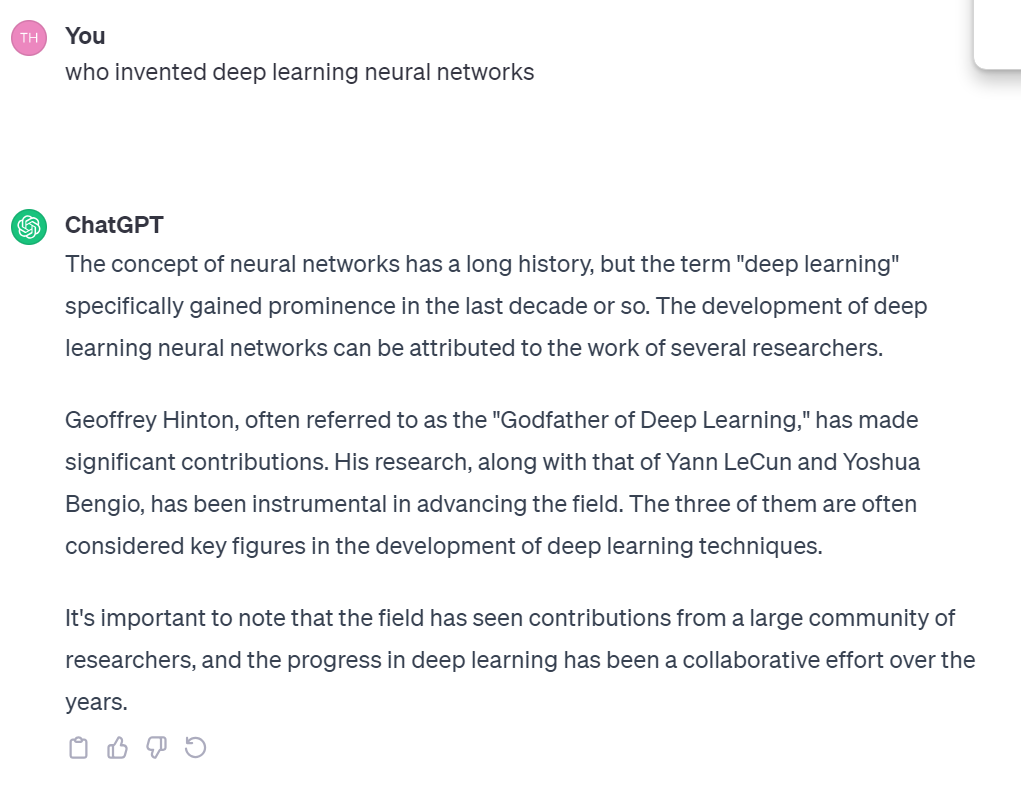
Data set split

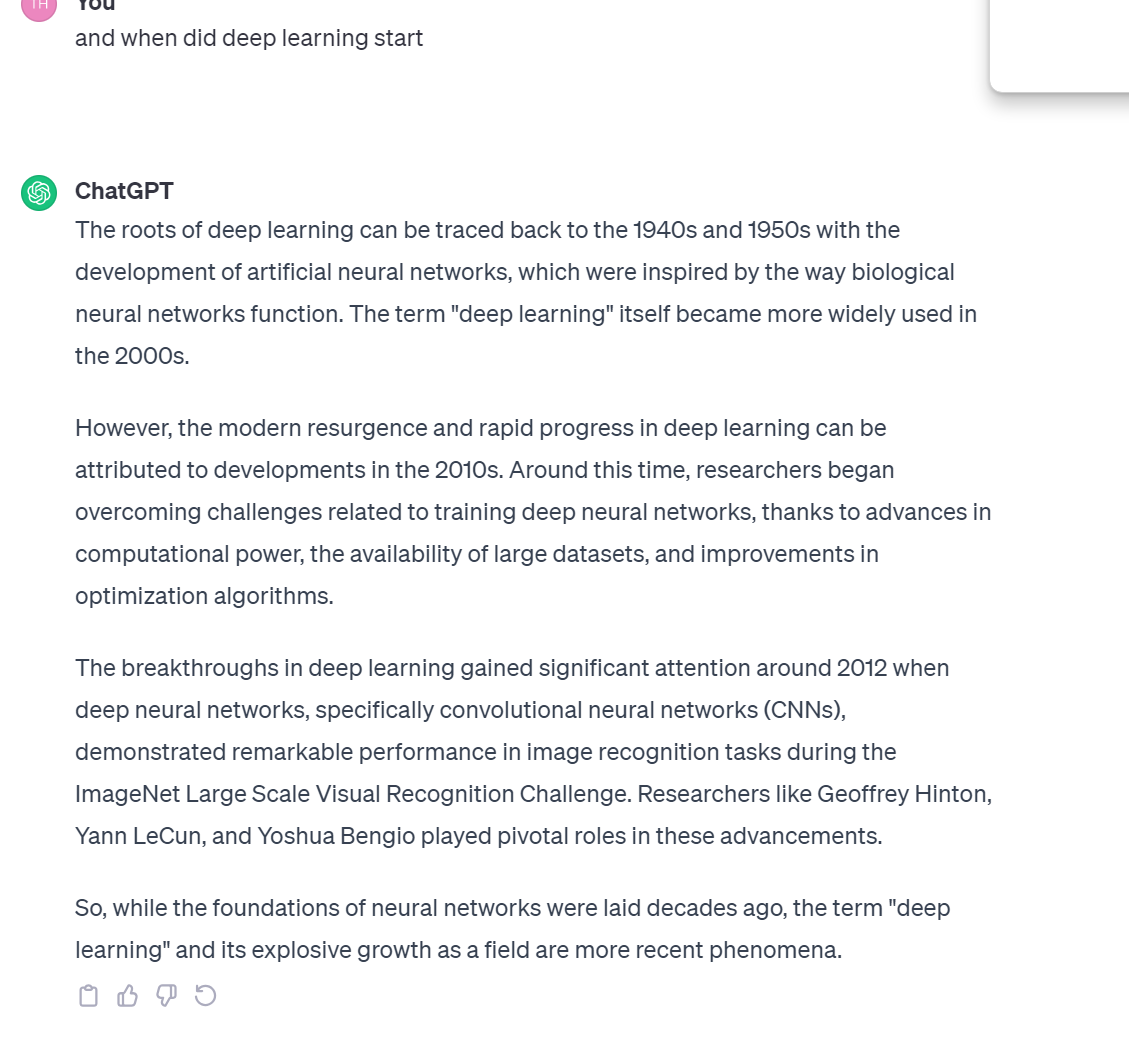
Wat is nog belangrijk bij processen data set

Slide 11 + slide 12

GPU : zijn er hier gamers, gaming computer GPU

Onderscheid machine learning deep learning

v



Slide 13

DL neurons

Filmpje Hinton ivm model hersenen?

Op slide 13 verander NEURON naar **NODE**

**VERMELD ACTIVATIE FUNCTIE**

Slide 14

Link naar sentiment

Slide 15 + 16 + 17 + 18 + 19

Welke info in de zin is emotie geladen?

Waar meest gewicht, welke lijnen meest vetgedrukt?

Sentiment uitleggen met DL

Er kunnen meerdere hidden layers zijn, theoretisch, hoe meer data hoe meer hidden layers je kunt selecteren

Softmax

logits

Slide 20 + 21 + 22 + 23

Encoder decoder : woord nr nummer, nummer nr woord

Vermeld LSTM GRU

Slide 24 + 25 transformer

ATTENTION IS ALL U NEED

Vaswani

Devlin

Notie v taalmodel

Hidden features word embeddings

Wat zijn word embeddings

Slide 26 + 27

Cosine similarity

Slide 28 + 29 + 30

Self attention

Email voorbeeld @

The cat on the mat sleeps

Self attention alle woorden gerelateerd, FUNDAMENTEEL verschillend v modellen waar alleen x vorige woorden of x volgende woorden worden in rekening gebracht

Slide 31

Example self attention gevisualiseerd

Slide 32

Leg matrix uit : LINK met self attention leggen

Slide 33

Types modellen

Slide 34 + 35 + 36

Pre training en fine tuning

Slide 37

Chatgpt

Waarom andere modellen dan chatgpt

Slide 38

Graad openheid modellen

Slide 39

Huggingface

Slide 40

Pre train en fine tuning sentiment via transfer learning

Slide 41 + 42

Mask en nsp

15pct random selection v mask

Slide 43 + 44

Transfer learning en fine tuning

Slide 45 tokenization **weglaten**

**Slide 46**

**3 usage levels**

**Pretraining : verwijs nr distilbert**

**Slide 47 + 48 + 49 + 50 + 51**

**Simple emoties makkelijk**

**Meer fine grained moeilijk**

**V slide 50, vraag studenten wat nut sentiment analys is**

Vraag studenten wat voordeel transformer kan zijn tov bow en tfidf

Slide 52

Bow tfidf bert

Slide 53 : uitleg assignment sentiment –

Leg uit dat neutral sentiment vaak moeilijk is

Slide 54 + 54

zero shot learning

leonardo da vinci

wie was Leonardo da vinci

slide 57 assignment

27112023

-check nog eens of zonder gpu ook werkt

-#if no arguments for model, it takes distilbert by default

Toon voorbeeld met ander model

classifier = pipeline("sentiment-analysis", device=0)

from transformers import pipeline, BertTokenizer, BertForSequenceClassification

# Load BERT tokenizer and model

tokenizer = BertTokenizer.from\_pretrained("bert-base-uncased")

model = BertForSequenceClassification.from\_pretrained("bert-base-uncased")

# Create a sentiment analysis pipeline with BERT

classifier = pipeline("sentiment-analysis", model=model, tokenizer=tokenizer, device=0)

# Example usage

result = classifier("I love using transformers for natural language processing!")

print(result)

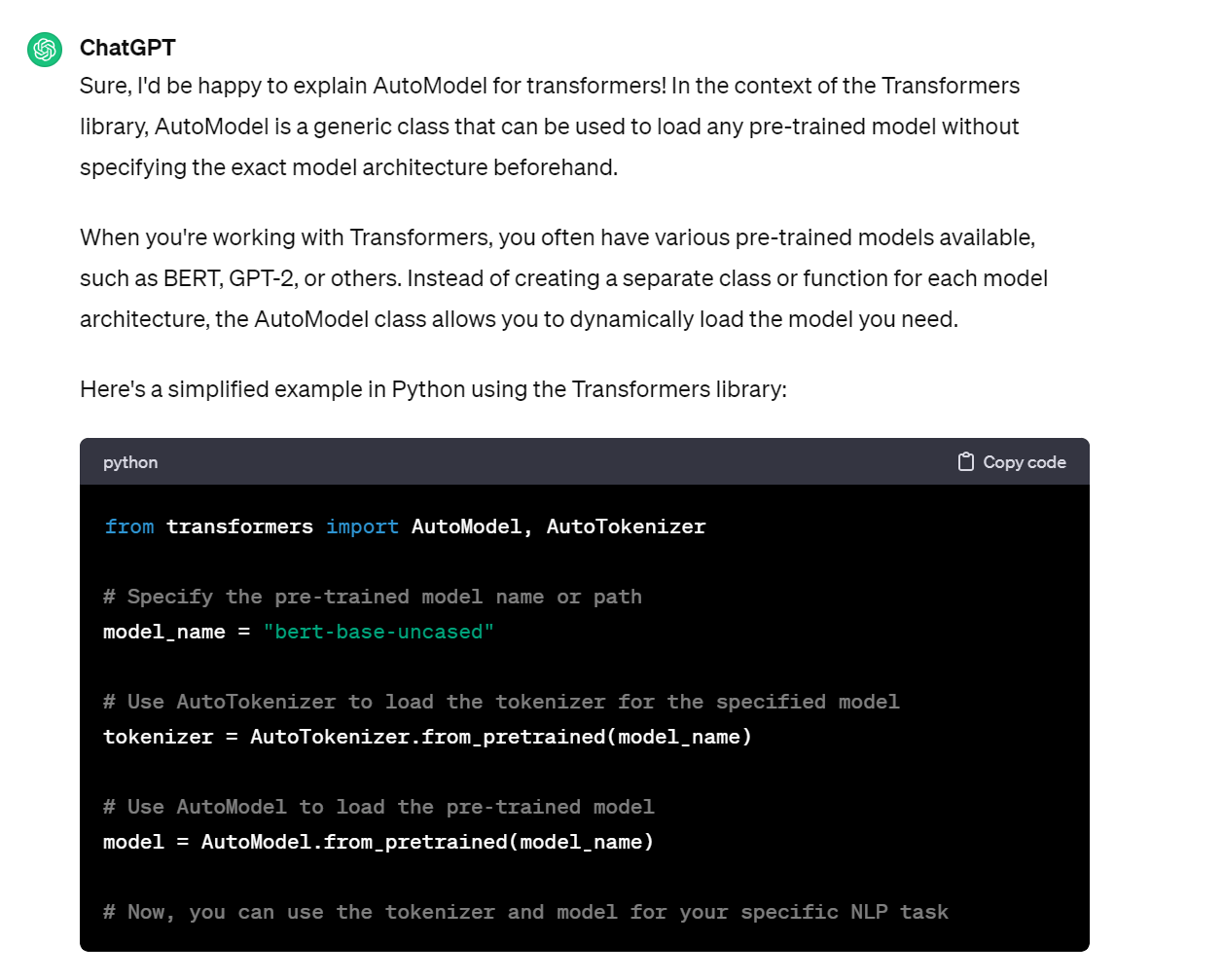
-check command again

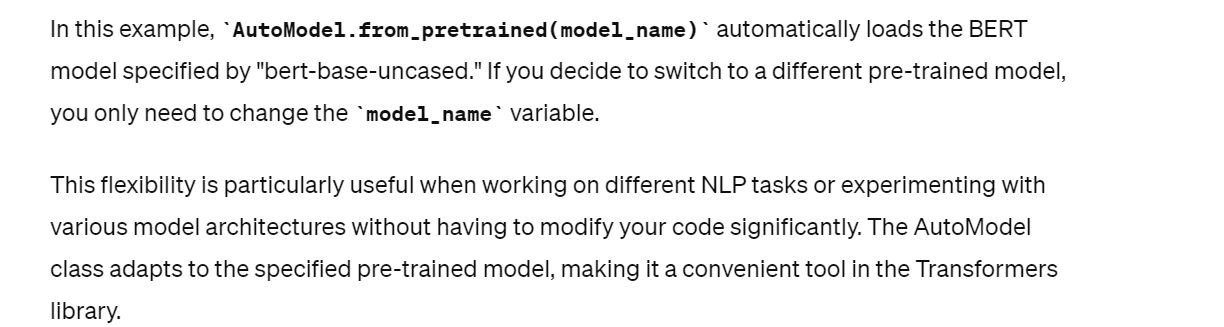
#overview of pipeline args

classifier = pipeline?

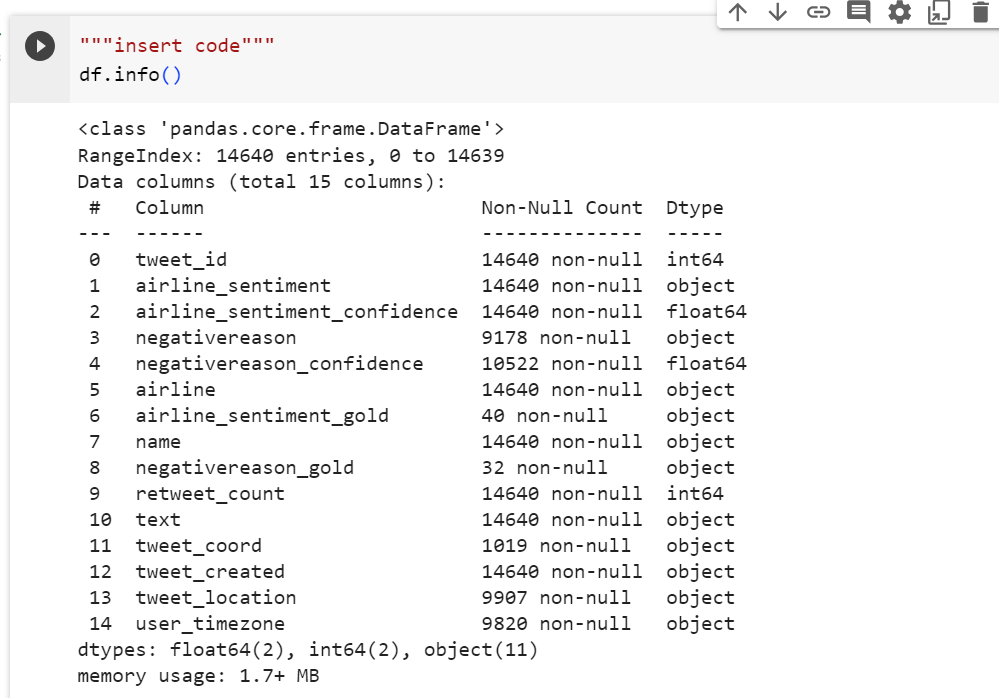
Leg automodel uit

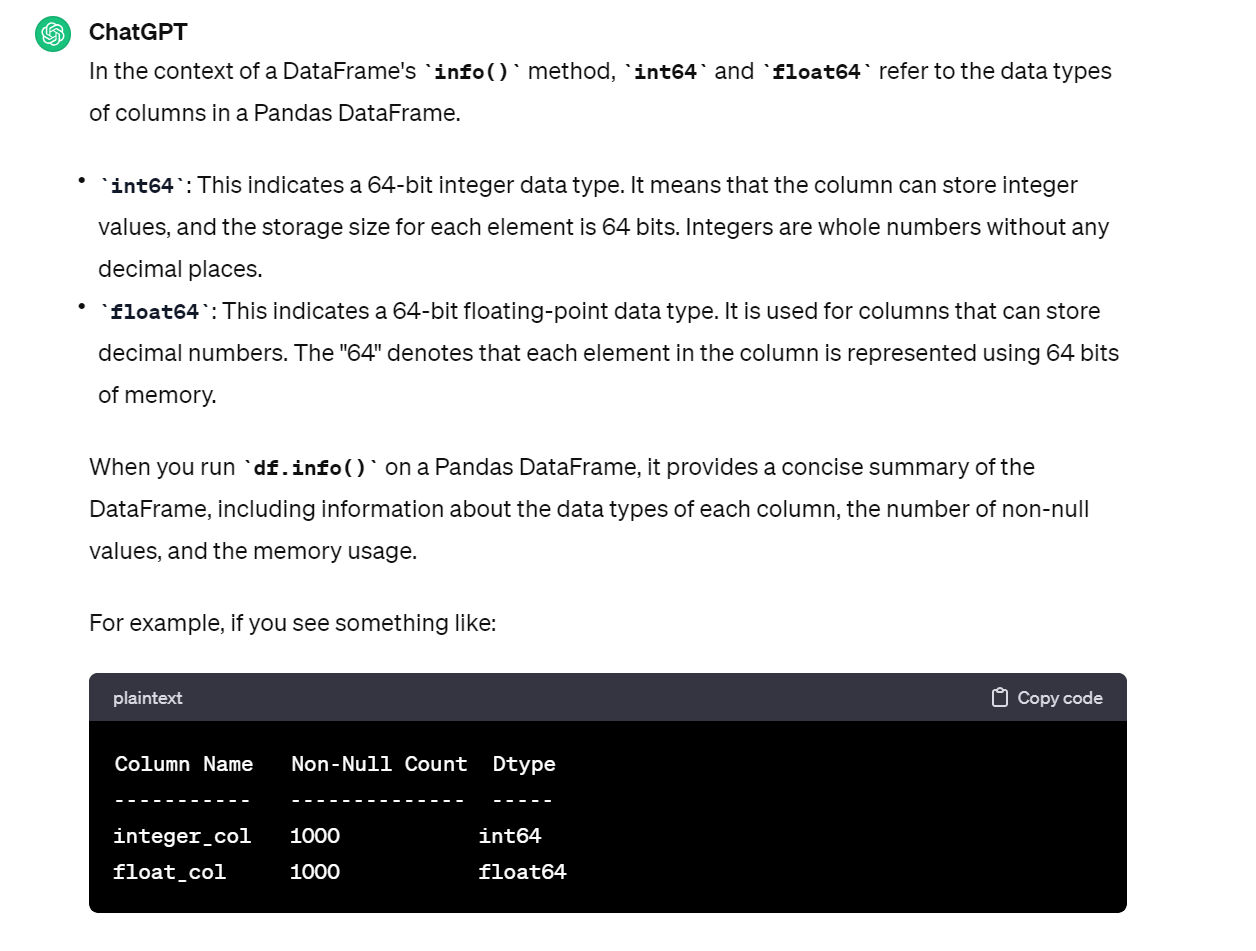
from transformers import AutoModel

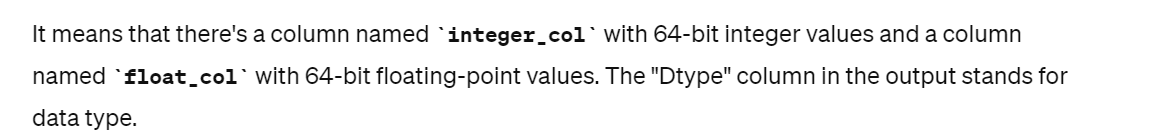




Explain int64







Wat betekent non null count

Non null count : er ontbreken geen data

Zeg studenten : sentiment label = airline sentiment

Leg random\_state uit

6.1 niet, geen conf measures

Voorbeelden voor neutral

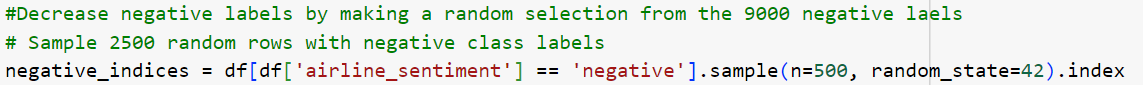
She is quiet.

He is quiet.

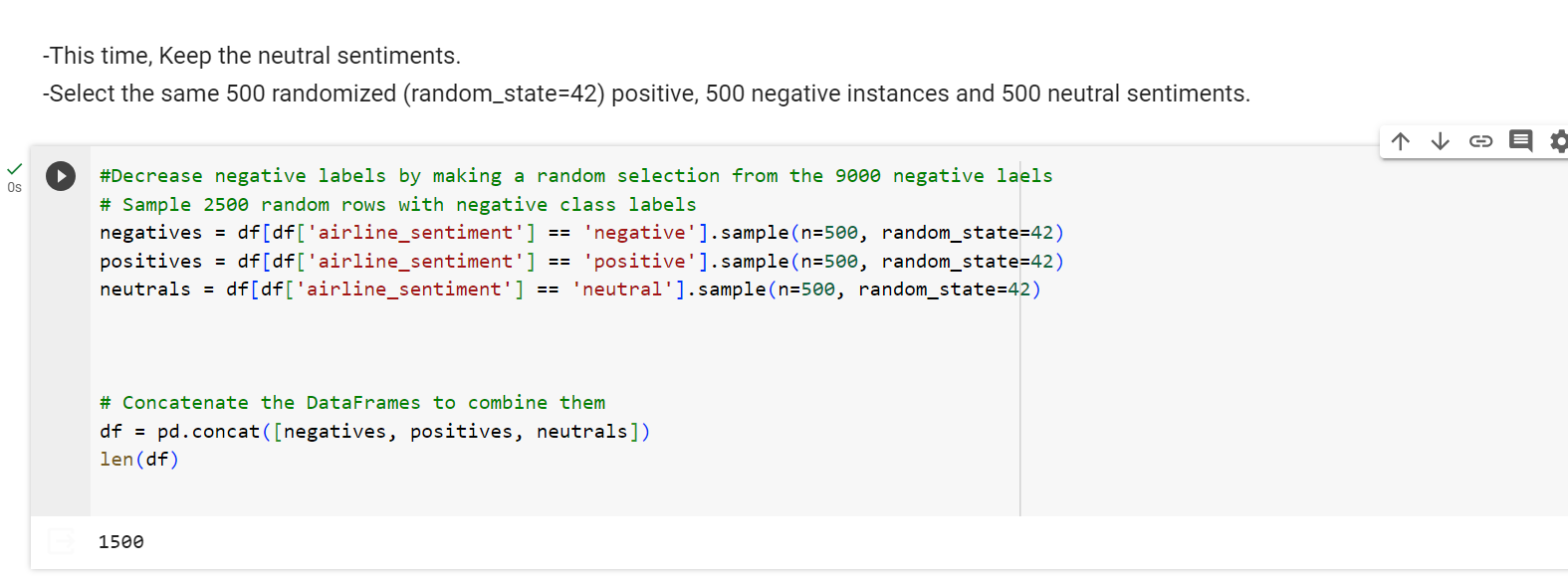
He is studying

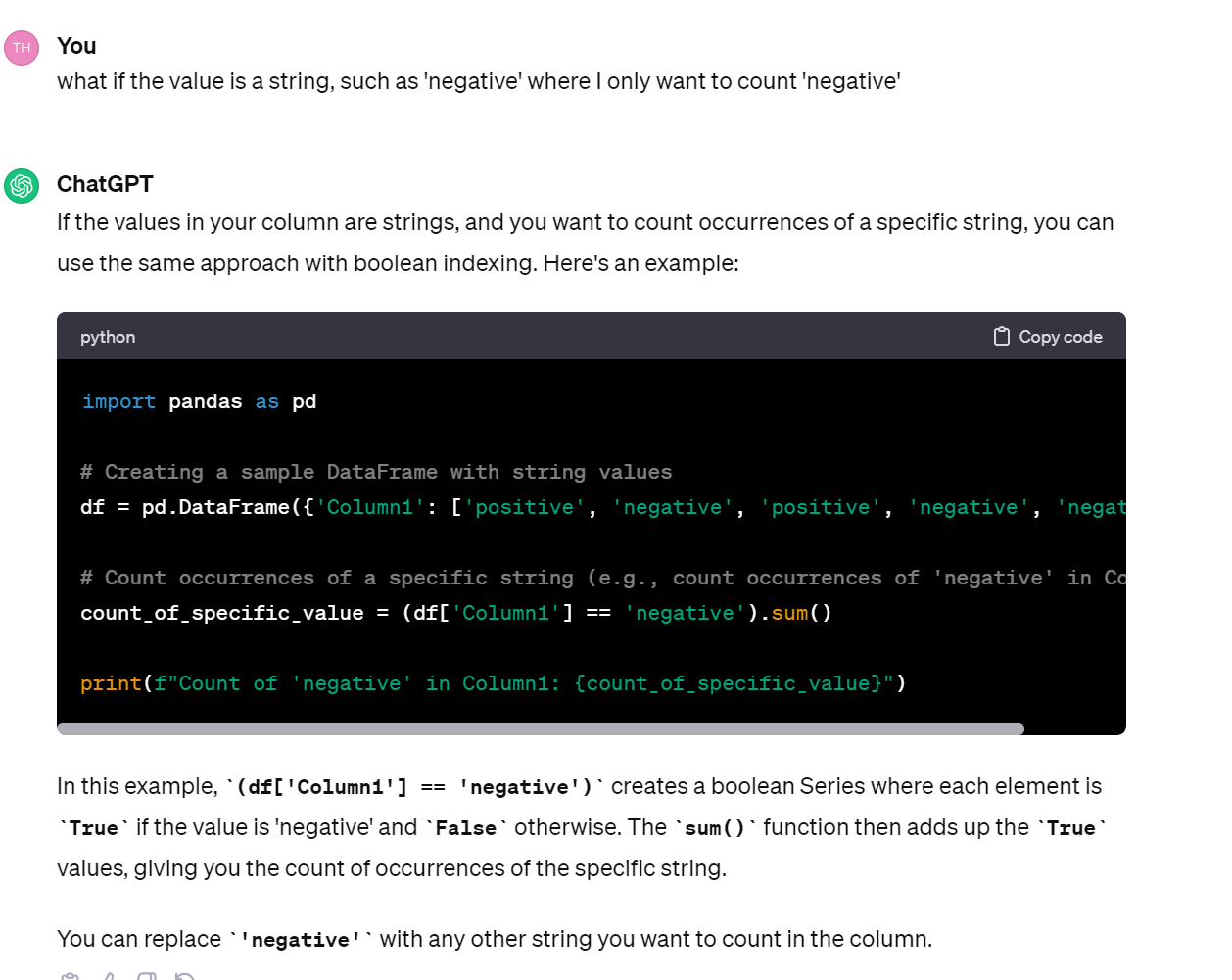
The car is white

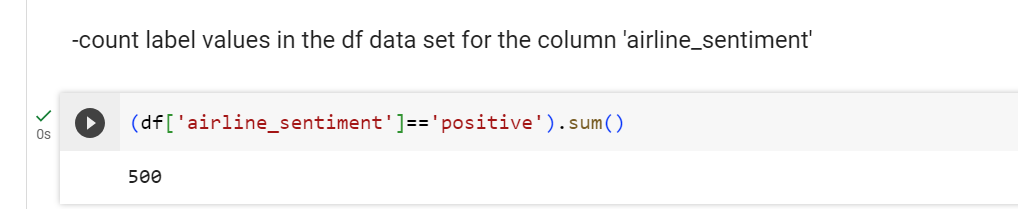
TD made a mistake in



Ask students to find shorter way for selecting pos neg neutr







Check nog eens verschil loc vs iloc

