ScaDS.AI
DRESDEN LEIPZIG

CENTER FOR SCALABLE DATA ANALYTICS AND ARTIFICIAL INTELLIGENCE

Bio-image Analysis with the Help of Large Language Models

Robert Haase

Funded by

Bundesministerium
für Bildung
und Forschung

SACHSEN

Diese Maßnahme wird gefördert durch die Bundesregierung aufgrund eines Beschlusses des Deutschen Bundestages. Diese Maßnahme wird mitfinanziert durch Steuermittel auf der Grundlage des von den Abgeordneten des Sächsischen Landtags beschlossenen Haushaltes.

These slides can be reused under the terms of the CC-BY 4.0 license unless mentioned otherwise.
https://doi.org/10.5281/zenodo.10811747

Quick survey

Question	Number of raised hands
Who has tried ChatGPT, Dall-E or other large language model based tools before?	
Who is using them on a daily basis?	
Who knows how they work under the hood (roughly)?	

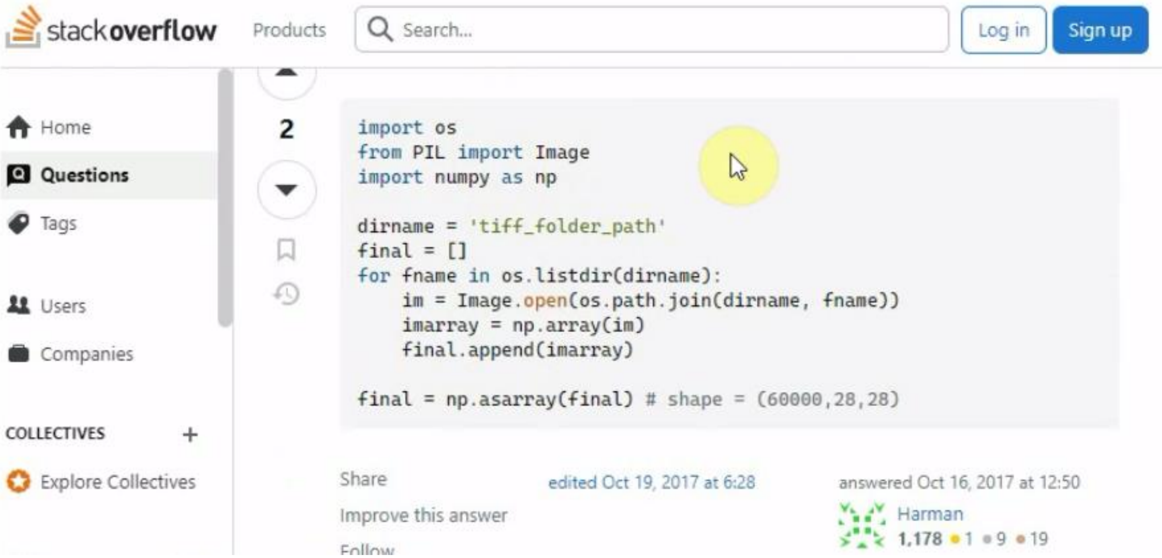
https://doi.org/10.5281/zenodo.10811747

Artificial intelligence in daily programmer's life

How it started

Artificial intelligence in daily programmer’s life

How it started



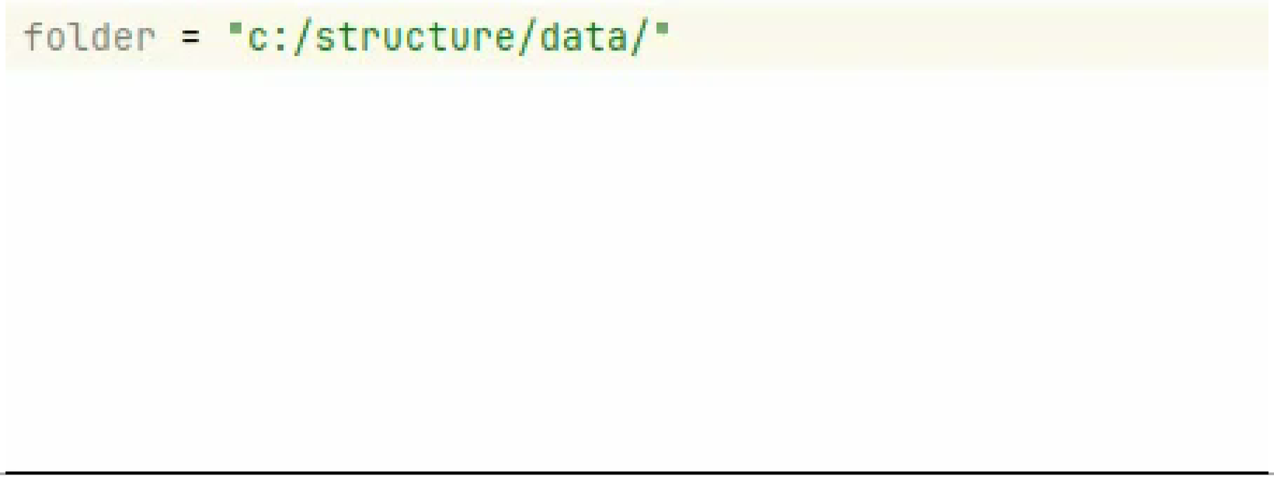
```
import os
from PIL import Image
import numpy as np

dirname = 'tiff_folder_path'
final = []
for fname in os.listdir(dirname):
    im = Image.open(os.path.join(dirname, fname))
    imarray = np.array(im)
    final.append(imarray)

final = np.asarray(final) # shape = (60000,28,28)
```

edited Oct 19, 2017 at 6:28

answered Oct 16, 2017 at 12:50
Harman
1,178 ●1 ●9 ●19

How it’s going

(GitHub copilot)

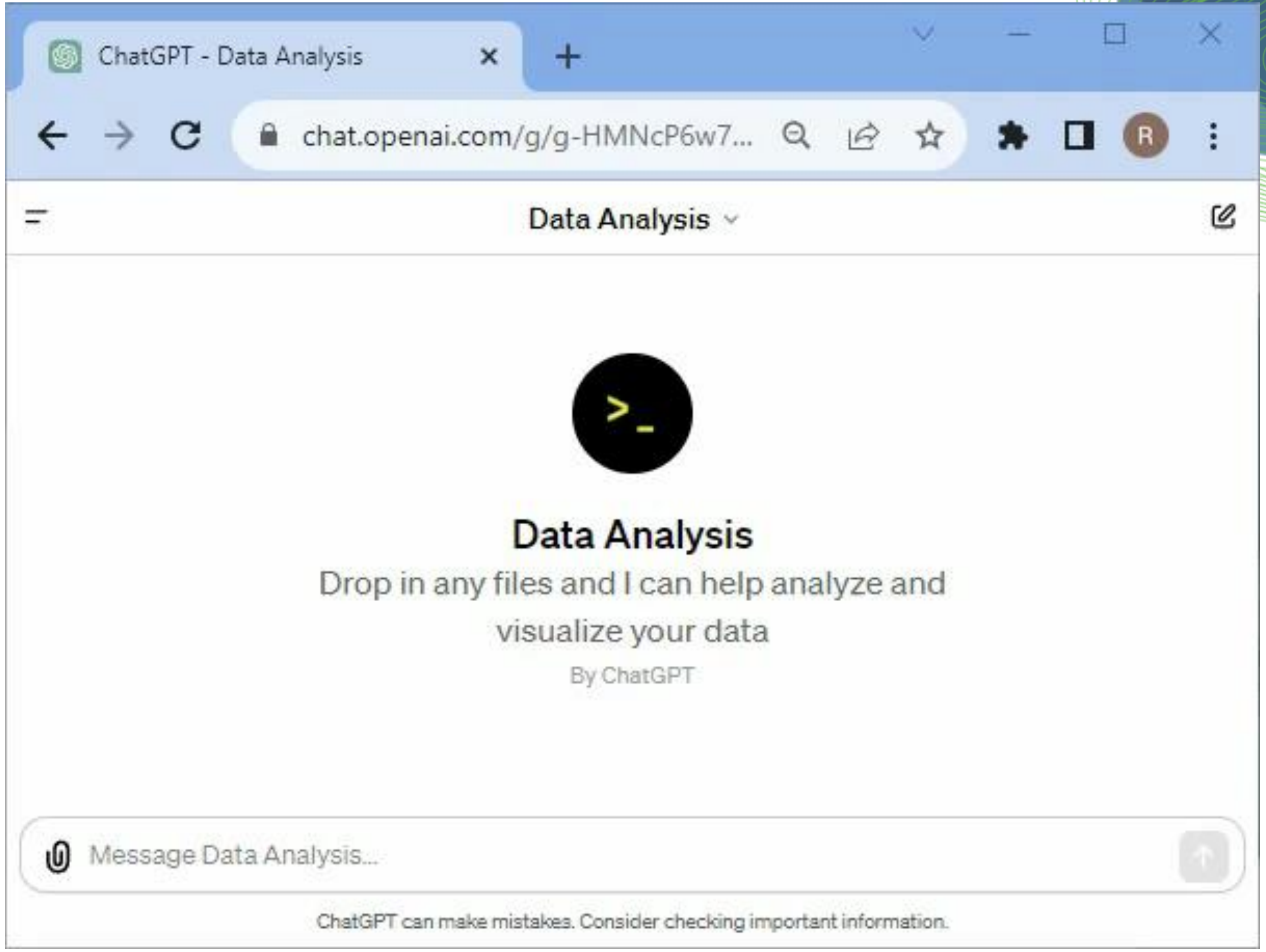


```
folder = "c:/structure/data/"
```

Prompting to analyze data

… using ChatGPT’s Data Analysis

- access data via prompts
- super easy to use
- requires uploading our data
- is a bit untransparent



Data Analysis

Drop in any files and I can help analyze and visualize your data

By ChatGPT

ChatGPT can make mistakes. Consider checking important information.

Data source: [videogame sales dataset by Gregory Smith](), licensed [CC-BY 4.0]()

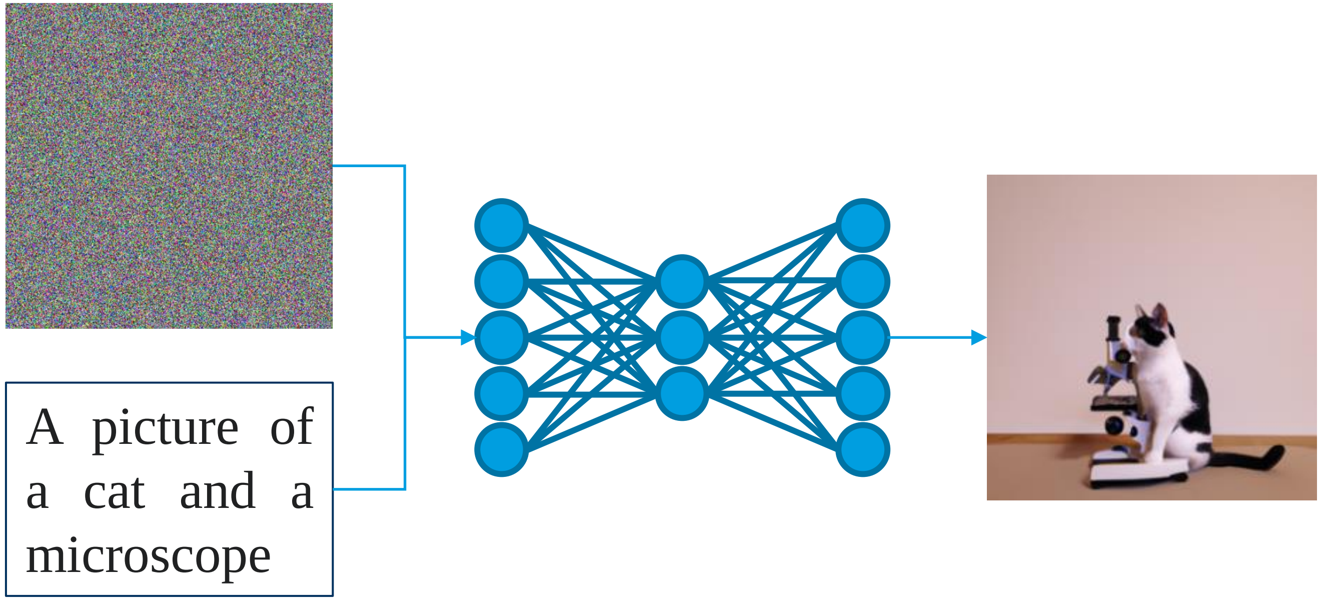
Short detour: Generative Artificial Intelligence

Definition: “Generative artificial intelligence […] is a type of artificial intelligence (AI) system capable of generating text, images, or other media in response to prompts.”[1]

Commonly based on Neural Networks

Bridges fields:
- Natural Language Processing (NLP)
- Computer Vision (CV)

Use-cases
- Translating text
- Writing emails, text, grant proposals
- Summarizing articles
- Writing code
- General question answering
- Image generation



A picture of a cat and a microscope

---

[1] source: https://en.wikipedia.org/wiki/Generative_artificial_intelligence

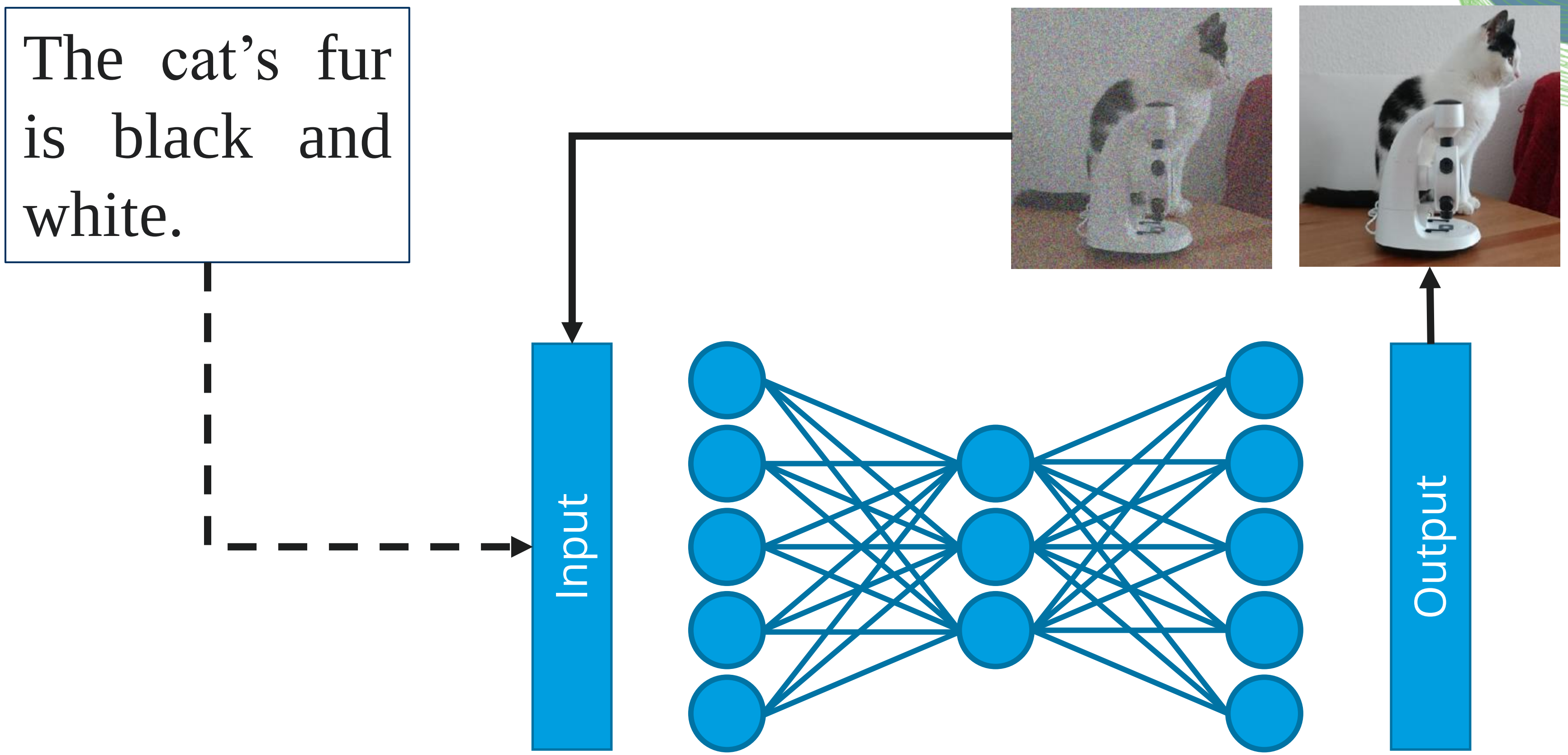
How does it work?

Combination of neural networks + other elements + various [multi-modal] data sources

Examples: GPT / DALL-E, Stable Diffusion

Use-case:

- Generate image from noise + text



The cat's fur is black and white.

Input

Output

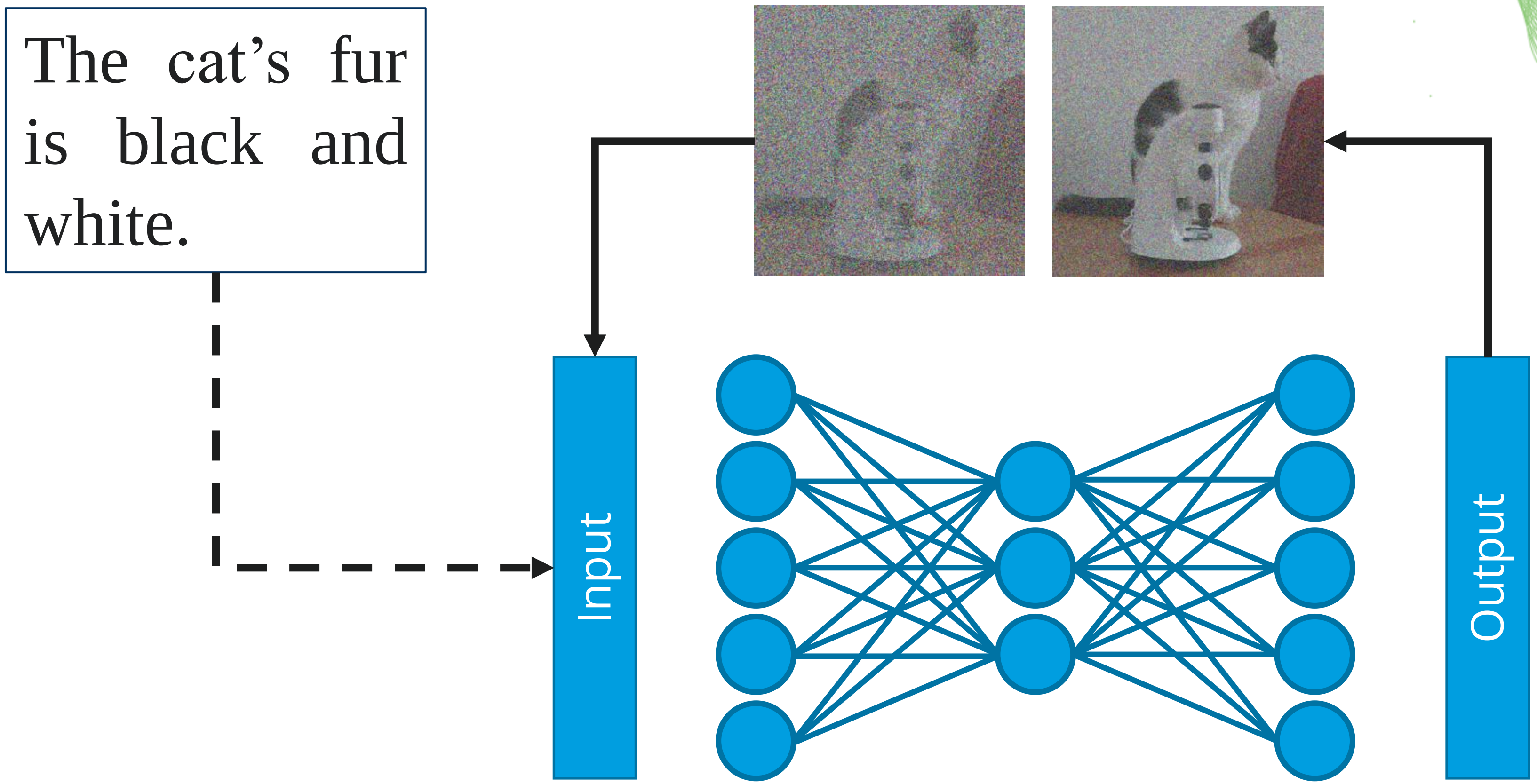
How does it work?

Combination of neural networks + other elements + various [multi-modal] data sources

Examples: GPT / DALL-E, Stable Diffusion

Use-case:

- Generate image from noise + text



The cat’s fur is black and white.

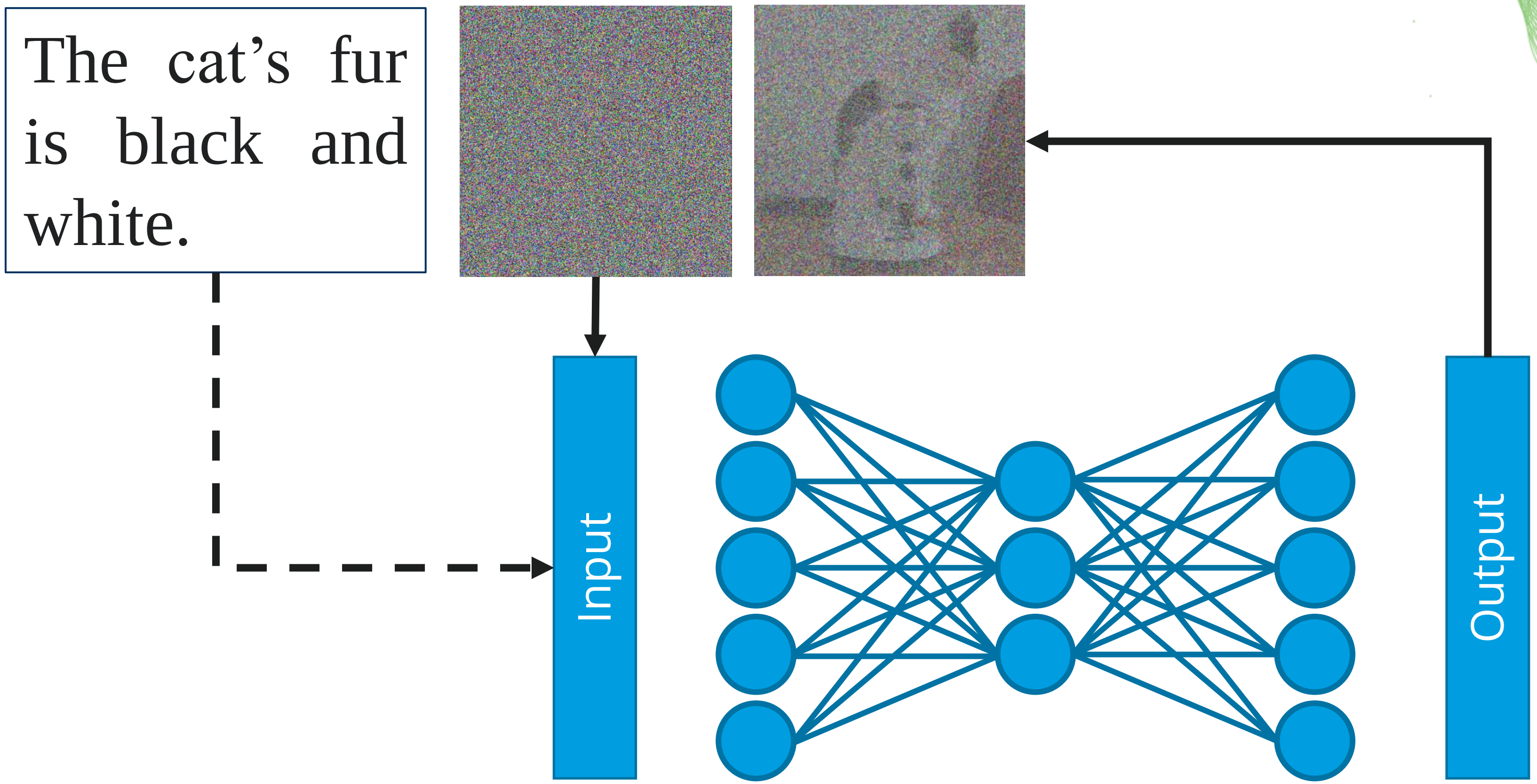
How does it work?

Combination of neural networks + other elements + various [multi-modal] data sources

Examples: GPT / DALL-E, Stable Diffusion

Use-case:

- Generate image from noise + text



The cat’s fur is black and white.

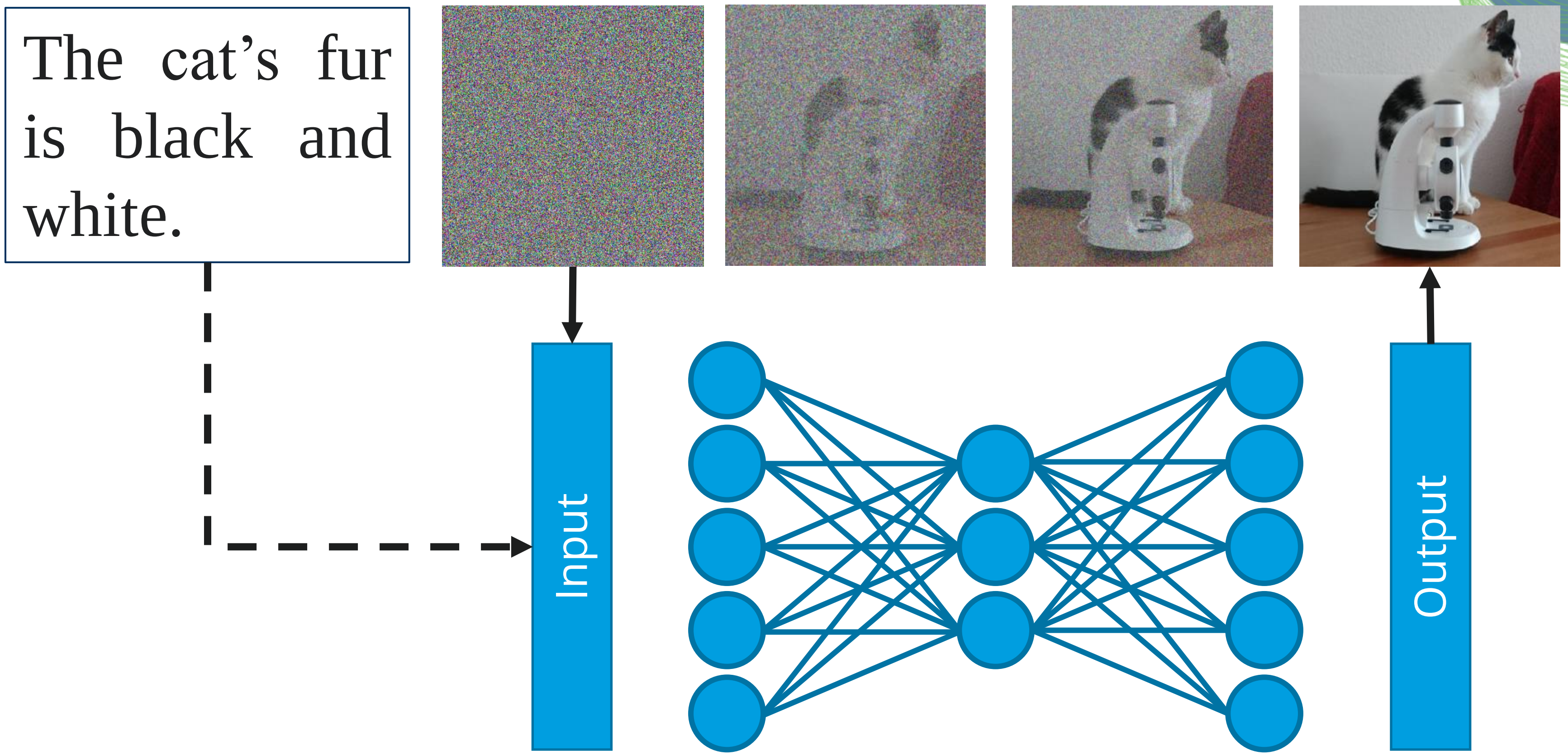
How does it work?

Combination of neural networks + other elements + various [multi-modal] data sources

Examples: GPT / DALL-E, Stable Diffusion

Use-case:

- Generate image from noise + text



The cat’s fur is black and white.

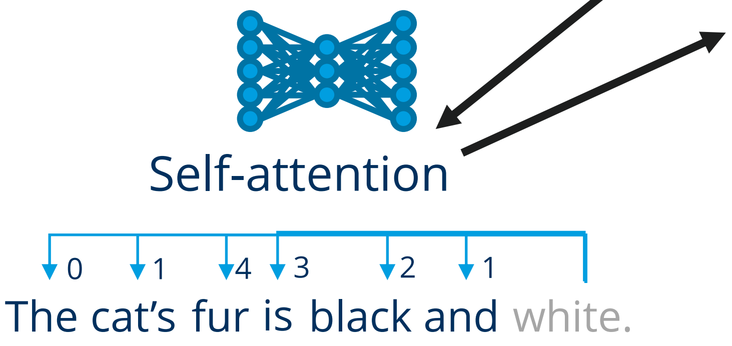
How does it work?

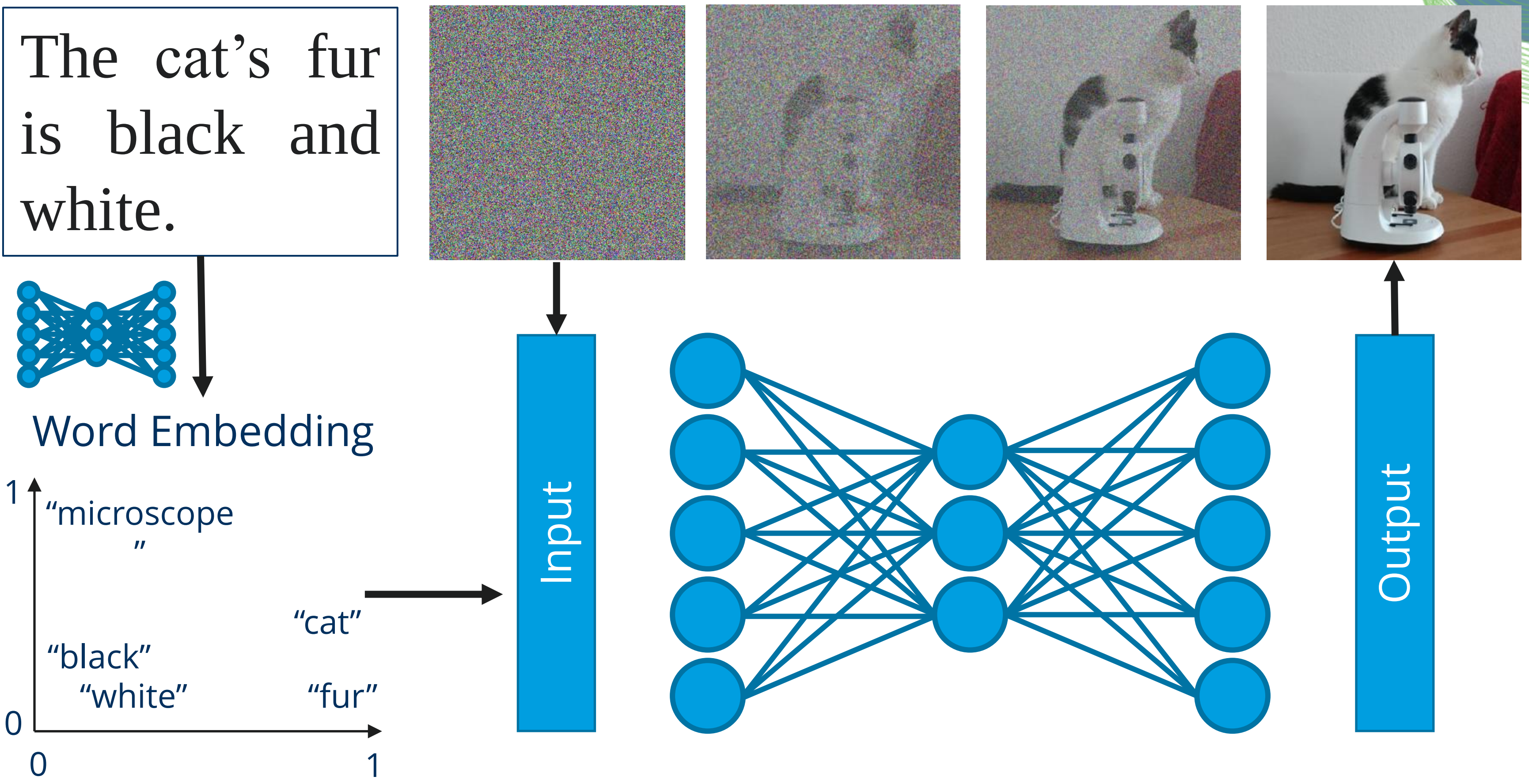
Combination of neural networks + other elements + various [multi-modal] data sources

Examples: GPT / DALL-E, Stable Diffusion

Use-case:

- Generate image from noise + text

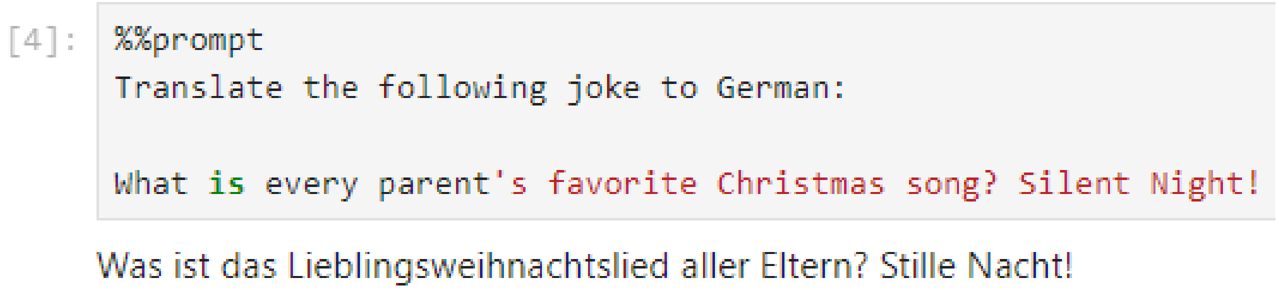




The cat's fur is black and white.

What are large language models good in?

Translation tasks

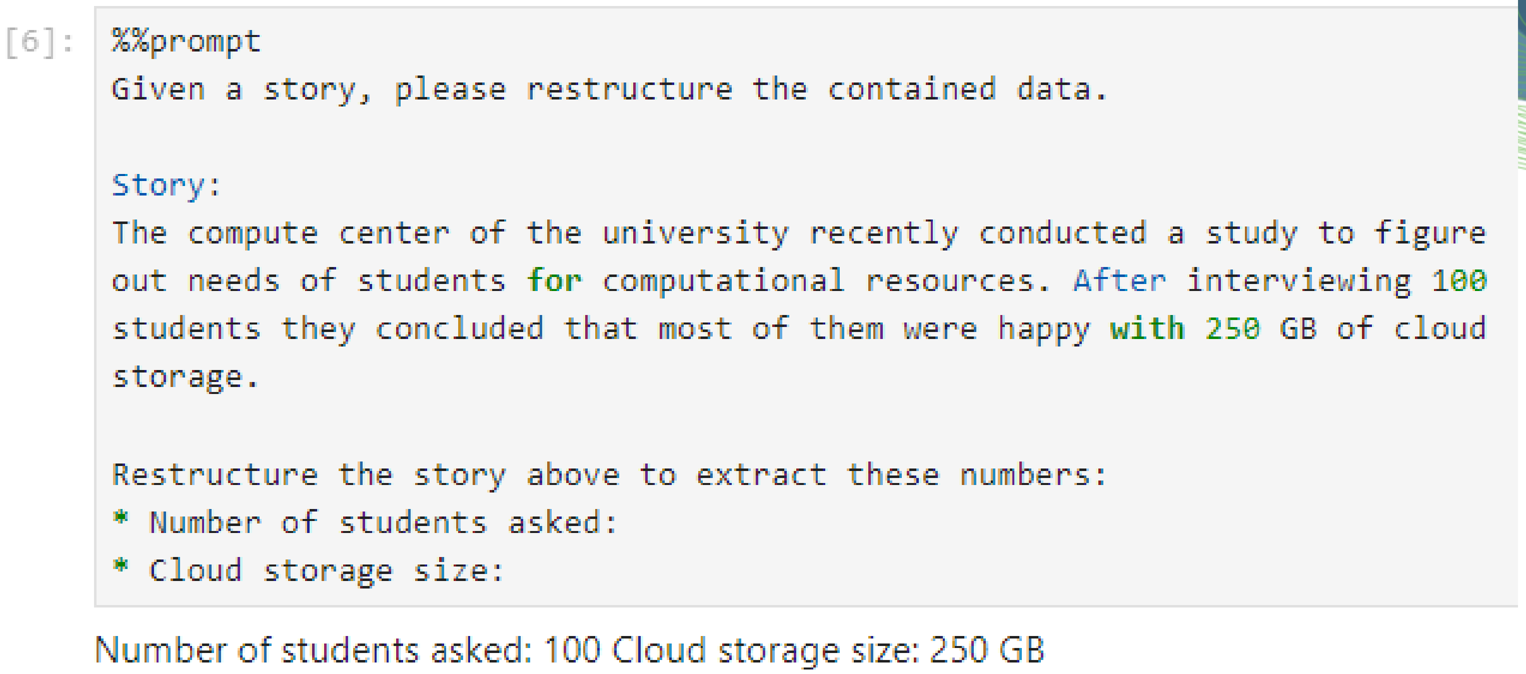


```
[4]: %%prompt
     Translate the following joke to German:

     What is every parent's favorite Christmas song? Silent Night!
```

Was ist das Lieblingsweihnachtslied aller Eltern? Stille Nacht!

What are large language models good in?

Restructuring data



```
[6]: %%prompt
     Given a story, please restructure the contained data.

     Story:
     The compute center of the university recently conducted a study to figure
     out needs of students for computational resources. After interviewing 100
     students they concluded that most of them were happy with 250 GB of cloud
     storage.

     Restructure the story above to extract these numbers:
     * Number of students asked:
     * Cloud storage size:
```

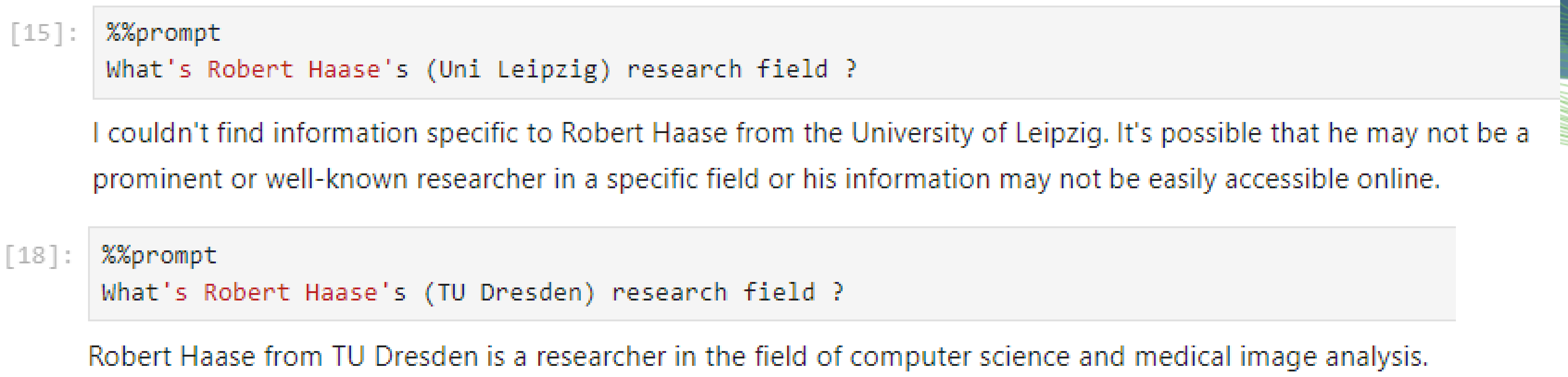
```
Number of students asked: 100 Cloud storage size: 250 GB
```

https://scads.github.io/prompt-engineering-tutorial-2023/01_prompts/02_use_cases.html

What are large language models not so good in?

Querying knowledge

(e.g. for recent information)



```
[15]: %%prompt
      What's Robert Haase's (Uni Leipzig) research field ?
```

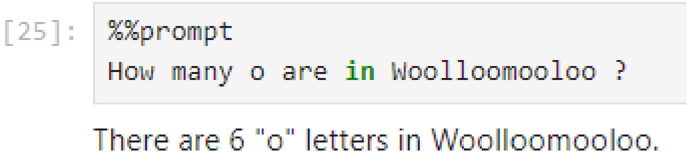
I couldn't find information specific to Robert Haase from the University of Leipzig. It's possible that he may not be a prominent or well-known researcher in a specific field or his information may not be easily accessible online.

```
[18]: %%prompt
      What's Robert Haase's (TU Dresden) research field ?
```

Robert Haase from TU Dresden is a researcher in the field of computer science and medical image analysis.

What are large language models not so good in?

Math (e.g. counting)



```
[25]: %%prompt
      How many o are in Woolloomooloo ?
```

There are 6 "o" letters in Woolloomooloo.

Woolloomooloo

New South Wales 2011

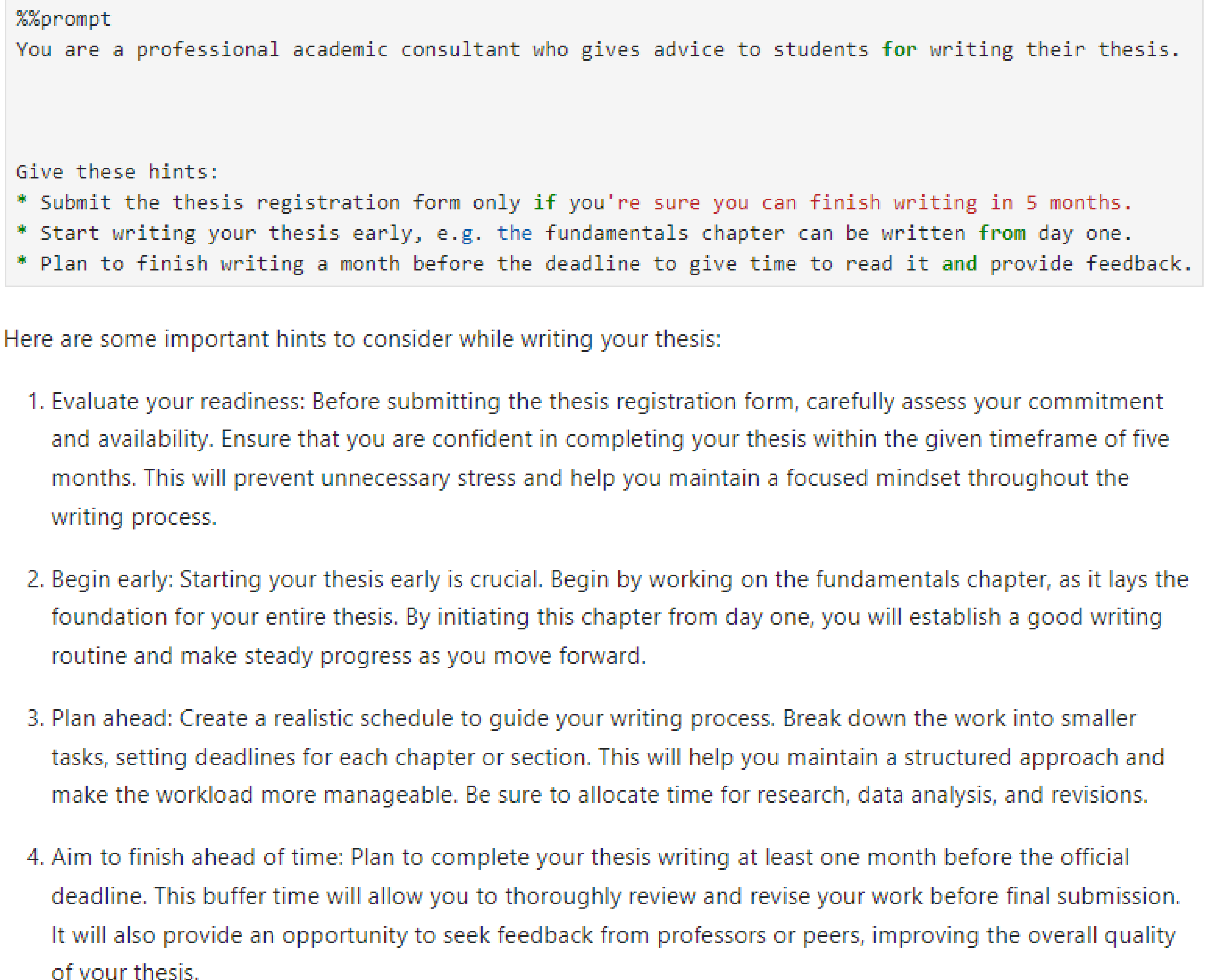
Australia

Partly cloudy · 23°C

9:24 PM

Prompt Engineering

- Set context / assign role
- Define output style
- Give hints / conditions
- List tasks
- Provide data



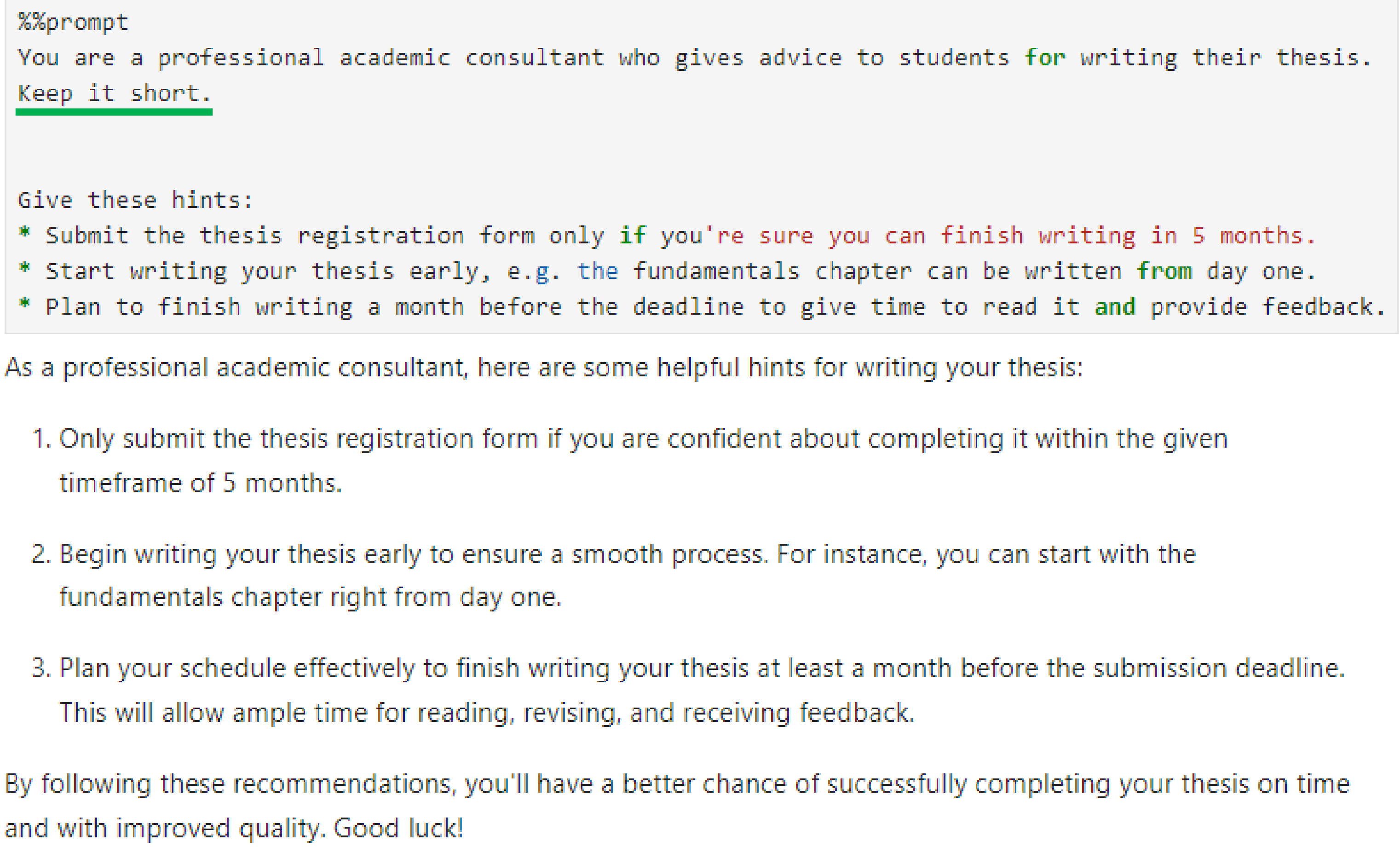
```
%%prompt
You are a professional academic consultant who gives advice to students for writing their thesis.



Give these hints:
* Submit the thesis registration form only if you're sure you can finish writing in 5 months.
* Start writing your thesis early, e.g. the fundamentals chapter can be written from day one.
* Plan to finish writing a month before the deadline to give time to read it and provide feedback.
```

Here are some important hints to consider while writing your thesis:

1. Evaluate your readiness: Before submitting the thesis registration form, carefully assess your commitment and availability. Ensure that you are confident in completing your thesis within the given timeframe of five months. This will prevent unnecessary stress and help you maintain a focused mindset throughout the writing process.
2. Begin early: Starting your thesis early is crucial. Begin by working on the fundamentals chapter, as it lays the foundation for your entire thesis. By initiating this chapter from day one, you will establish a good writing routine and make steady progress as you move forward.
3. Plan ahead: Create a realistic schedule to guide your writing process. Break down the work into smaller tasks, setting deadlines for each chapter or section. This will help you maintain a structured approach and make the workload more manageable. Be sure to allocate time for research, data analysis, and revisions.
4. Aim to finish ahead of time: Plan to complete your thesis writing at least one month before the official deadline. This buffer time will allow you to thoroughly review and revise your work before final submission. It will also provide an opportunity to seek feedback from professors or peers, improving the overall quality of your thesis.

Prompt Engineering

- Set context / assign role
- Define output style
- Give hints / conditions
- List tasks
- Provide data



```
%%prompt
You are a professional academic consultant who gives advice to students for writing their thesis.
Keep it short.


Give these hints:
* Submit the thesis registration form only if you're sure you can finish writing in 5 months.
* Start writing your thesis early, e.g. the fundamentals chapter can be written from day one.
* Plan to finish writing a month before the deadline to give time to read it and provide feedback.
```

As a professional academic consultant, here are some helpful hints for writing your thesis:

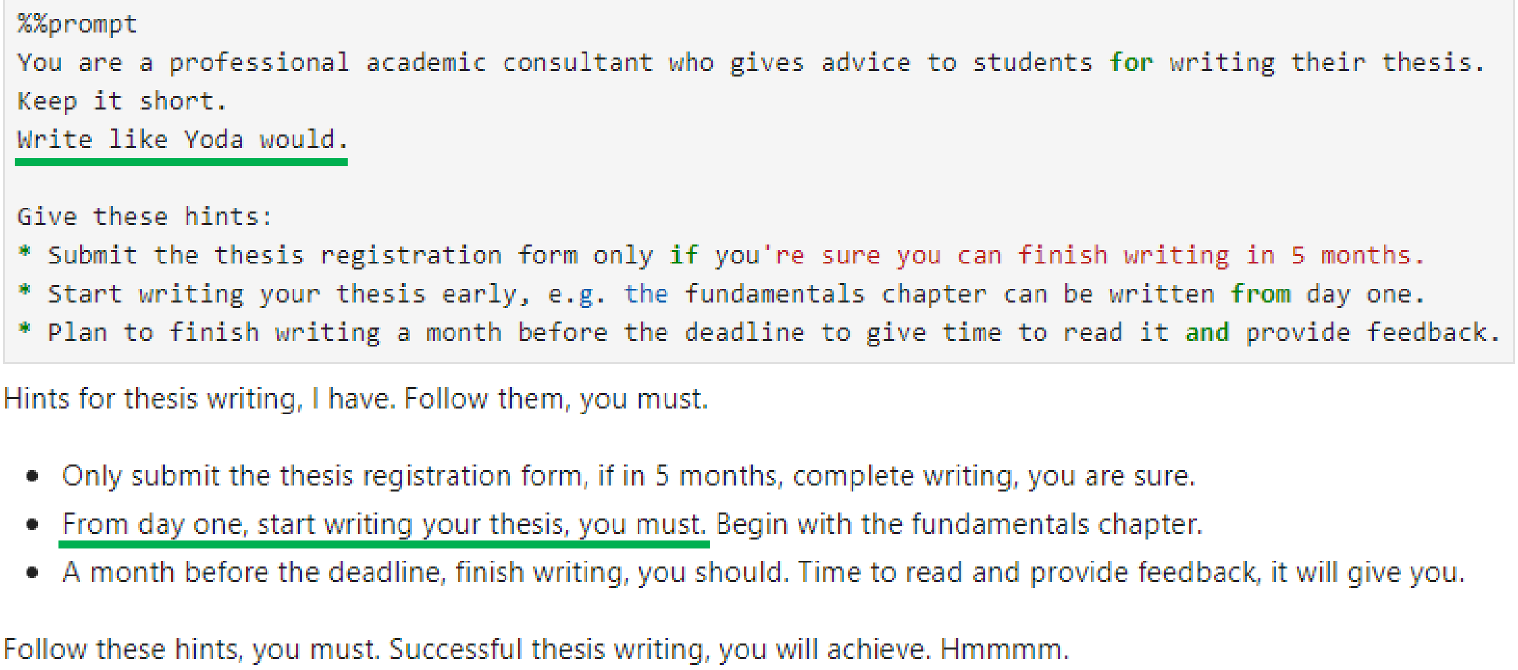
1. Only submit the thesis registration form if you are confident about completing it within the given timeframe of 5 months.
2. Begin writing your thesis early to ensure a smooth process. For instance, you can start with the fundamentals chapter right from day one.
3. Plan your schedule effectively to finish writing your thesis at least a month before the submission deadline. This will allow ample time for reading, revising, and receiving feedback.

By following these recommendations, you'll have a better chance of successfully completing your thesis on time and with improved quality. Good luck!

https://scads.github.io/prompt-engineering-tutorial-2023/01_prompts/03_prompt_engineering.html

Prompt Engineering

- Set context / assign role
- Define output style
- Give hints / conditions
- List tasks
- Provide data



```
%%prompt
You are a professional academic consultant who gives advice to students for writing their thesis.
Keep it short.
Write like Yoda would.

Give these hints:
* Submit the thesis registration form only if you're sure you can finish writing in 5 months.
* Start writing your thesis early, e.g. the fundamentals chapter can be written from day one.
* Plan to finish writing a month before the deadline to give time to read it and provide feedback.
```

Hints for thesis writing, I have. Follow them, you must.

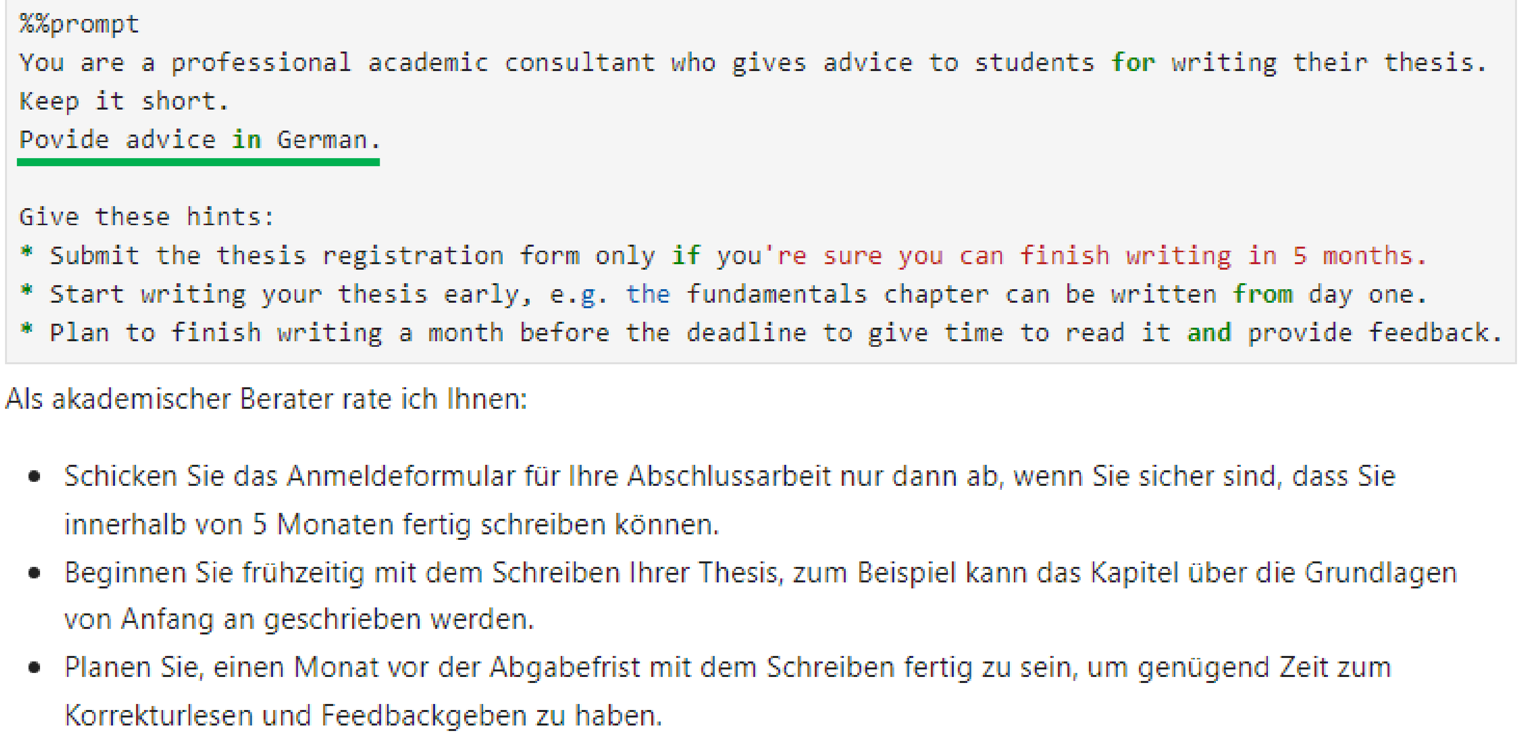
- Only submit the thesis registration form, if in 5 months, complete writing, you are sure.
- From day one, start writing your thesis, you must. Begin with the fundamentals chapter.
- A month before the deadline, finish writing, you should. Time to read and provide feedback, it will give you.

Follow these hints, you must. Successful thesis writing, you will achieve. Hmmmm.

https://scads.github.io/prompt-engineering-tutorial-2023/01_prompts/03_prompt_engineering.html

Prompt Engineering

- Set context / assign role
- Define output style
- Give hints / conditions
- List tasks
- Provide data



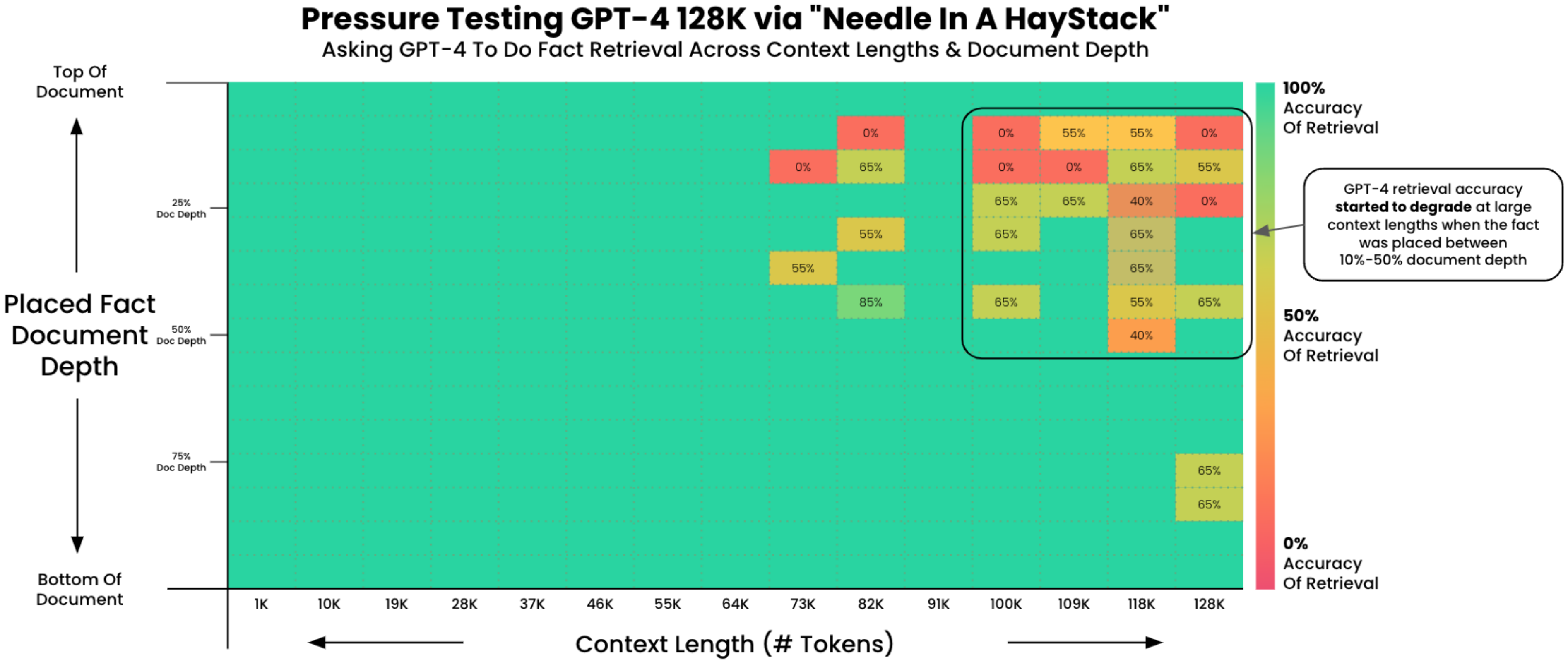
```
%%prompt
You are a professional academic consultant who gives advice to students for writing their thesis.
Keep it short.
Povide advice in German.

Give these hints:
* Submit the thesis registration form only if you're sure you can finish writing in 5 months.
* Start writing your thesis early, e.g. the fundamentals chapter can be written from day one.
* Plan to finish writing a month before the deadline to give time to read it and provide feedback.
```

Als akademischer Berater rate ich Ihnen:

- Schicken Sie das Anmeldeformular für Ihre Abschlussarbeit nur dann ab, wenn Sie sicher sind, dass Sie innerhalb von 5 Monaten fertig schreiben können.
- Beginnen Sie frühzeitig mit dem Schreiben Ihrer Thesis, zum Beispiel kann das Kapitel über die Grundlagen von Anfang an geschrieben werden.
- Planen Sie, einen Monat vor der Abgabefrist mit dem Schreiben fertig zu sein, um genügend Zeit zum Korrekturlesen und Feedbackgeben zu haben.

https://scads.github.io/prompt-engineering-tutorial-2023/01_prompts/03_prompt_engineering.html

Context length



Pressure Testing GPT-4 128K via "Needle In A HayStack"

Asking GPT-4 To Do Fact Retrieval Across Context Lengths & Document Depth

GPT-4 retrieval accuracy **started to degrade** at large context lengths when the fact was placed between 10%-50% document depth

Goal: Test GPT-4 Ability To Retrieve Information From Large Context Windows

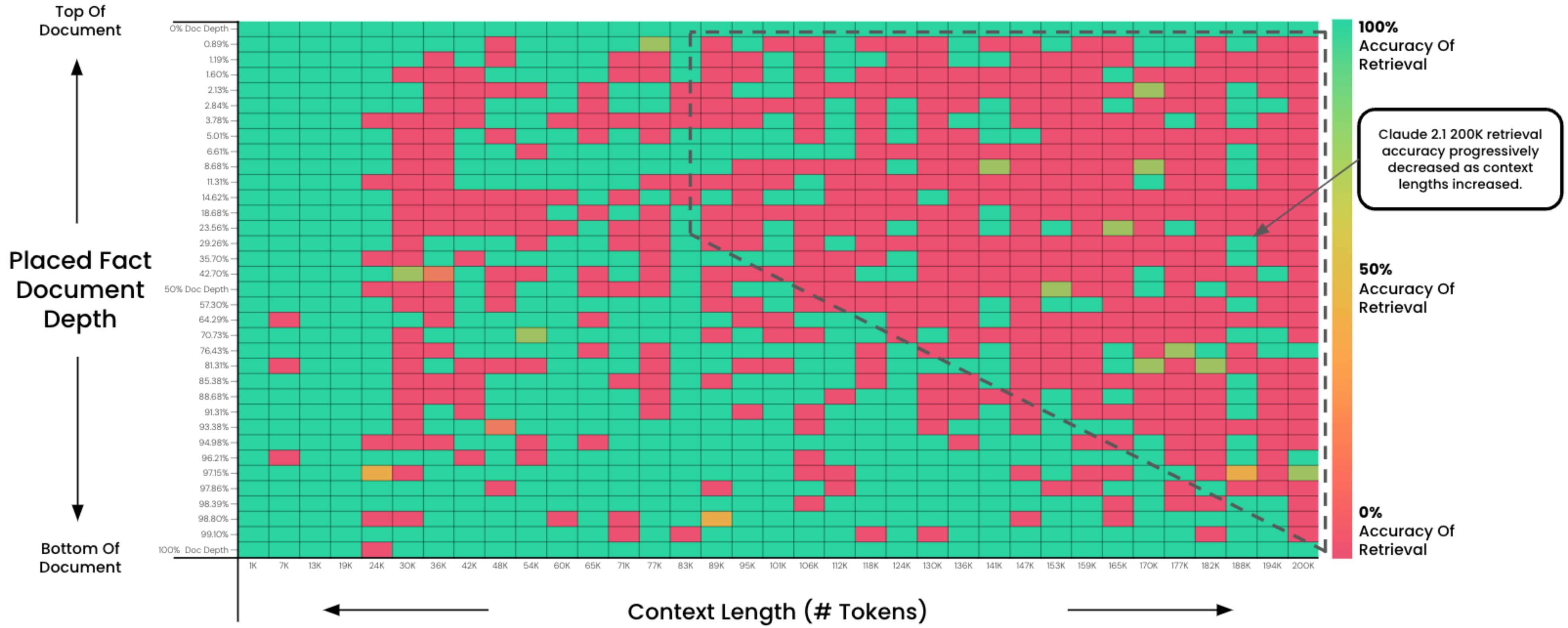
A fact was placed within a document. GPT-4 (1106-preview) was then asked to retrieve it. The output was evaluated for accuracy. This test was run at 15 different document depths (top > bottom) and 15 different context lengths (1K >128K tokens). 2x tests were run for larger contexts for a larger sample size.

Visualization by Greg Kamradt, Licensed MIT (modified), Source: https://github.com/gkamradt/LLMTest_NeedleInAHaystack

Context length

Pressure Testing Claude-2.1 200K via "Needle In A HayStack"

Asking Claude 2.1 To Do Fact Retrieval Across Context Lengths & Document Depth



Claude 2.1 200K retrieval accuracy progressively decreased as context lengths increased.

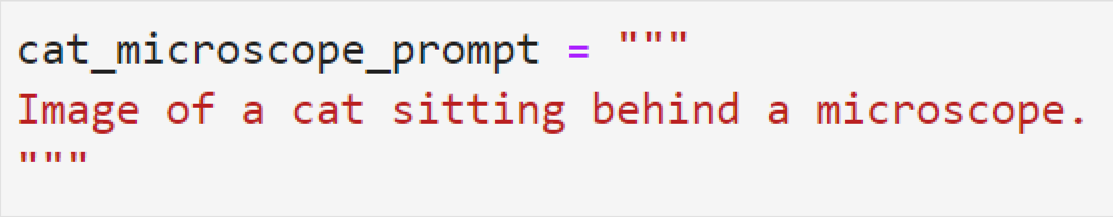
Goal: Test Claude 2.1 Ability To Retrieve Information From Large Context Windows

A fact was placed within a document. Claude 2.1 (200K) was then asked to retrieve it. The output was evaluated (with GPT-4) for accuracy.
This test was run at 35 different document depths (top › bottom) and 35 different context lengths (1K ›200K tokens).
Document Depths followed a sigmoid distribution

Visualization by Greg Kamradt, Licensed MIT (modified), Source: https://github.com/gkamradt/LLMTest_NeedleInAHaystack

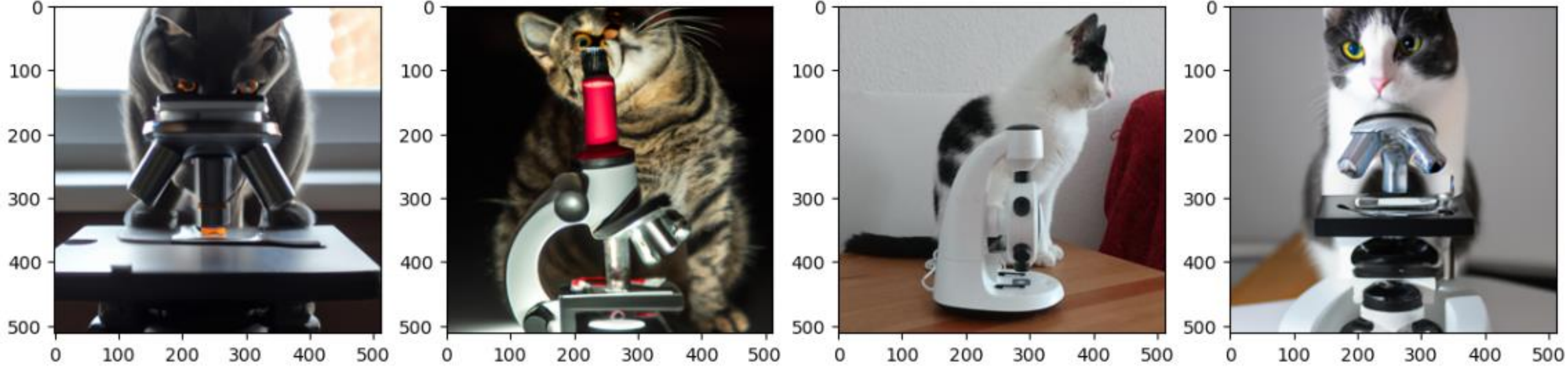
Prompt engineering

Prompts can also be used to generate images, e.g. with DALL-E.
One can generate quite realistic images given a detailed prompt.



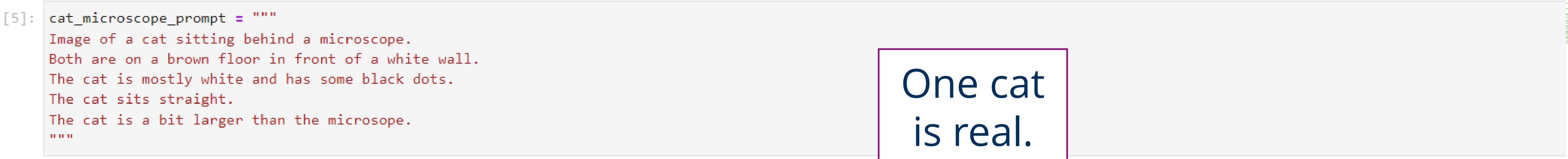
```
cat_microscope_prompt = """
Image of a cat sitting behind a microscope.
"""
```

One cat is real.



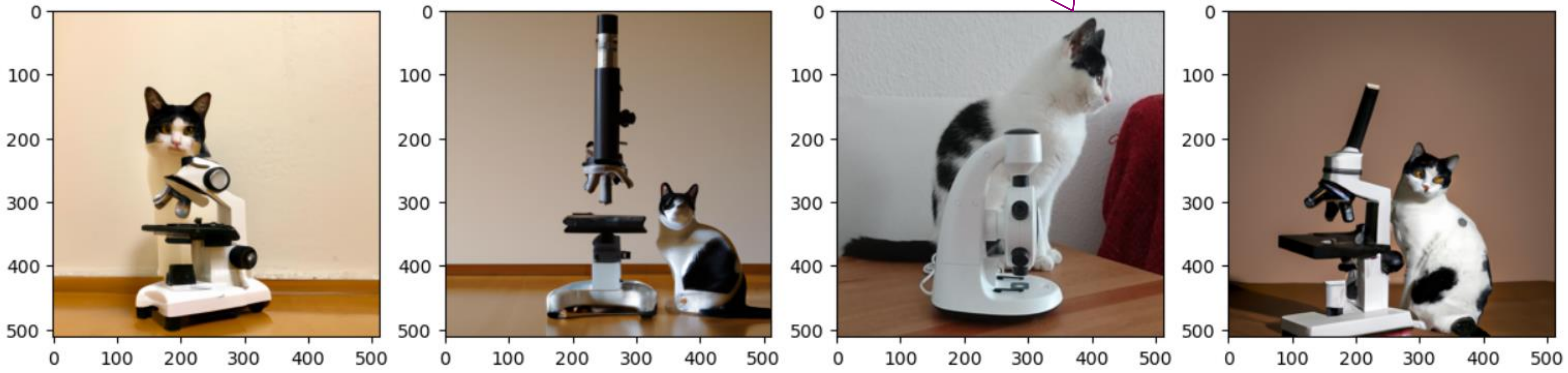
Prompt engineering

Prompts can also be used to generate images, e.g. with DALL-E.
One can generate quite realistic images given a detailed prompt.



```
[5]: cat_microscope_prompt = """
Image of a cat sitting behind a microscope.
Both are on a brown floor in front of a white wall.
The cat is mostly white and has some black dots.
The cat sits straight.
The cat is a bit larger than the microsope.
"""
```

One cat is real.

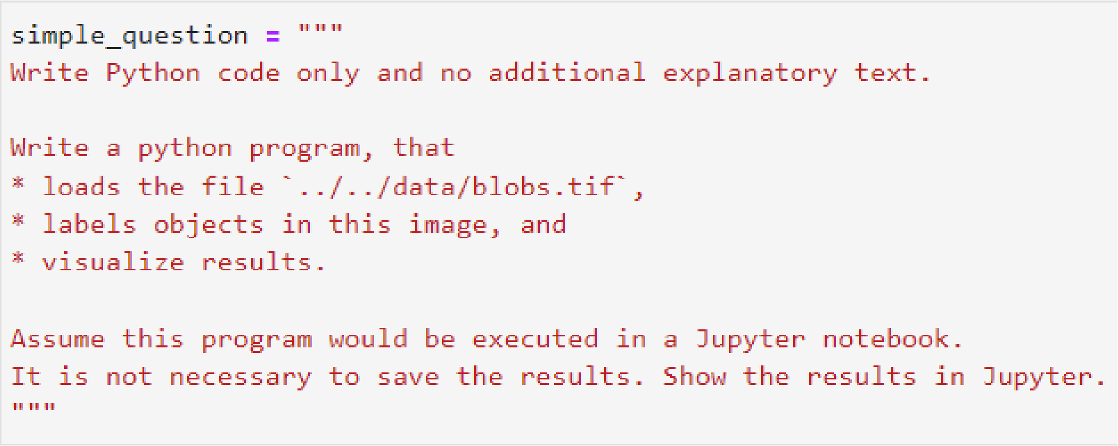


https://haesleinhuepf.github.io/BioImageAnalysisNotebooks/07_prompt_engineering/02_generating_images.html

Prompt engineering

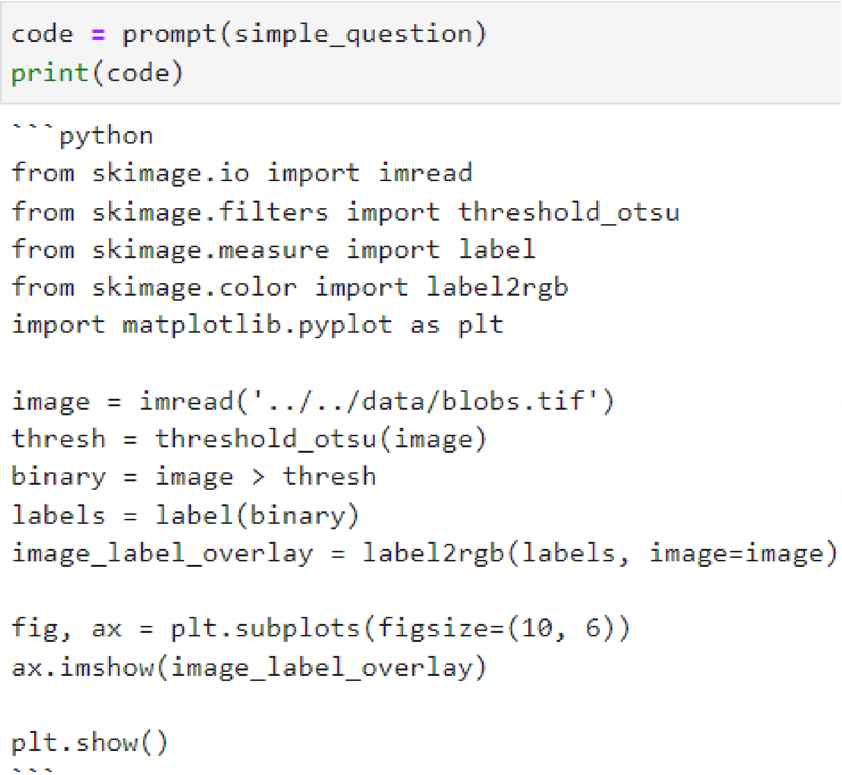
chatGPT can solve simple image analysis tasks

Prompt



````
simple_question = """
Write Python code only and no additional explanatory text.

Write a python program, that
* loads the file `../../data/blobs.tif`,
* labels objects in this image, and
* visualize results.

Assume this program would be executed in a Jupyter notebook.
It is not necessary to save the results. Show the results in Jupyter.
"""
````

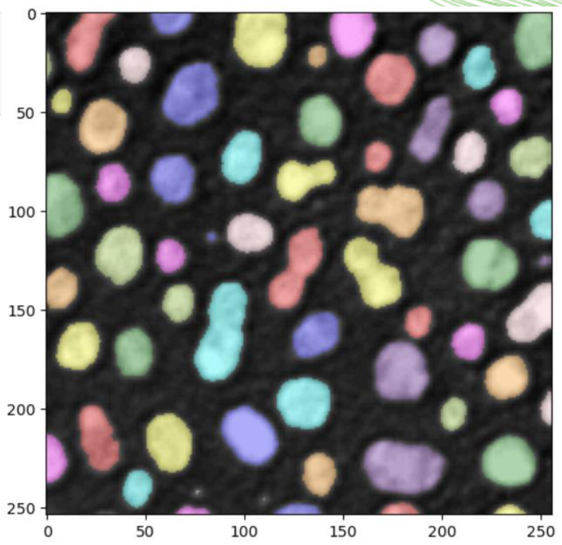
Response



````
code = prompt(simple_question)
print(code)
````

~~~
```python
from skimage.io import imread
from skimage.filters import threshold_otsu
from skimage.measure import label
from skimage.color import label2rgb
import matplotlib.pyplot as plt

image = imread('../../data/blobs.tif')
thresh = threshold_otsu(image)
binary = image > thresh
labels = label(binary)
image_label_overlay = label2rgb(labels, image=image)

fig, ax = plt.subplots(figsize=(10, 6))
ax.imshow(image_label_overlay)

plt.show()
```
~~~

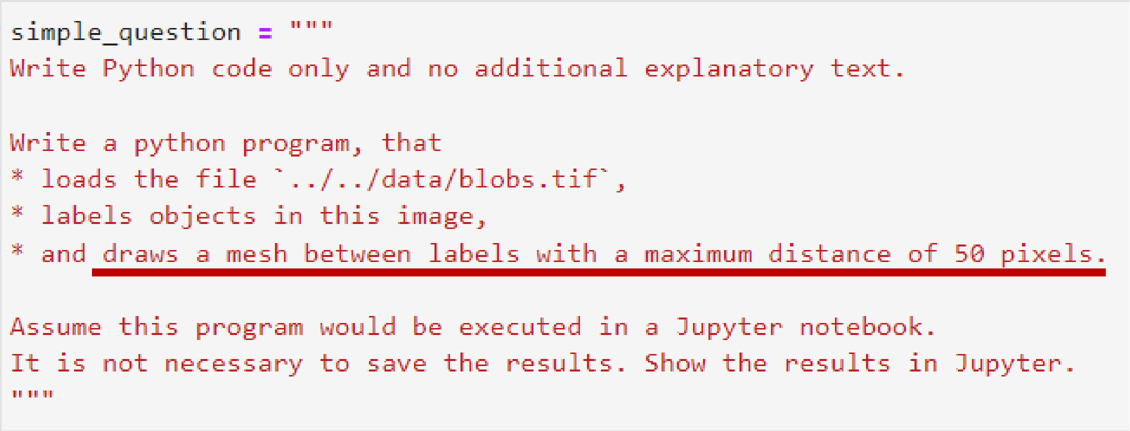
Result



https://haesleinhuepf.github.io/BioImageAnalysisNotebooks/07_prompt_engineering/04_generating_code_for_processing_images.html

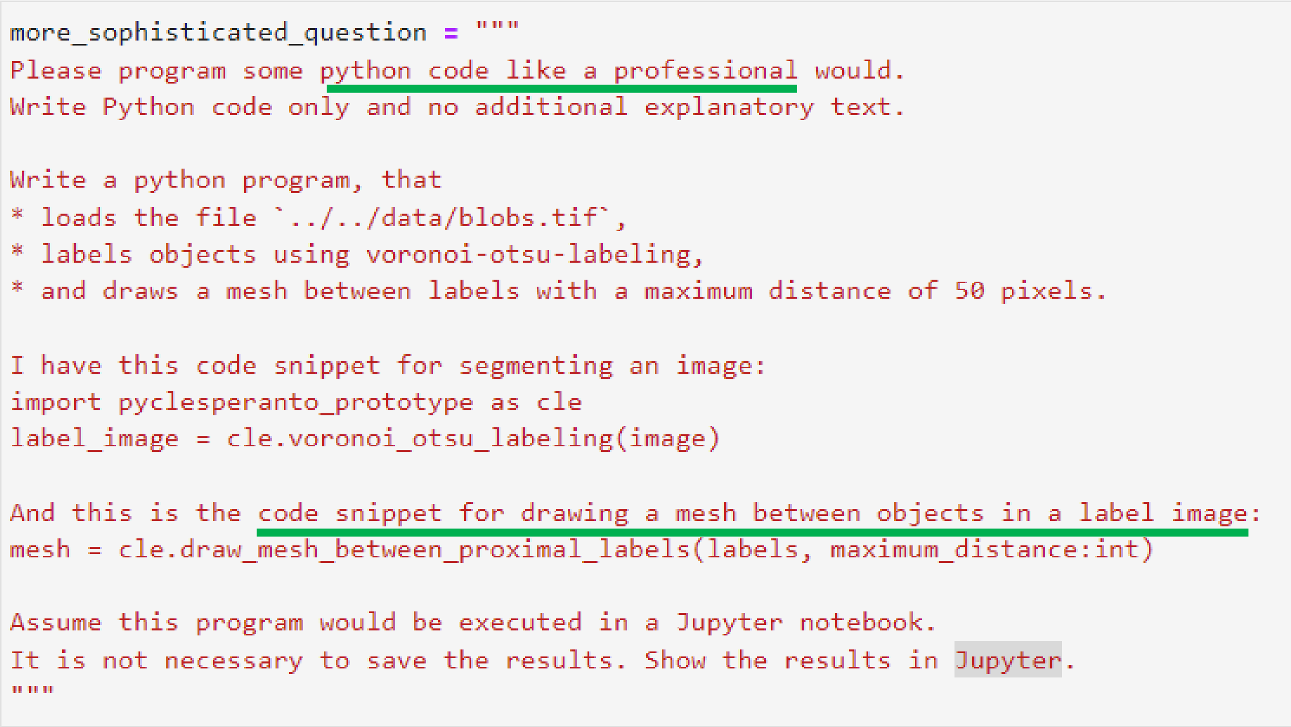
Prompt engineering

With more advanced tasks, it requires hints



```
simple_question = """
Write Python code only and no additional explanatory text.

Write a python program, that
* loads the file `../../data/blobs.tif`,
* labels objects in this image,
* and draws a mesh between labels with a maximum distance of 50 pixels.

Assume this program would be executed in a Jupyter notebook.
It is not necessary to save the results. Show the results in Jupyter.
"""
```

ChatGPT does not know how to do this and hallucinates code that does not work



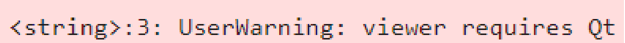
```
more_sophisticated_question = """
Please program some python code like a professional would.
Write Python code only and no additional explanatory text.

Write a python program, that
* loads the file `../../data/blobs.tif`,
* labels objects using voronoi-otsu-labeling,
* and draws a mesh between labels with a maximum distance of 50 pixels.

I have this code snippet for segmenting an image:
import pyclesperanto_prototype as cle
label_image = cle.voronoi_otsu_labeling(image)

And this is the code snippet for drawing a mesh between objects in a label image:
mesh = cle.draw_mesh_between_proximal_labels(labels, maximum_distance:int)

Assume this program would be executed in a Jupyter notebook.
It is not necessary to save the results. Show the results in Jupyter.
"""
```

https://haesleinhuepf.github.io/BioImageAnalysisNotebooks/07_prompt_engineering/05_generating_advanced_image_processing_code.html

Prompt engineering

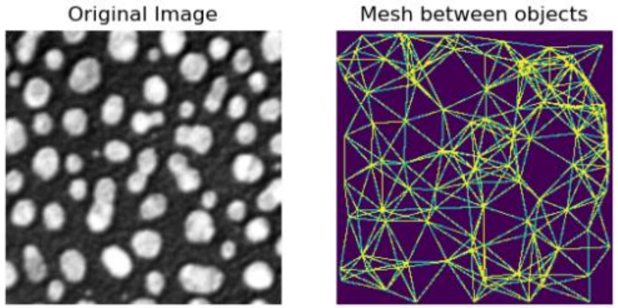
With more advanced tasks, it requires hints

Attempt 1

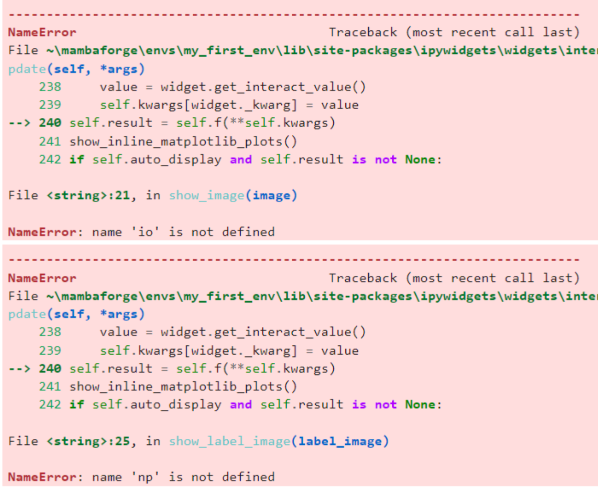


```
<string>:3: UserWarning: viewer requires Qt
```

Attempt 2



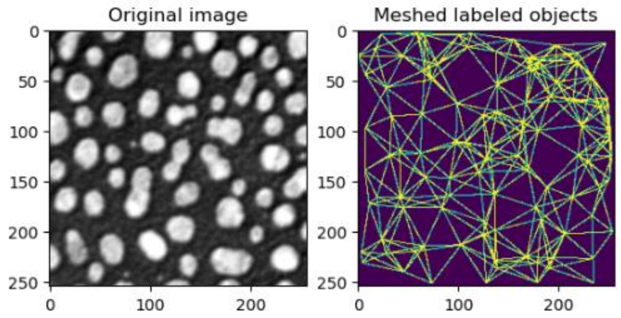
Attempt 3



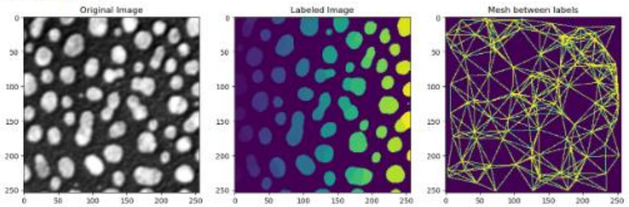
```
---------------------------------------------------------------------------
NameError                                 Traceback (most recent call last)
File ~\mambaforge\envs\my_first_env\lib\site-packages\ipywidgets\widgets\inte
pdate(self, *args)
    238     value = widget.get_interact_value()
    239     self.kwargs[widget._kwarg] = value
--> 240 self.result = self.f(**self.kwargs)
    241 show_inline_matplotlib_plots()
    242 if self.auto_display and self.result is not None:

File <string>:21, in show_image(image)

NameError: name 'io' is not defined
```

```
---------------------------------------------------------------------------
NameError                                 Traceback (most recent call last)
File ~\mambaforge\envs\my_first_env\lib\site-packages\ipywidgets\widgets\inte
pdate(self, *args)
    238     value = widget.get_interact_value()
    239     self.kwargs[widget._kwarg] = value
--> 240 self.result = self.f(**self.kwargs)
    241 show_inline_matplotlib_plots()
    242 if self.auto_display and self.result is not None:

File <string>:25, in show_label_image(label_image)

NameError: name 'np' is not defined
```

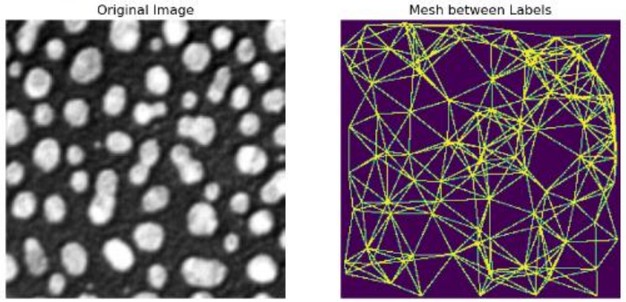
Attempt 6



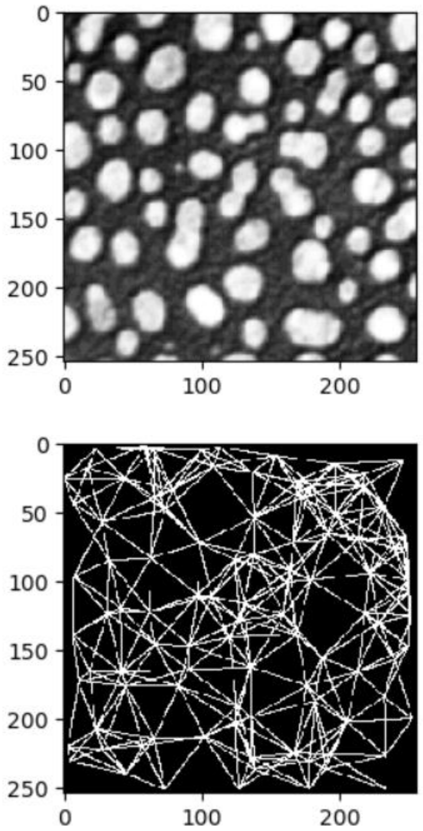
Attempt 9



Attempt 10



Attempt 8



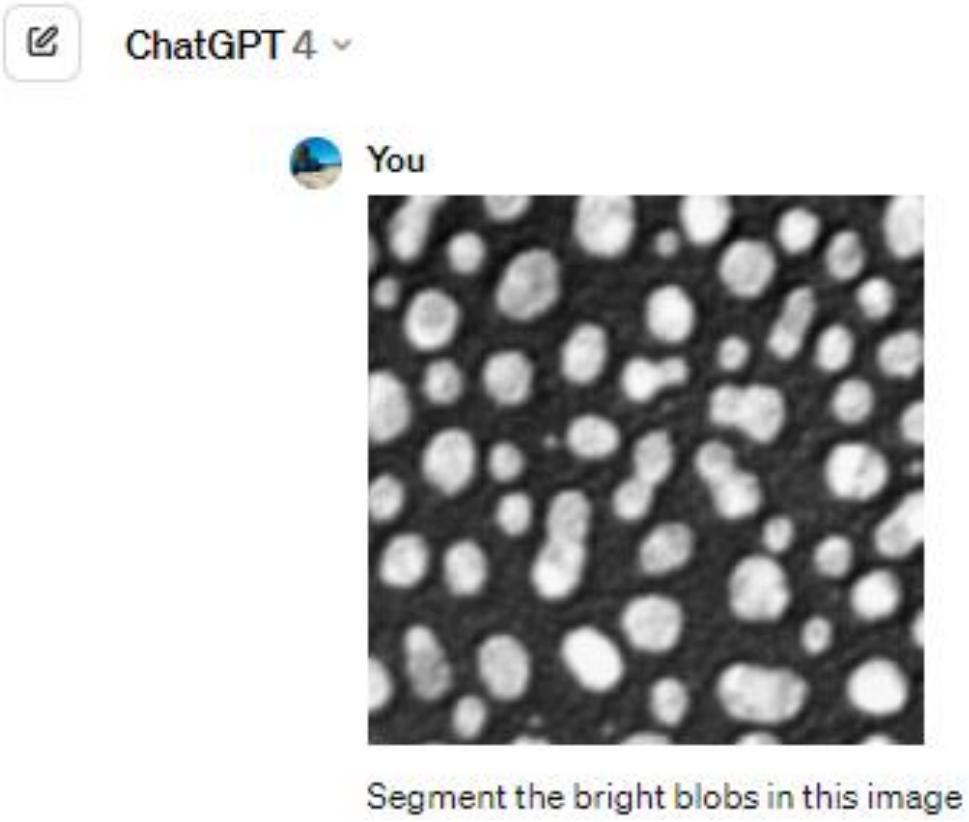
The *more sophisticated* prompt had errors in 4 out of 10 runs.

The *more sophisticated* prompt produced useful results in 5 out of 10 runs.

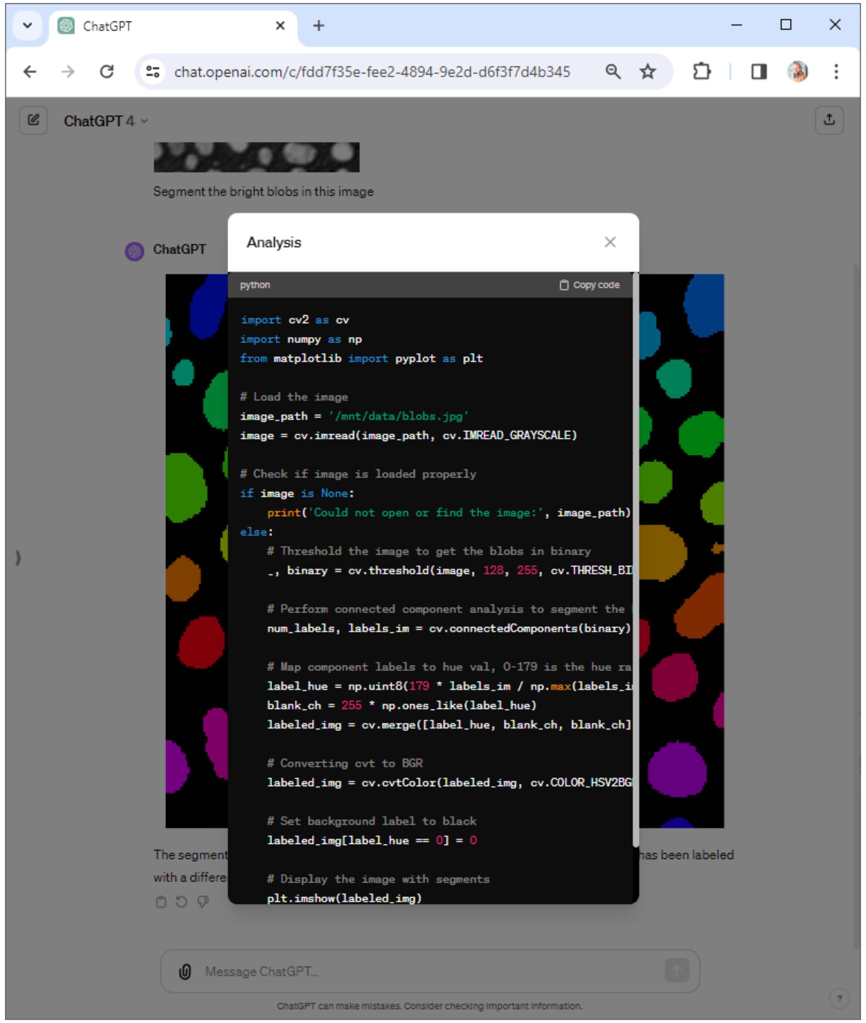
https://haesleinhuepf.github.io/BioImageAnalysisNotebooks/07_prompt_engineering/05_generating_advanced_image_processing_code.html

ChatGPT for analyzing scientific image data

- ChatGPT can perform [yet simple] analysis tasks,
- and tell you what it did.



Segment the bright blobs in this image


ChatGPT
chat.openai.com/c/fdd7f35e-fee2-4894-9e2d-d6f3f7d4b345
ChatGPT 4
ChatGPT
Analysis
python
Copy code
Message ChatGPT...
ChatGPT can make mistakes. Consider checking important information.

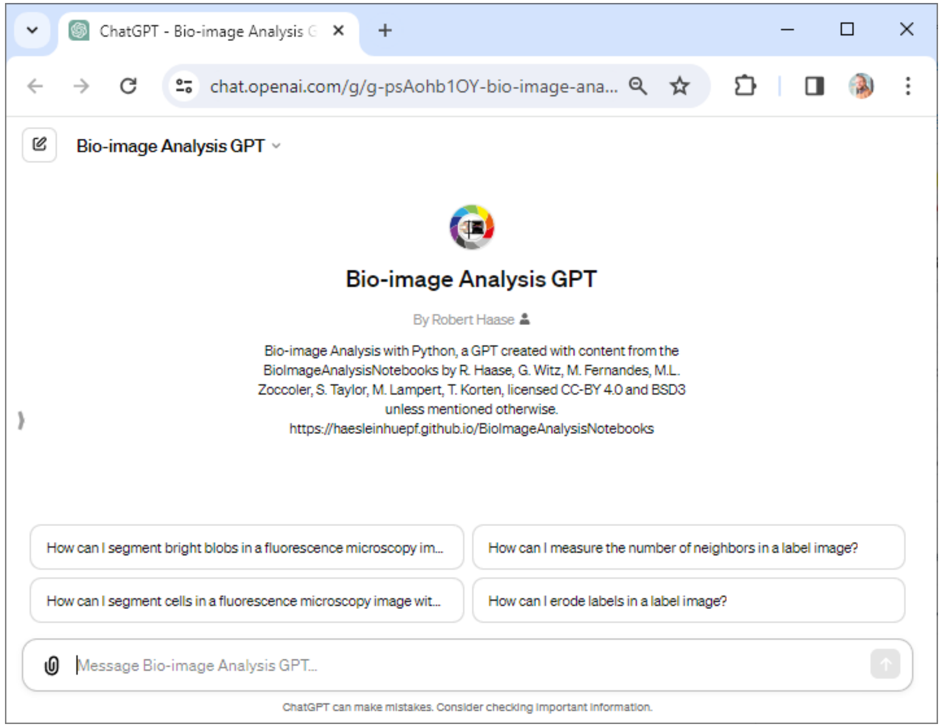
Segment the bright blobs in this image

```python
import cv2 as cv
import numpy as np
from matplotlib import pyplot as plt

# Load the image
image_path = '/mnt/data/blobs.jpg'
image = cv.imread(image_path, cv.IMREAD_GRAYSCALE)

# Check if image is loaded properly
if image is None:
    print('Could not open or find the image:', image_path)
else:
    # Threshold the image to get the blobs in binary
    _, binary = cv.threshold(image, 128, 255, cv.THRESH_BI

    # Perform connected component analysis to segment the
    num_labels, labels_im = cv.connectedComponents(binary)

    # Map component labels to hue val, 0-179 is the hue ra
    label_hue = np.uint8(179 * labels_im / np.max(labels_i
    blank_ch = 255 * np.ones_like(label_hue)
    labeled_img = cv.merge([label_hue, blank_ch, blank_ch]

    # Converting cvt to BGR
    labeled_img = cv.cvtColor(labeled_img, cv.COLOR_HSV2BG

    # Set background label to black
    labeled_img[label_hue == 0] = 0

    # Display the image with segments
    plt.imshow(labeled_img)
```

https://chat.openai.com/g/g-HMNcP6w7d-data-analyst

How about a Bio-image Analysis GPT

OpenAI allows customization of ChatGPT


ChatGPT - Bio-image Analysis G
chat.openai.com/g/g-psAohb1OY-bio-image-ana...
Bio-image Analysis GPT

Bio-image Analysis GPT

By Robert Haase

Bio-image Analysis with Python, a GPT created with content from the BioImageAnalysisNotebooks by R. Haase, G. Witz, M. Fernandes, M.L. Zoccoler, S. Taylor, M. Lampert, T. Korten, licensed CC-BY 4.0 and BSD3 unless mentioned otherwise.
https://haesleinhuepf.github.io/BioImageAnalysisNotebooks

How can I segment bright blobs in a fluorescence microscopy im...

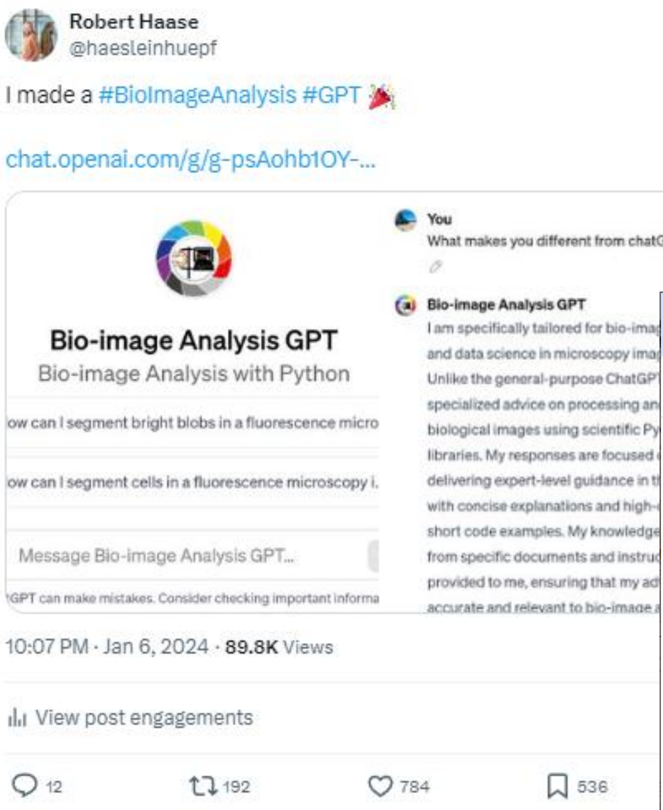
How can I measure the number of neighbors in a label image?

How can I segment cells in a fluorescence microscopy image wit...

How can I erode labels in a label image?

Message Bio-image Analysis GPT...

ChatGPT can make mistakes. Consider checking important information.



Robert Haase
@haesleinhuepf

I made a #BioImageAnalysis #GPT 🎉

chat.openai.com/g/g-psAohb1OY-...

Bio-image Analysis GPT
Bio-image Analysis with Python

ow can I segment bright blobs in a fluorescence micro

ow can I segment cells in a fluorescence microscopy i.

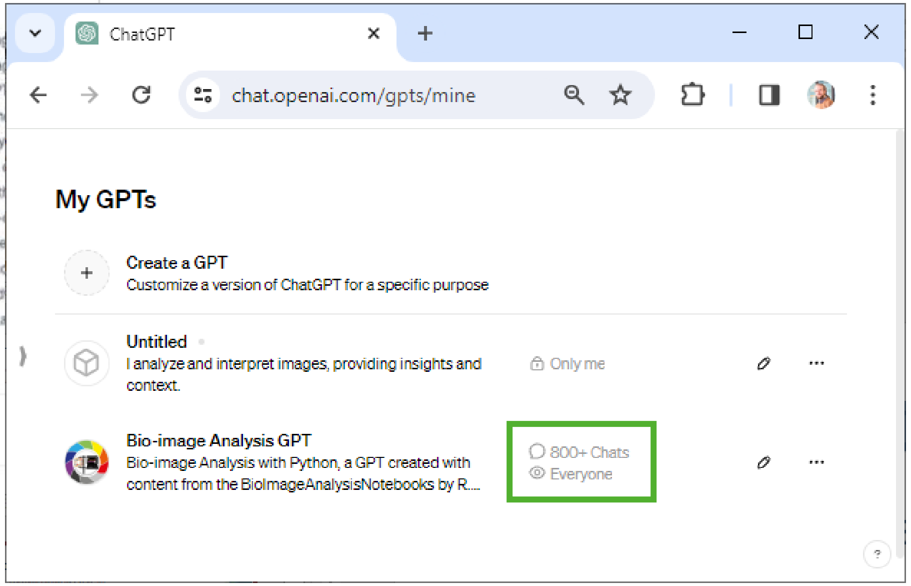
Message Bio-image Analysis GPT...

GPT can make mistakes. Consider checking important informa

You
What makes you different from chatGPT?

Bio-image Analysis GPT
I am specifically tailored for bio-imag
and data science in microscopy imag
Unlike the general-purpose ChatGP
specialized advice on processing an
biological images using scientific Py
libraries. My responses are focused
delivering expert-level guidance in t
with concise explanations and high-
short code examples. My knowledge
from specific documents and instruc
provided to me, ensuring that my ad
accurate and relevant to bio-image a

10:07 PM · Jan 6, 2024 · **89.8K** Views



My GPTs

Create a GPT
Customize a version of ChatGPT for a specific purpose

Untitled
I analyze and interpret images, providing insights and context.
Only me

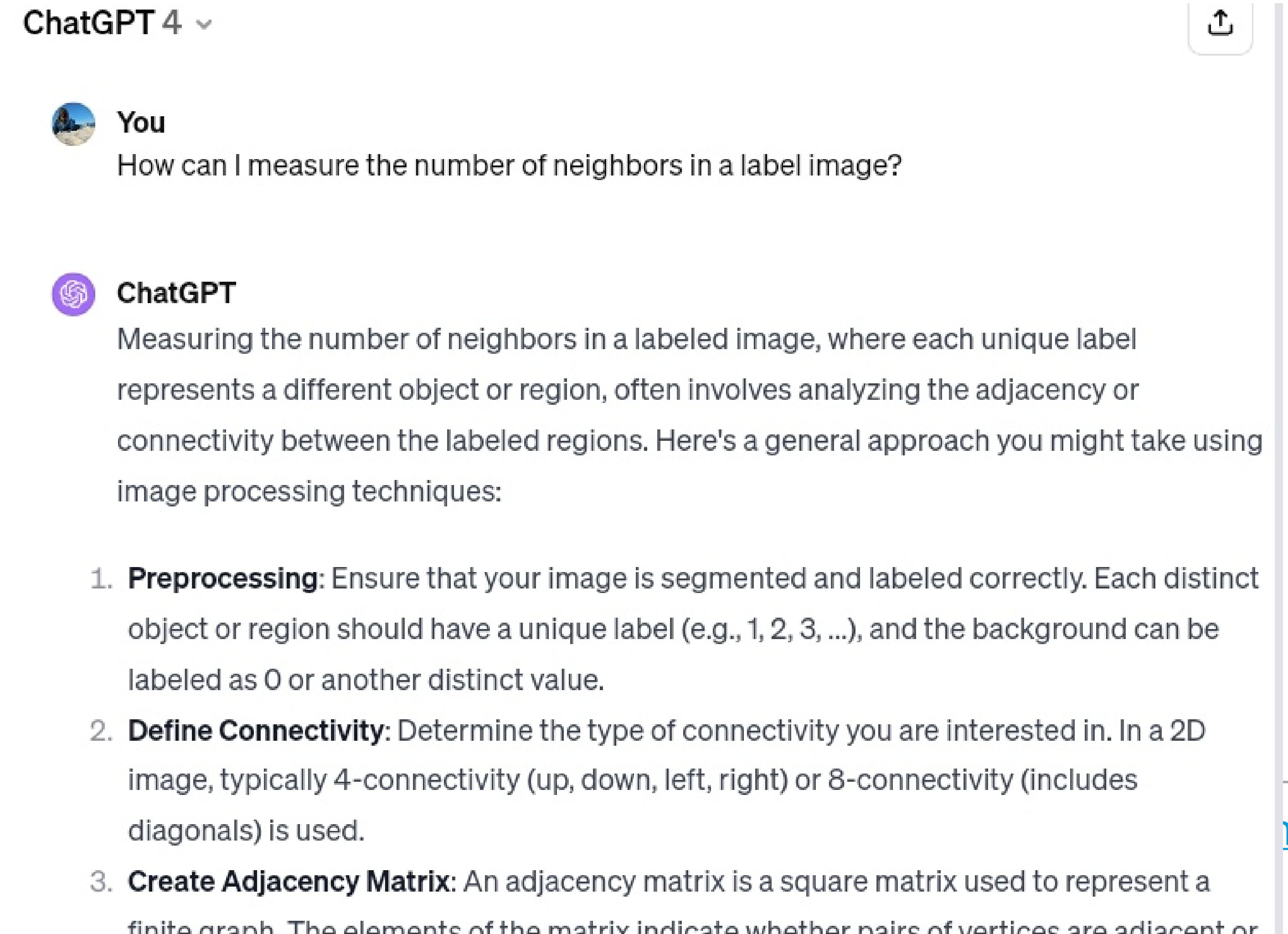
Bio-image Analysis GPT
Bio-image Analysis with Python, a GPT created with content from the BioImageAnalysisNotebooks by R...
800+ Chats
Everyone

https://chat.openai.com/g/g-psAohb1OY- bio-image-analysis-gpt

https://twitter.com/haesleinhuepf/status/1743741036701077849

ChatGPT for analyzing scientific image data

ChatGPT in comparison with the Bio-image Analysis GPT



You

How can I measure the number of neighbors in a label image?

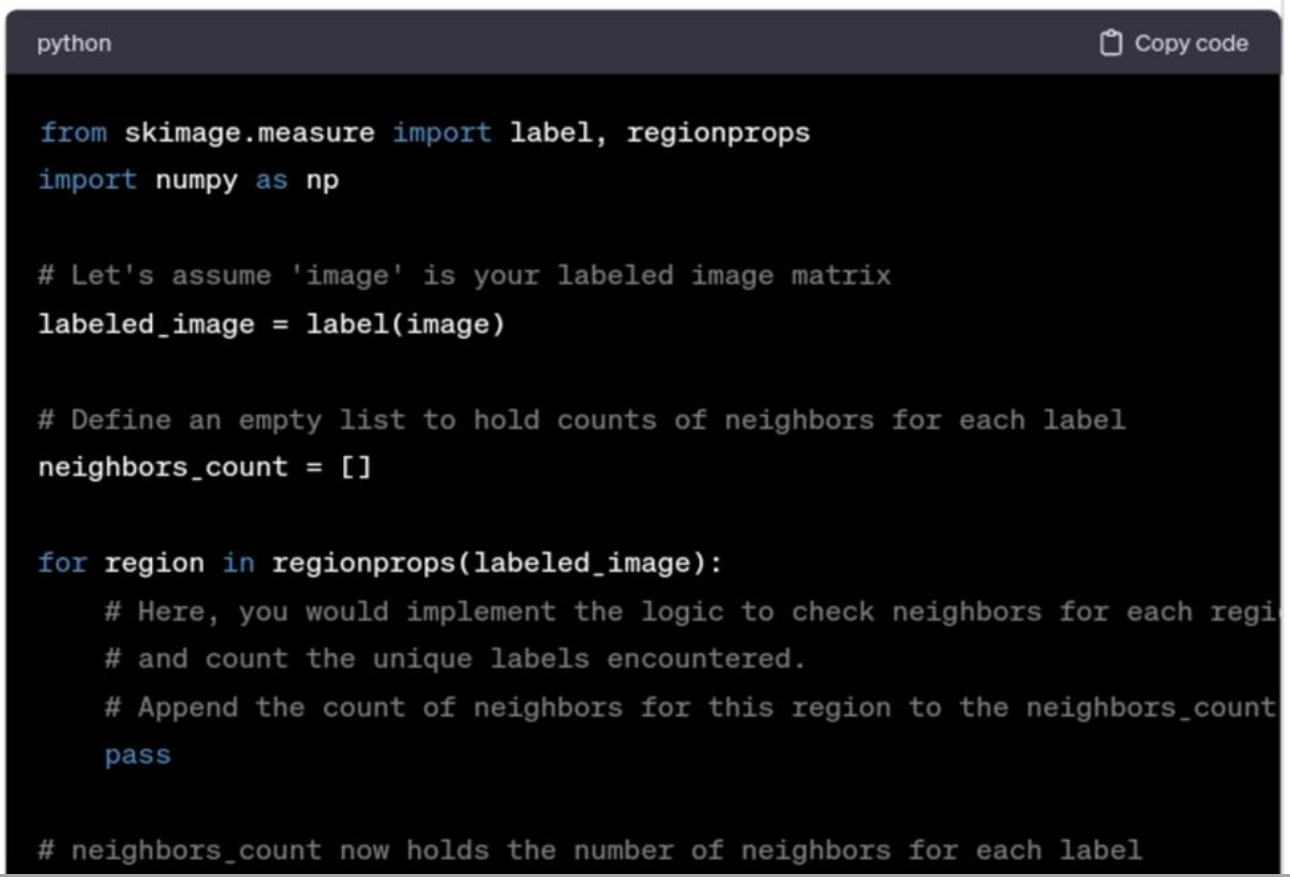
ChatGPT

Measuring the number of neighbors in a labeled image, where each unique label represents a different object or region, often involves analyzing the adjacency or connectivity between the labeled regions. Here's a general approach you might take using image processing techniques:

1. **Preprocessing**: Ensure that your image is segmented and labeled correctly. Each distinct object or region should have a unique label (e.g., 1, 2, 3, ...), and the background can be labeled as 0 or another distinct value.
2. **Define Connectivity**: Determine the type of connectivity you are interested in. In a 2D image, typically 4-connectivity (up, down, left, right) or 8-connectivity (includes diagonals) is used.
3. **Create Adjacency Matrix**: An adjacency matrix is a square matrix used to represent a finite graph. The elements of the matrix indicate whether pairs of vertices are adjacent or

ChatGPT for analyzing scientific image data

ChatGPT in comparison with the Bio-image Analysis GPT



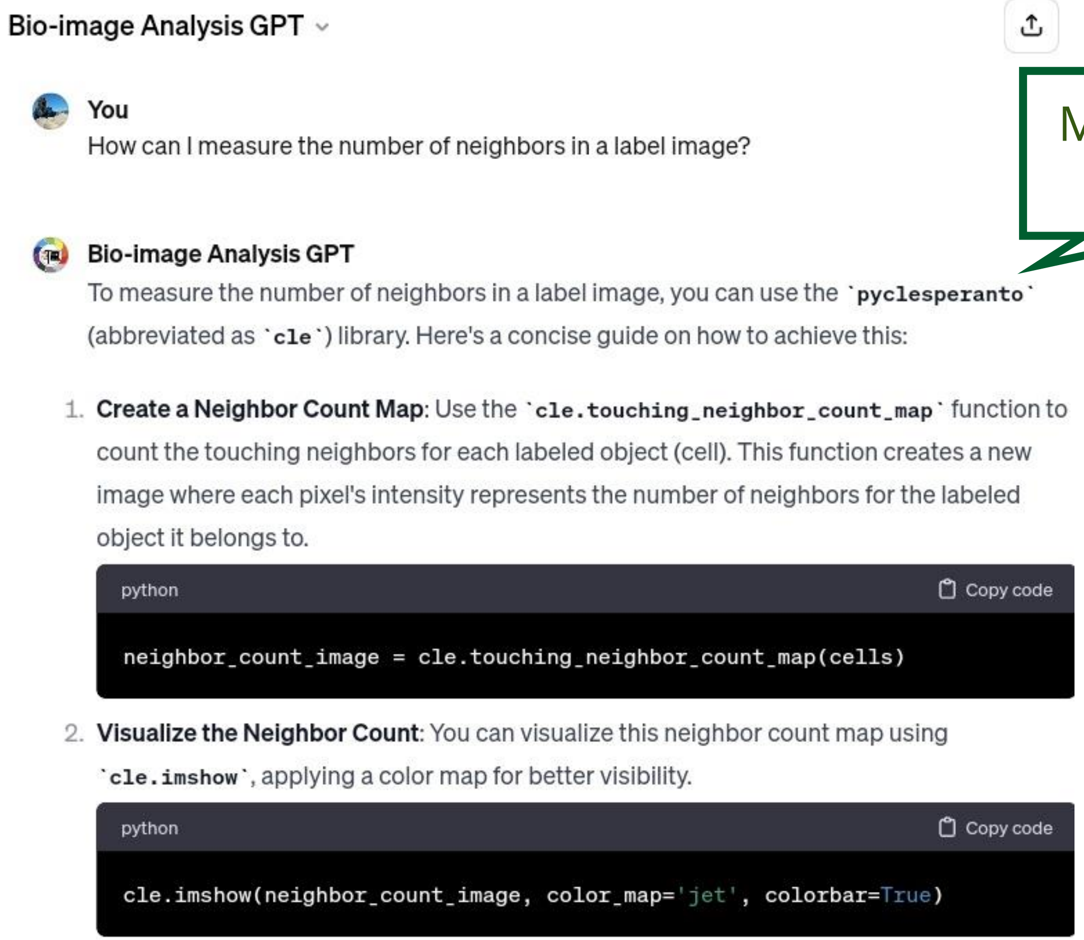
```python
from skimage.measure import label, regionprops
import numpy as np

# Let's assume 'image' is your labeled image matrix
labeled_image = label(image)

# Define an empty list to hold counts of neighbors for each label
neighbors_count = []

for region in regionprops(labeled_image):
    # Here, you would implement the logic to check neighbors for each regi
    # and count the unique labels encountered.
    # Append the count of neighbors for this region to the neighbors_count
    pass

# neighbors_count now holds the number of neighbors for each label
```

Limited Bio-image Analysis domain-specific knowledge

https://chat.openai.com/g/g-psAohb1OY-bio-image-analysis-gpt

ChatGPT for analyzing scientific image data

ChatGPT in comparison with the Bio-image Analysis GPT



Bio-image Analysis GPT

You

How can I measure the number of neighbors in a label image?

Bio-image Analysis GPT

To measure the number of neighbors in a label image, you can use the `pyclesperanto` (abbreviated as `cle`) library. Here's a concise guide on how to achieve this:

1. **Create a Neighbor Count Map**: Use the `cle.touching_neighbor_count_map` function to count the touching neighbors for each labeled object (cell). This function creates a new image where each pixel's intensity represents the number of neighbors for the labeled object it belongs to.

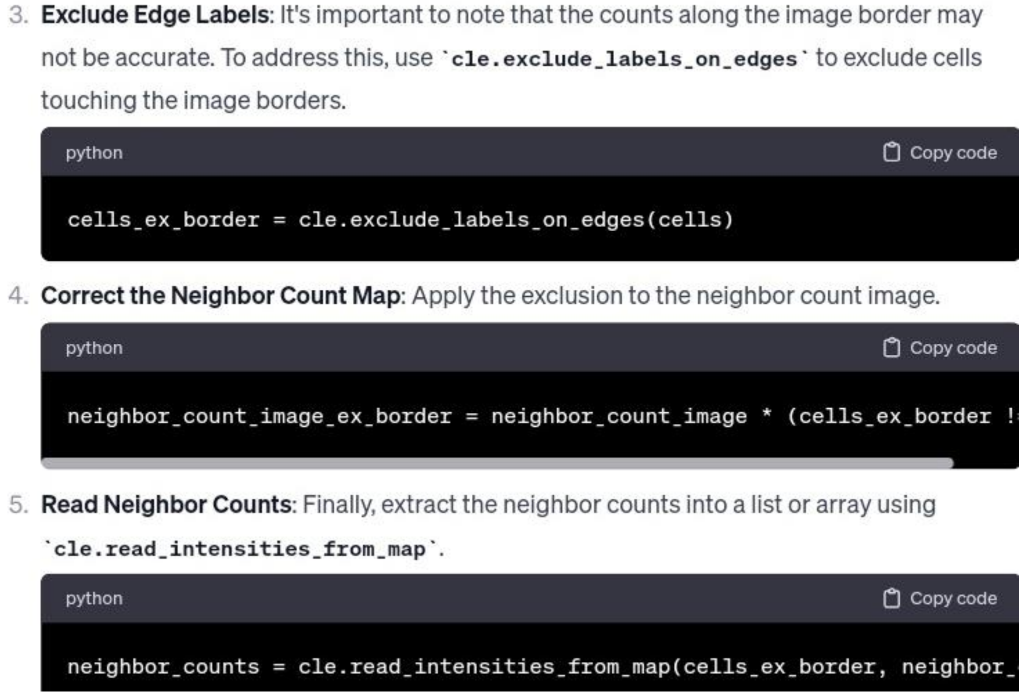
```python
neighbor_count_image = cle.touching_neighbor_count_map(cells)
```

2. **Visualize the Neighbor Count**: You can visualize this neighbor count map using `cle.imshow`, applying a color map for better visibility.

```python
cle.imshow(neighbor_count_image, color_map='jet', colorbar=True)
```

More concise responses

Containing some Bio-image Analysis knowledge



3. **Exclude Edge Labels**: It's important to note that the counts along the image border may not be accurate. To address this, use `cle.exclude_labels_on_edges` to exclude cells touching the image borders.

```python
cells_ex_border = cle.exclude_labels_on_edges(cells)
```

4. **Correct the Neighbor Count Map**: Apply the exclusion to the neighbor count image.

```python
neighbor_count_image_ex_border = neighbor_count_image * (cells_ex_border !
```

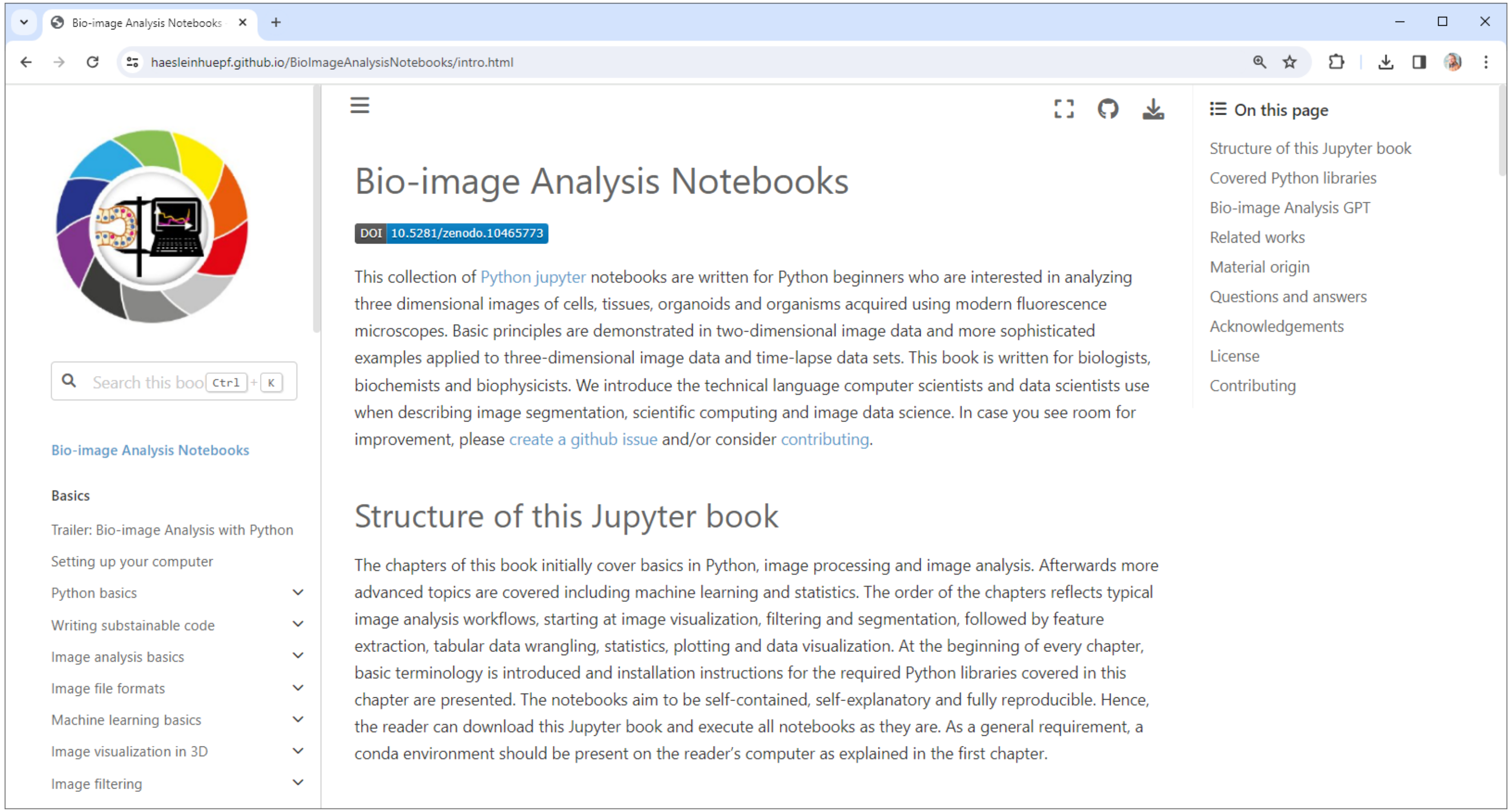
5. **Read Neighbor Counts**: Finally, extract the neighbor counts into a list or array using `cle.read_intensities_from_map`.

```python
neighbor_counts = cle.read_intensities_from_map(cells_ex_border, neighbor_
```

https://chat.openai.com/g/g-psAohb1OY-bio-image-analysis-gpt

ChatGPT for analyzing scientific image data

Bio-image Analysis GPT is extensible, it’s knowledge base is made from:


Bio-image Analysis Notebooks
haesleinhuepf.github.io/BioImageAnalysisNotebooks/intro.html
Search this boo Ctrl + K

Bio-image Analysis Notebooks

Basics

- Trailer: Bio-image Analysis with Python
- Setting up your computer
- Python basics
- Writing substainable code
- Image analysis basics
- Image file formats
- Machine learning basics
- Image visualization in 3D
- Image filtering

Bio-image Analysis Notebooks

DOI 10.5281/zenodo.10465773

This collection of Python jupyter notebooks are written for Python beginners who are interested in analyzing three dimensional images of cells, tissues, organoids and organisms acquired using modern fluorescence microscopes. Basic principles are demonstrated in two-dimensional image data and more sophisticated examples applied to three-dimensional image data and time-lapse data sets. This book is written for biologists, biochemists and biophysicists. We introduce the technical language computer scientists and data scientists use when describing image segmentation, scientific computing and image data science. In case you see room for improvement, please create a github issue and/or consider contributing.

Structure of this Jupyter book

The chapters of this book initially cover basics in Python, image processing and image analysis. Afterwards more advanced topics are covered including machine learning and statistics. The order of the chapters reflects typical image analysis workflows, starting at image visualization, filtering and segmentation, followed by feature extraction, tabular data wrangling, statistics, plotting and data visualization. At the beginning of every chapter, basic terminology is introduced and installation instructions for the required Python libraries covered in this chapter are presented. The notebooks aim to be self-contained, self-explanatory and fully reproducible. Hence, the reader can download this Jupyter book and execute all notebooks as they are. As a general requirement, a conda environment should be present on the reader’s computer as explained in the first chapter.

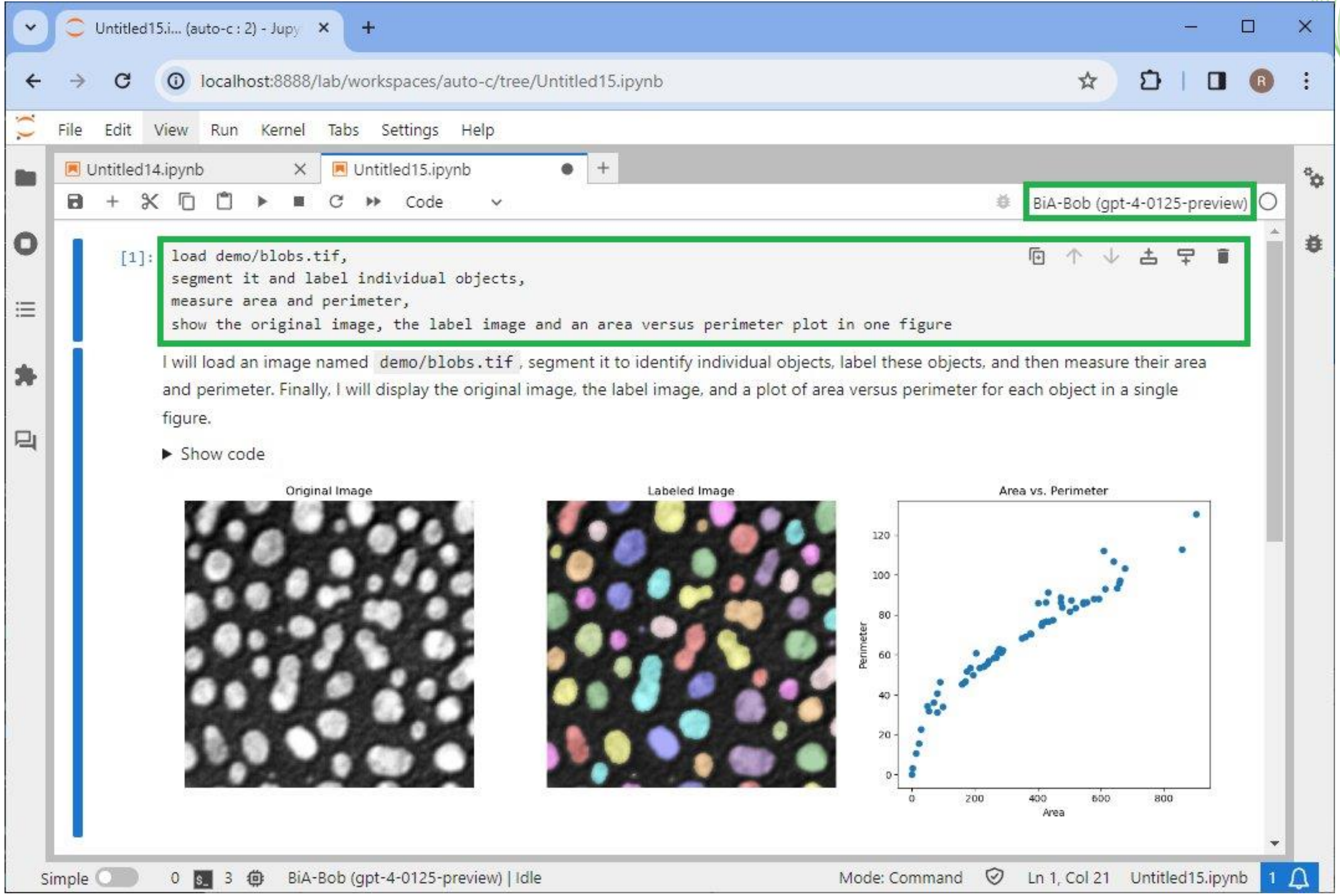
On this page
Structure of this Jupyter book
Covered Python libraries
Bio-image Analysis GPT
Related works
Material origin
Questions and answers
Acknowledgements
License
Contributing

Just submit a pull-request to the Bio-image Analysis Notebooks

https://haesleinhuepf.github.io/BioImageAnalysisNotebooks
https://chat.openai.com/g/g-psAohb1OY-bio-image-analysis-gpt

How about doing this locally? BiA-Bob

- Open-source alternative to OpenAI's Data Analyst
- No need to upload your [image] data
- Extensible through plugins


Untitled15.i... (auto-c : 2) - Jupy
localhost:8888/lab/workspaces/auto-c/tree/Untitled15.ipynb
File
Edit
View
Run
Kernel
Tabs
Settings
Help
Untitled14.ipynb
Untitled15.ipynb
Code
BiA-Bob (gpt-4-0125-preview)
[1]:

```
load demo/blobs.tif,
segment it and label individual objects,
measure area and perimeter,
show the original image, the label image and an area versus perimeter plot in one figure
```

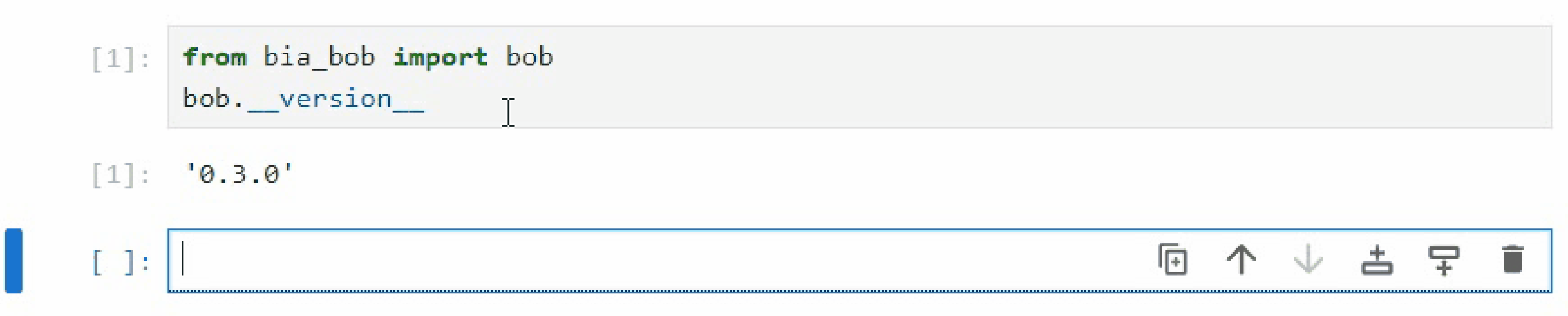
I will load an image named `demo/blobs.tif`, segment it to identify individual objects, label these objects, and then measure their area and perimeter. Finally, I will display the original image, the label image, and a plot of area versus perimeter for each object in a single figure.

With Christian Tischer (EMBL Heidelberg) @tischitischer

https://github.com/haesleinhuepf/bia-bob

How about doing this locally? BiA-Bob

- Open-source alternative to OpenAI's Data Analyst
- No need to upload your [image] data
- Extensible through plugins



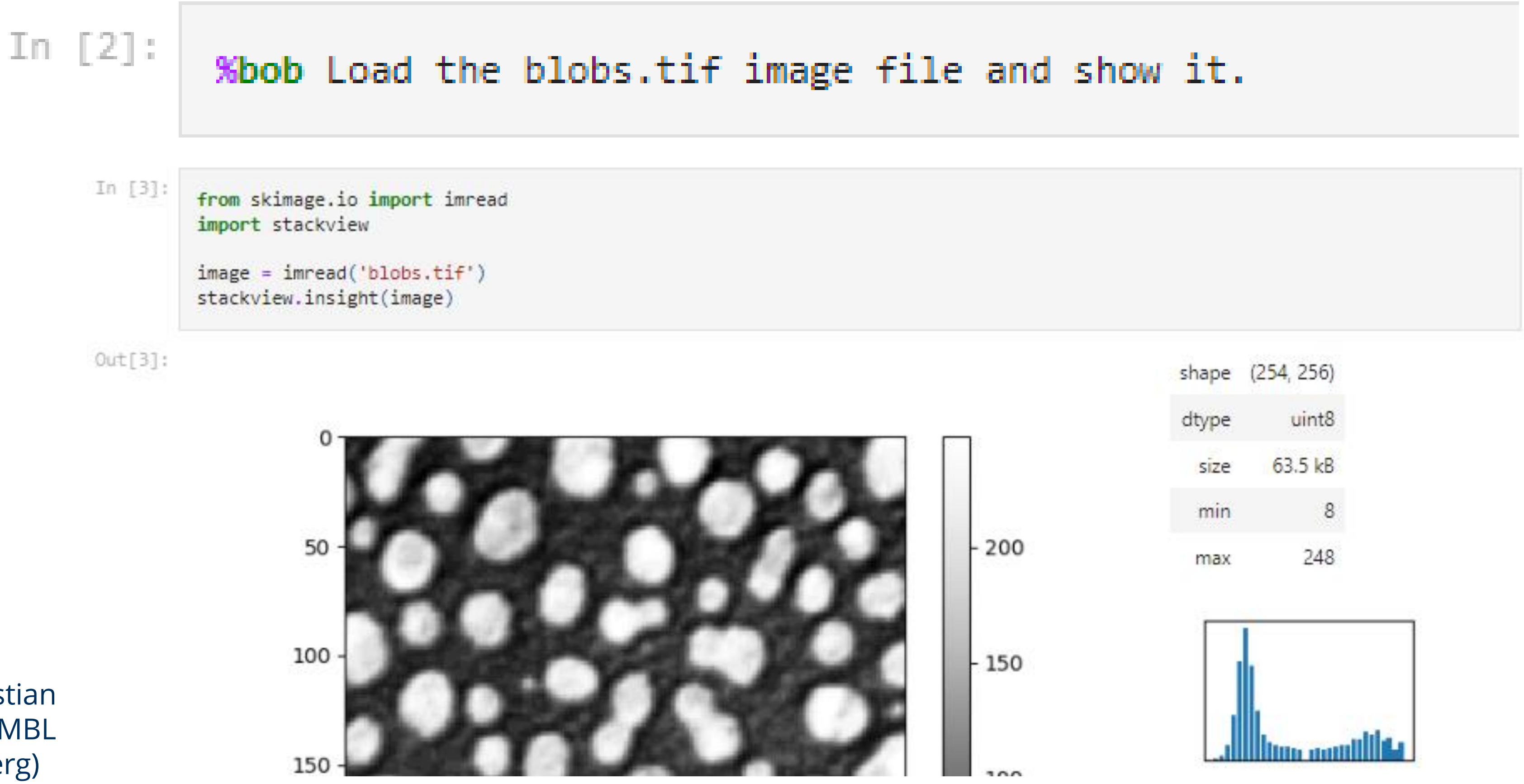
```
[1]: from bia_bob import bob
     bob.__version__

[1]: '0.3.0'

[ ]: 
```

With Christian Tischer (EMBL Heidelberg) @tischitischer

https://github.com/haesleinhuepf/bia-bob

Analyzing images using LLMs / BiA-Bob

Use the `%bob` … syntax to ask for small things.



```
In [2]:  %bob Load the blobs.tif image file and show it.
```

```
In [3]:  from skimage.io import imread
         import stackview

         image = imread('blobs.tif')
         stackview.insight(image)

Out[3]:
```

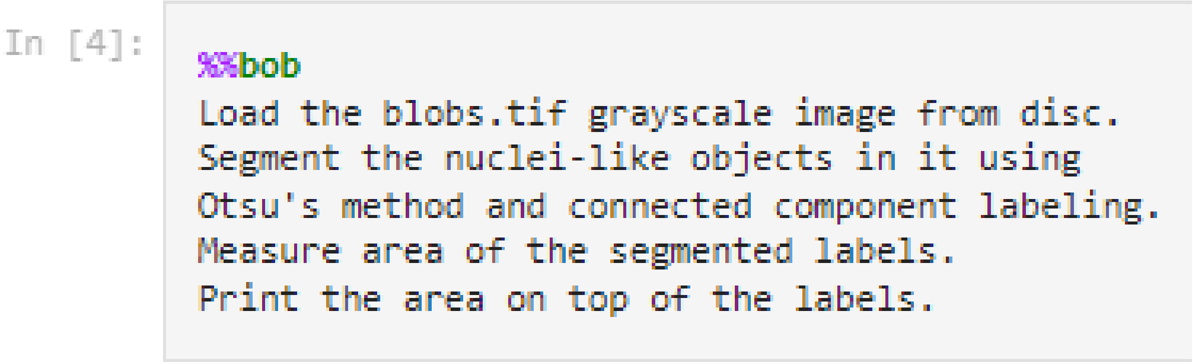
shape	(254, 256)
dtype	uint8
size	63.5 kB
min	8
max	248

With Christian Tischer (EMBL Heidelberg)
@tischitischer

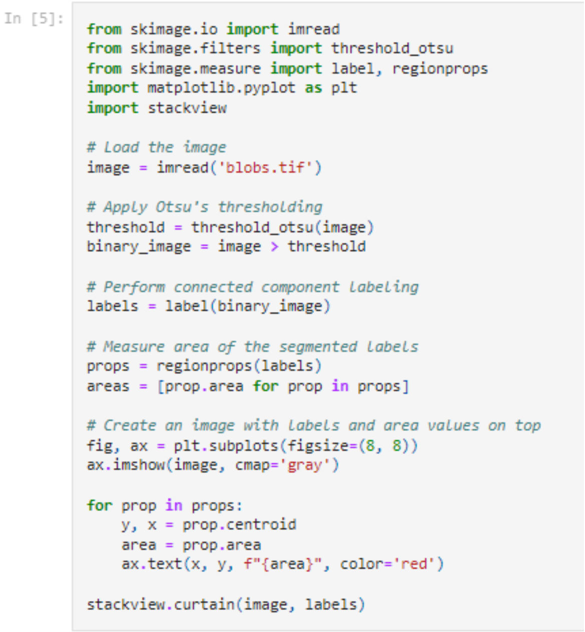
https://github.com/haesleinhuepf/bia-bob/blob/main/demo/basic_demo.ipynb

Analyzing images using LLMs / BiA-Bob

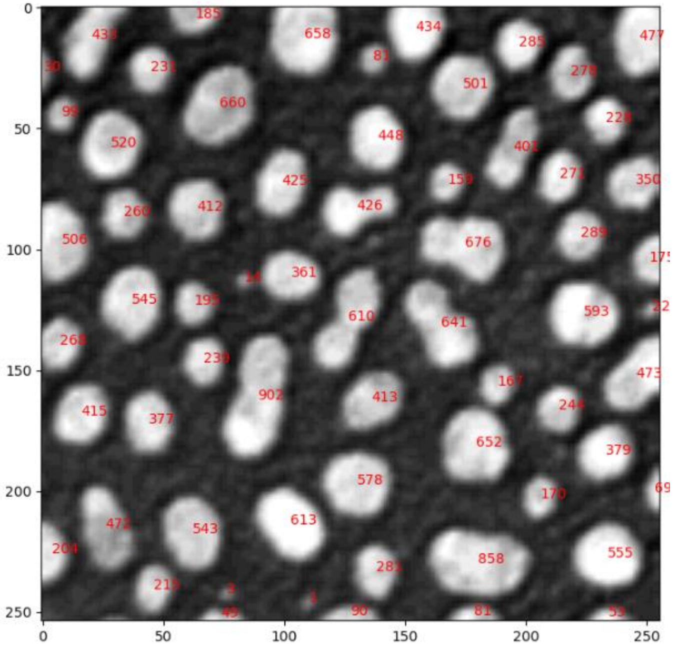
Use the `%%bob ...` syntax to ask for complex analysis tasks.



```
In [4]:  %%bob
         Load the blobs.tif grayscale image from disc.
         Segment the nuclei-like objects in it using
         Otsu's method and connected component labeling.
         Measure area of the segmented labels.
         Print the area on top of the labels.
```



```
In [5]:
from skimage.io import imread
from skimage.filters import threshold_otsu
from skimage.measure import label, regionprops
import matplotlib.pyplot as plt
import stackview

# Load the image
image = imread('blobs.tif')

# Apply Otsu's thresholding
threshold = threshold_otsu(image)
binary_image = image > threshold

# Perform connected component labeling
labels = label(binary_image)

# Measure area of the segmented labels
props = regionprops(labels)
areas = [prop.area for prop in props]

# Create an image with labels and area values on top
fig, ax = plt.subplots(figsize=(8, 8))
ax.imshow(image, cmap='gray')

for prop in props:
    y, x = prop.centroid
    area = prop.area
    ax.text(x, y, f"{area}", color='red')

stackview.curtain(image, labels)
```

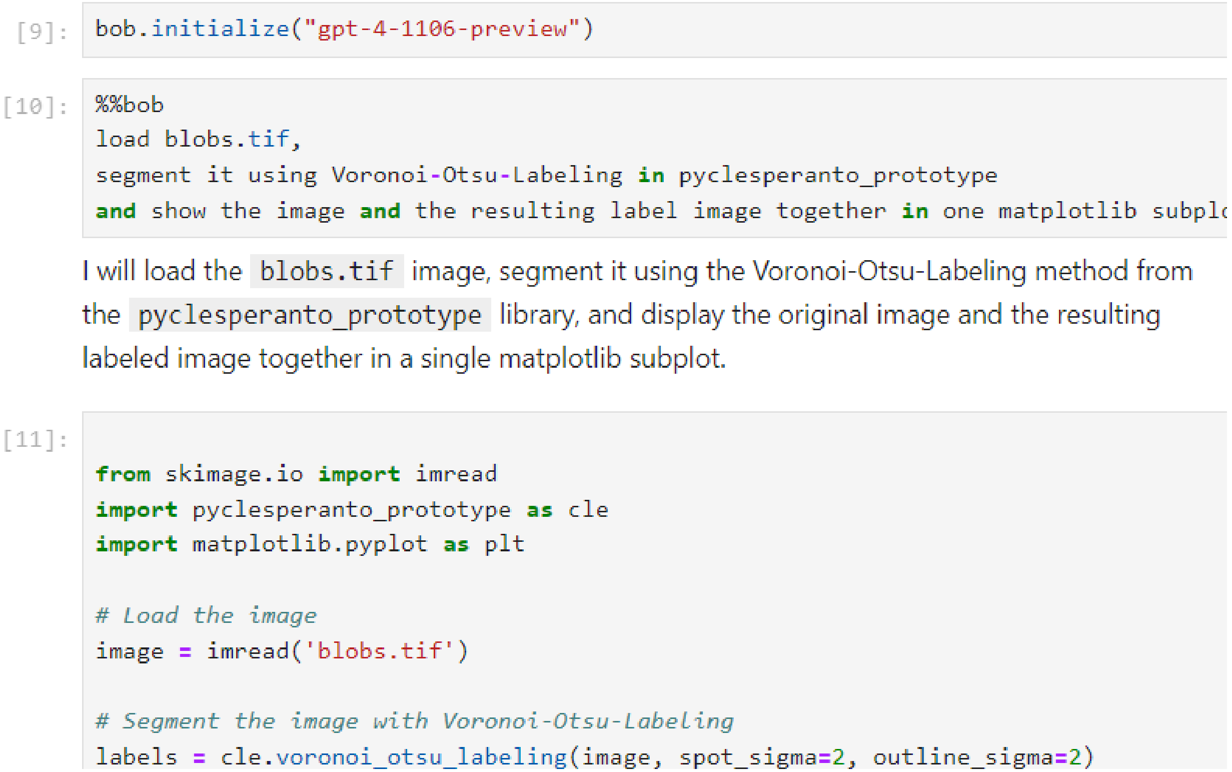


With Christian Tischer (EMBL Heidelberg) @tischitischer

https://github.com/haesleinhuepf/bia-bob/blob/main/demo/basic_demo.ipynb

BiA-Bob: Available language models / backends

- ChatGPT by OpenAI



[9]:

```
bob.initialize("gpt-4-1106-preview")
```

[10]:

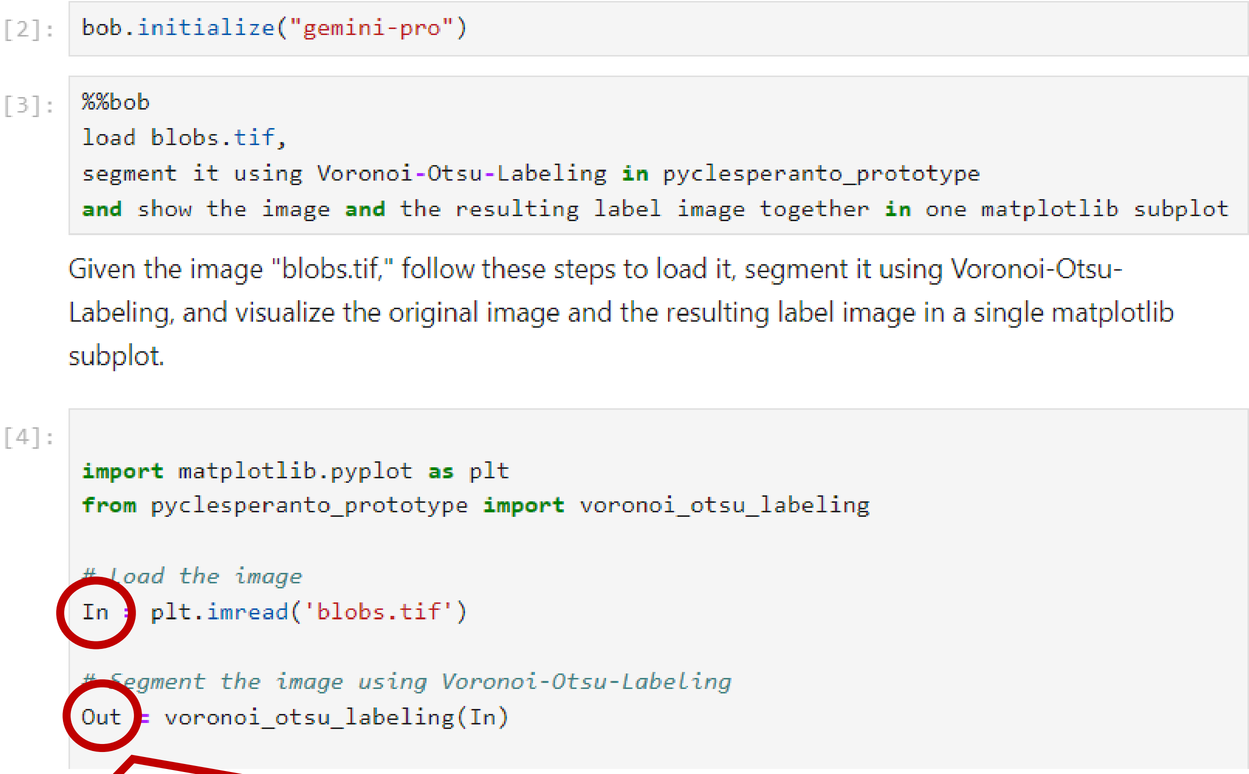
```
%%bob
load blobs.tif,
segment it using Voronoi-Otsu-Labeling in pyclesperanto_prototype
and show the image and the resulting label image together in one matplotlib subpl
```

I will load the `blobs.tif` image, segment it using the Voronoi-Otsu-Labeling method from the `pyclesperanto_prototype` library, and display the original image and the resulting labeled image together in a single matplotlib subplot.

[11]:

```
from skimage.io import imread
import pyclesperanto_prototype as cle
import matplotlib.pyplot as plt

# Load the image
image = imread('blobs.tif')

# Segment the image with Voronoi-Otsu-Labeling
labels = cle.voronoi_otsu_labeling(image, spot_sigma=2, outline_sigma=2)
```

- Gemini by Google



[2]:

```
bob.initialize("gemini-pro")
```

[3]:

```
%%bob
load blobs.tif,
segment it using Voronoi-Otsu-Labeling in pyclesperanto_prototype
and show the image and the resulting label image together in one matplotlib subplot
```

Given the image "blobs.tif," follow these steps to load it, segment it using Voronoi-Otsu-Labeling, and visualize the original image and the resulting label image in a single matplotlib subplot.

[4]:

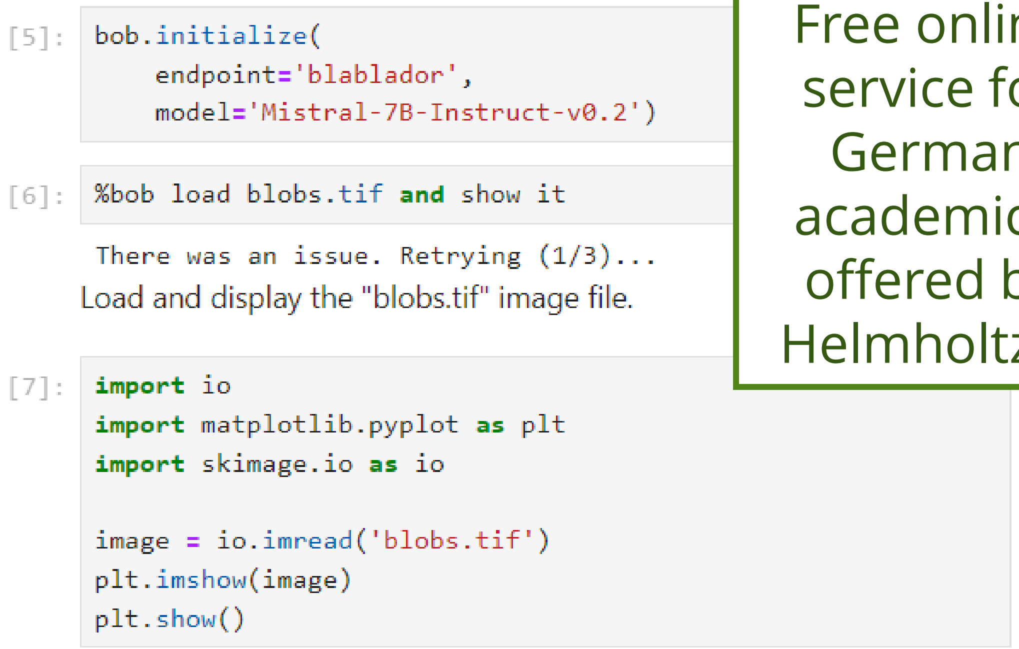
```
import matplotlib.pyplot as plt
from pyclesperanto_prototype import voronoi_otsu_labeling

# Load the image
In = plt.imread('blobs.tif')

# Segment the image using Voronoi-Otsu-Labeling
Out = voronoi_otsu_labeling(In)
```

Interesting variable names

https://github.com/haesleinhuepf/bia-bob/blob/main/demo/gemini.ipynb

BiA-Bob: Available language models / backends

- Mistral through Blablador



```
[5]: bob.initialize(
         endpoint='blablador',
         model='Mistral-7B-Instruct-v0.2')
```

```
[6]: %bob load blobs.tif and show it
```

```
There was an issue. Retrying (1/3)...
```

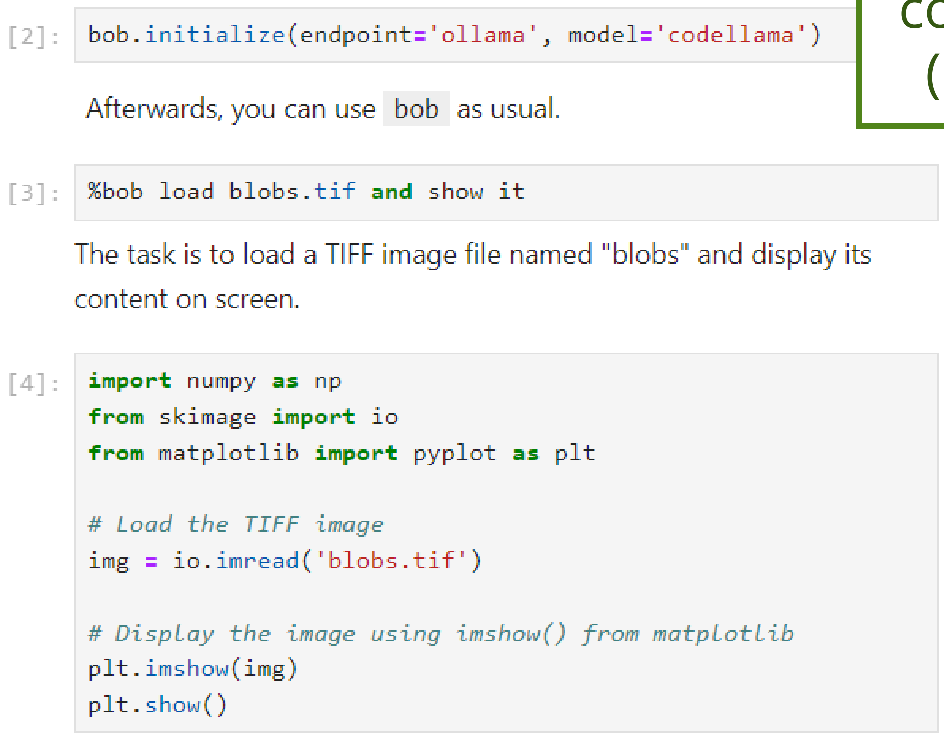
Load and display the "blobs.tif" image file.

```
[7]: import io
     import matplotlib.pyplot as plt
     import skimage.io as io

     image = io.imread('blobs.tif')
     plt.imshow(image)
     plt.show()
```

Free online service for German academics, offered by HelmholtzAI

- CodeLLama via ollama

Runs on your local computer (offline)



```
[2]: bob.initialize(endpoint='ollama', model='codellama')
```

Afterwards, you can use `bob` as usual.

```
[3]: %bob load blobs.tif and show it
```

The task is to load a TIFF image file named "blobs" and display its content on screen.

```
[4]: import numpy as np
     from skimage import io
     from matplotlib import pyplot as plt

     # Load the TIFF image
     img = io.imread('blobs.tif')

     # Display the image using imshow() from matplotlib
     plt.imshow(img)
     plt.show()
```

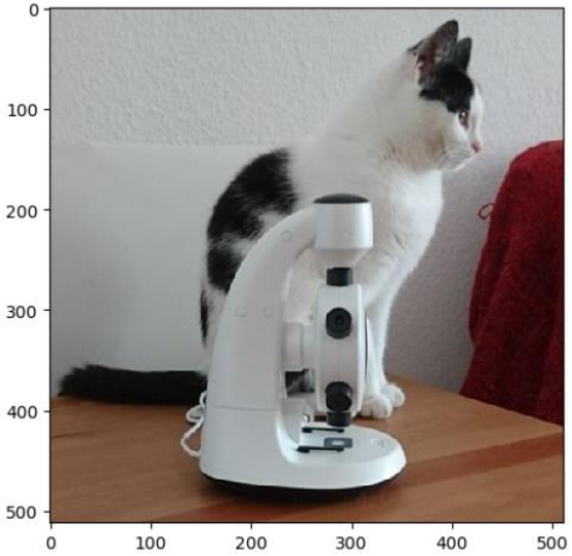
Many OpenSource models *seem* less capable than the commercial models, when it comes to complex code

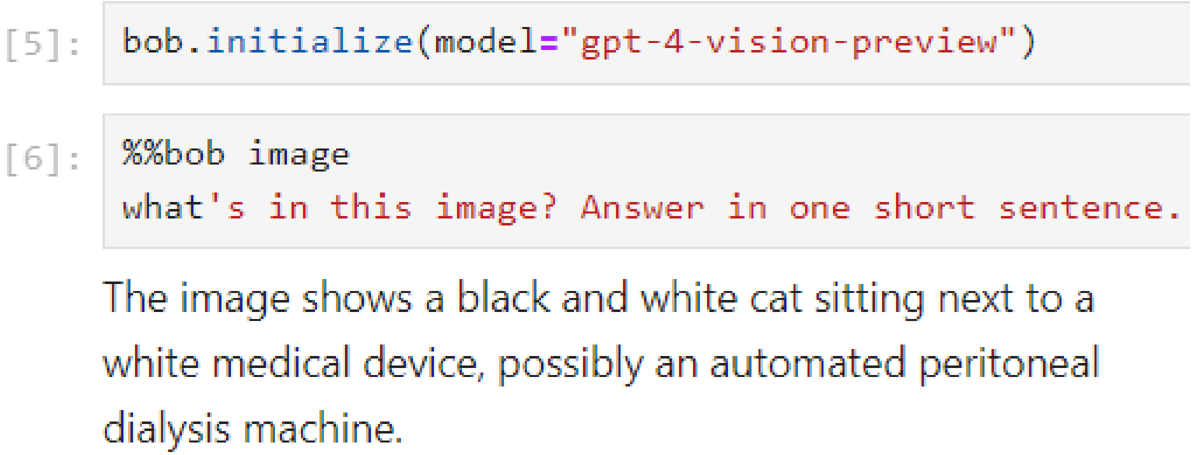
https://login.helmholtz.de/oauth2-as/oauth2-authz-web-entry
https://ollama.com
https://github.com/haesleinhuepf/bia-bob/blob/main/demo/custom_endpoints.ipynb

BiA-Bob: Vision models

Image-input, for interpreting natural images

- ChatGPT by OpenAI





```
[5]: bob.initialize(model="gpt-4-vision-preview")
```

```
[6]: %%bob image
     what's in this image? Answer in one short sentence.
```

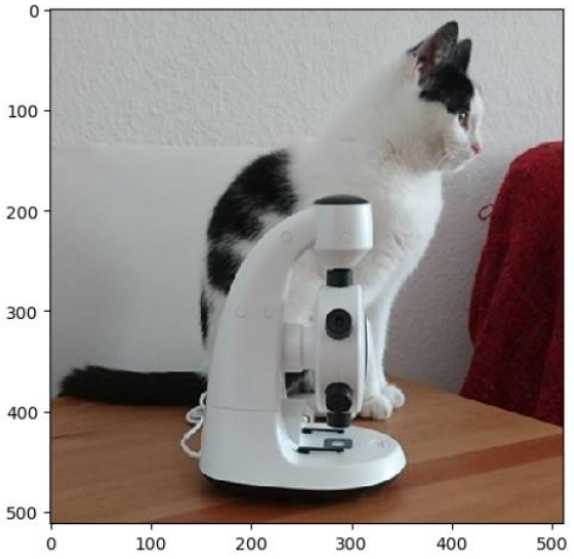
The image shows a black and white cat sitting next to a white medical device, possibly an automated peritoneal dialysis machine.

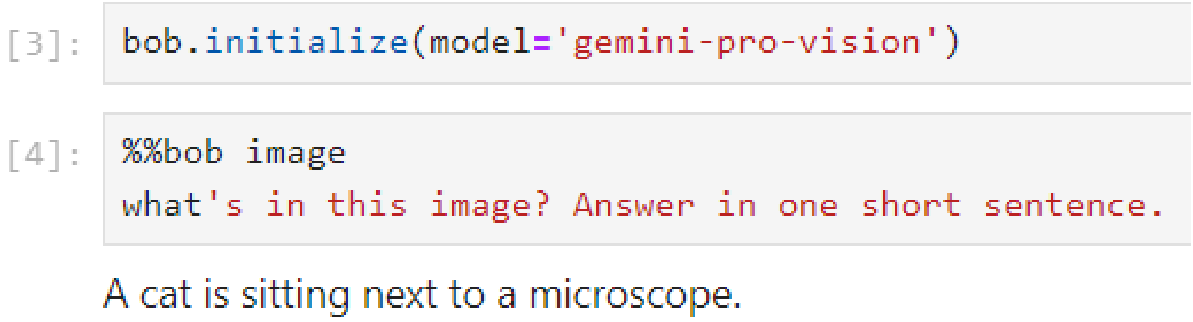
https://github.com/haesleinhuepf/bia-bob/blob/main/demo/vision.ipynb

BiA-Bob: Vision models 🤓

Image-input, for interpreting natural images

- Gemini by Google





```
[3]: bob.initialize(model='gemini-pro-vision')
```

```
[4]: %%bob image
     what's in this image? Answer in one short sentence.
```

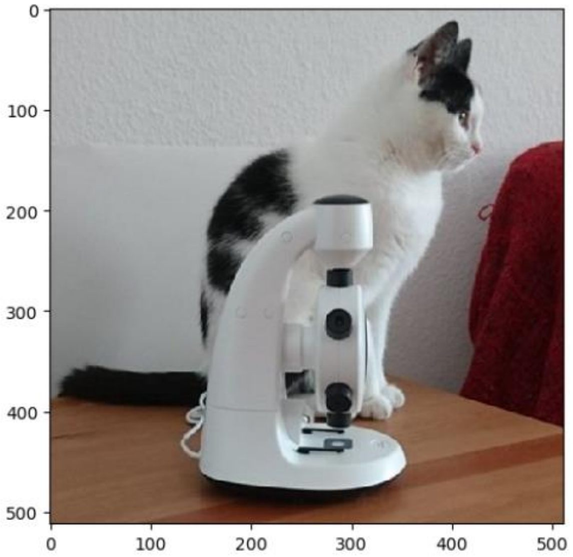
A cat is sitting next to a microscope.

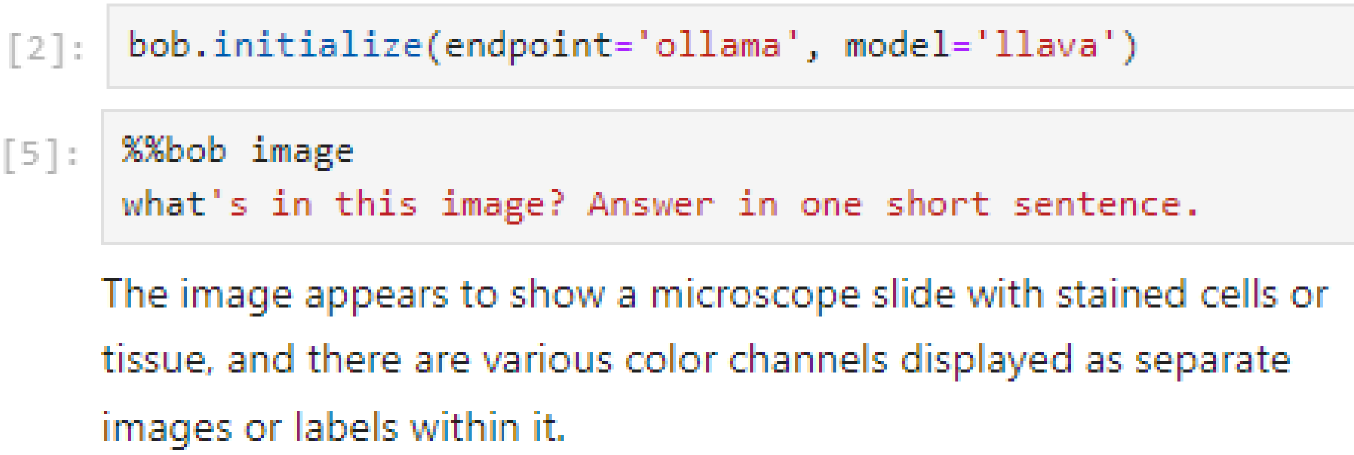
https://github.com/haesleinhuepf/bia-bob/blob/main/demo/vision.ipynb

BiA-Bob: Vision models 🤓

Image-input, for interpreting natural images

- Llava via ollama

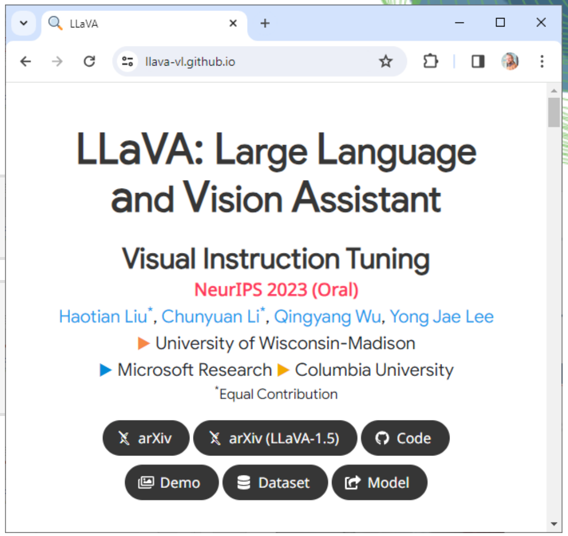




```
[2]: bob.initialize(endpoint='ollama', model='llava')
```

```
[5]: %%bob image
     what's in this image? Answer in one short sentence.
```

The image appears to show a microscope slide with stained cells or tissue, and there are various color channels displayed as separate images or labels within it.



LLaVA: Large Language and Vision Assistant

Visual Instruction Tuning

NeurIPS 2023 (Oral)

Haotian Liu*, Chunyuan Li*, Qingyang Wu, Yong Jae Lee

▶ University of Wisconsin-Madison

▶ Microsoft Research ▶ Columbia University

*Equal Contribution

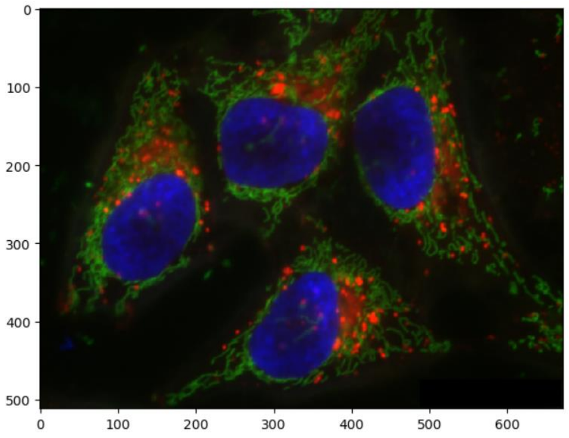
Work in progress, join efforts if you’re interested!

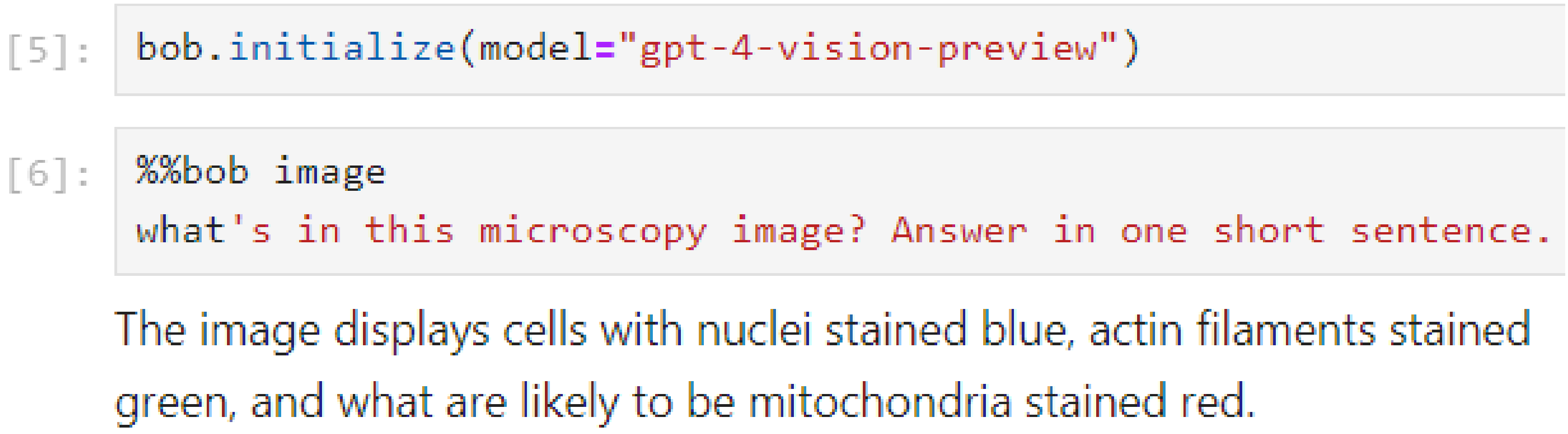
https://github.com/haesleinhuepf/bia-bob/pull/115
https://llava-vl.github.io/

BiA-Bob: Vision models 🤓

Image-input, to guide analysis / code generation

- ChatGPT by OpenAI





```
[5]:  bob.initialize(model="gpt-4-vision-preview")

[6]:  %%bob image
      what's in this microscopy image? Answer in one short sentence.
```

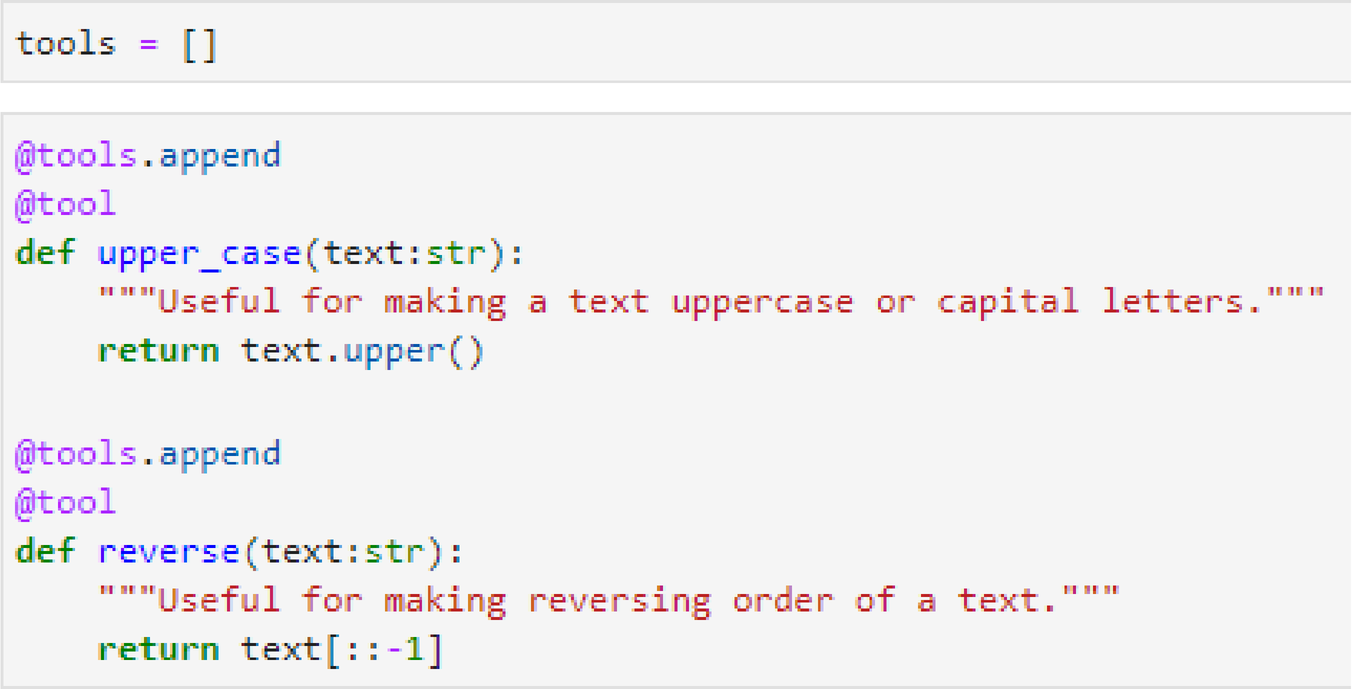
The image displays cells with nuclei stained blue, actin filaments stained green, and what are likely to be mitochondria stained red.

https://github.com/haesleinhuepf/bia-bob/blob/main/demo/vision-microscopy.ipynb

Function Calling using LangChain

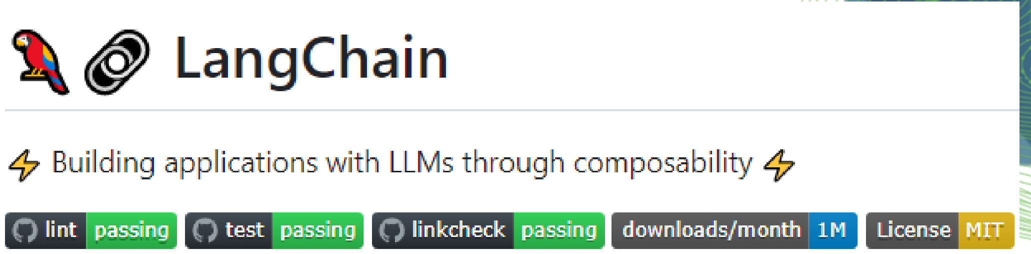
LangChain is used to combine tools.

It uses chatGPT under the hood.



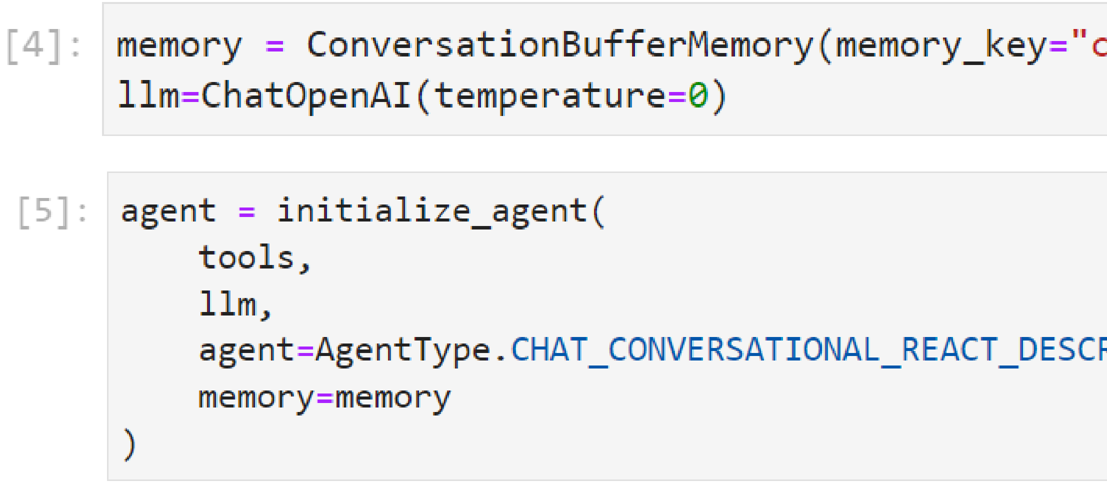
```
tools = []
```

```
@tools.append
@tool
def upper_case(text:str):
    """Useful for making a text uppercase or capital letters."""
    return text.upper()

@tools.append
@tool
def reverse(text:str):
    """Useful for making reversing order of a text."""
    return text[::-1]
```



🦜🔗 LangChain

⚡ Building applications with LLMs through composability ⚡



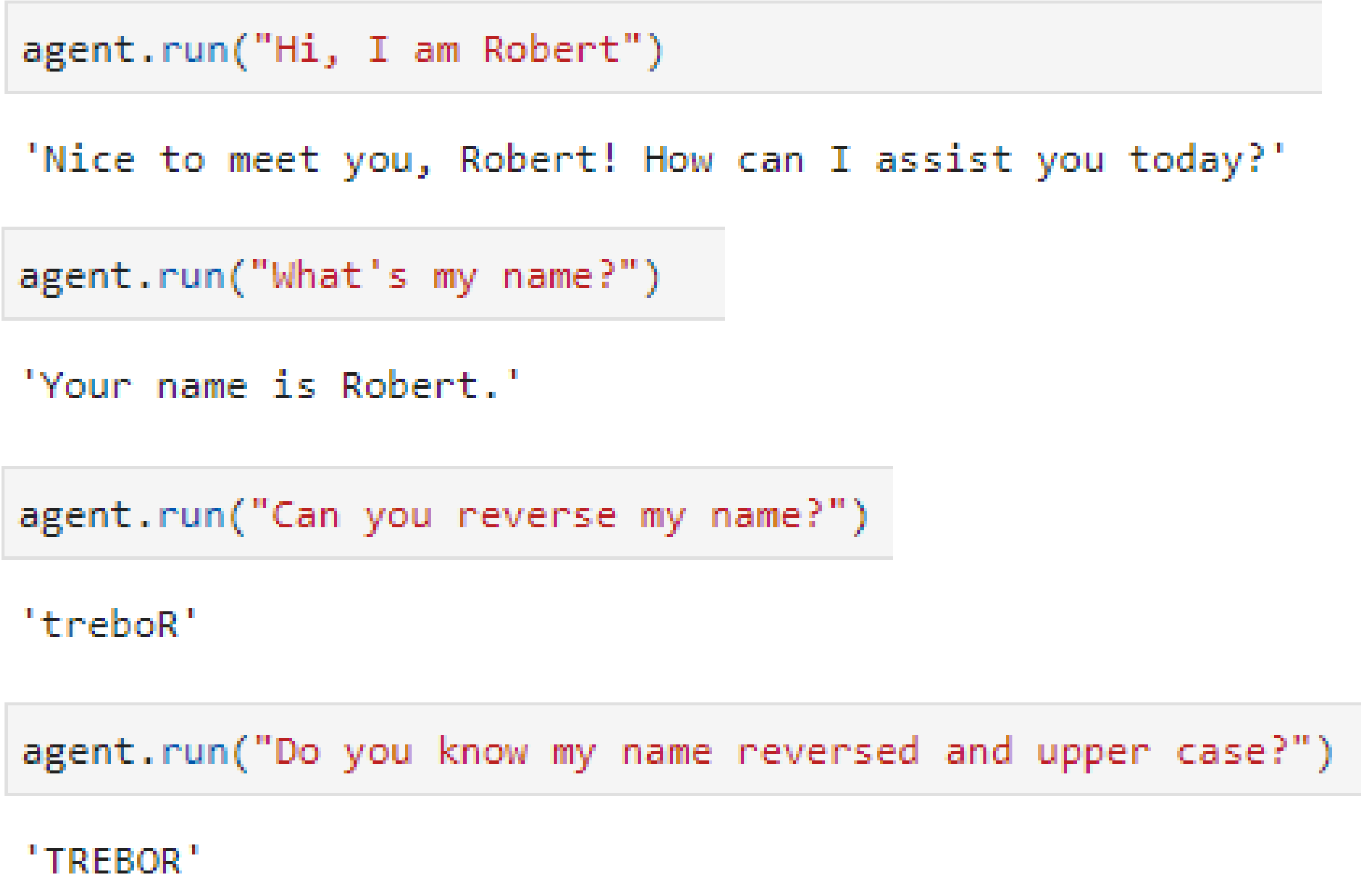
```
[4]: memory = ConversationBufferMemory(memory_key="c
     llm=ChatOpenAI(temperature=0)
```

```
[5]: agent = initialize_agent(
         tools,
         llm,
         agent=AgentType.CHAT_CONVERSATIONAL_REACT_DESCR
         memory=memory
     )
```

https://github.com/langchain-ai/langchain
https://scads.github.io/prompt-engineering-tutorial-2023/01_prompts/07_langchain.html

Function Calling using LangChain

After combining tools, large langue model and memory in an *agent*, you can interact with it.



```
agent.run("Hi, I am Robert")
```

```
'Nice to meet you, Robert! How can I assist you today?'
```

```
agent.run("What's my name?")
```

```
'Your name is Robert.'
```

```
agent.run("Can you reverse my name?")
```

```
'treboR'
```

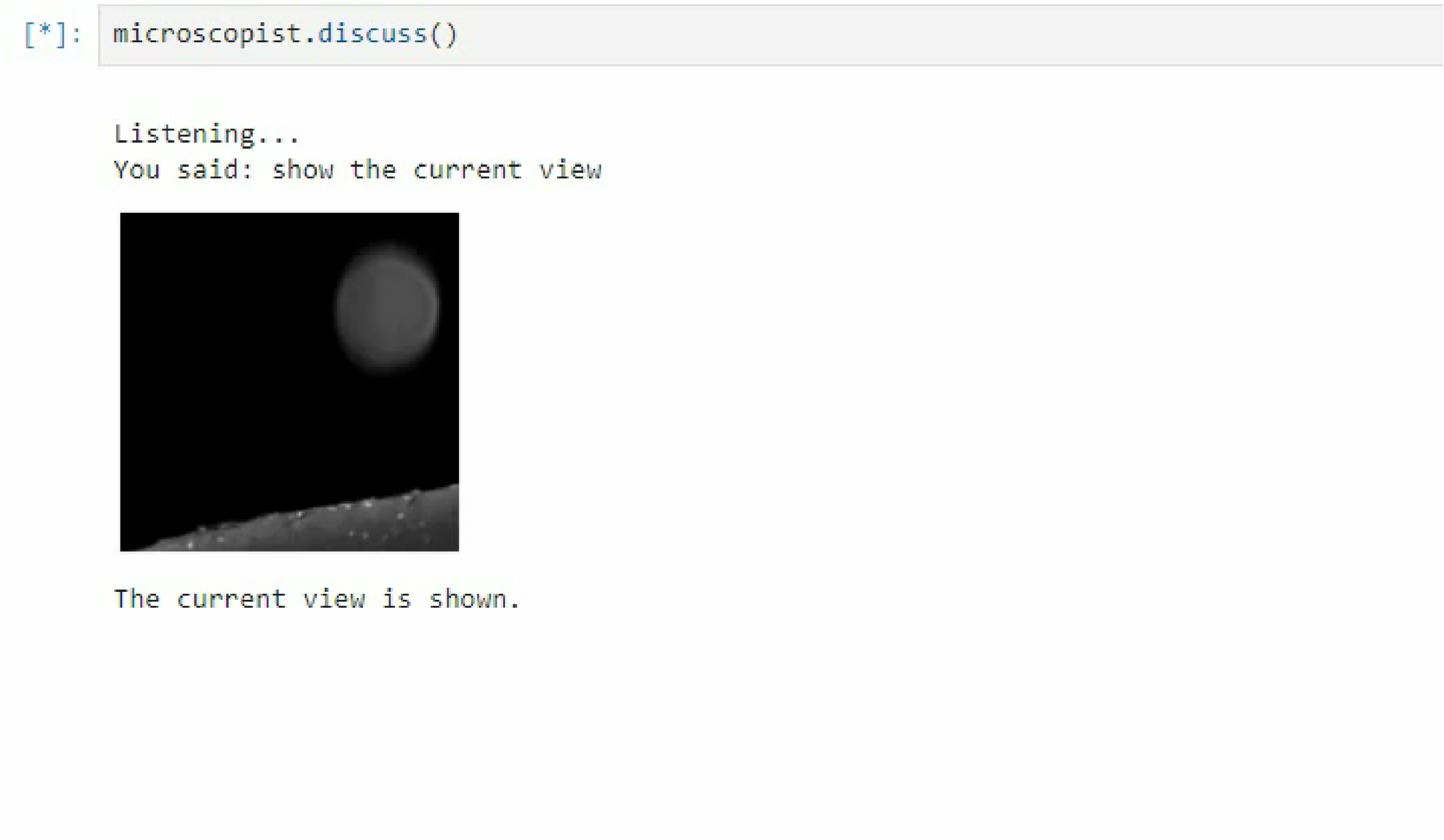
```
agent.run("Do you know my name reversed and upper case?")
```

```
'TREBOR'
```

https://github.com/langchain-ai/langchain
https://scads.github.io/prompt-engineering-tutorial-2023/01_prompts/07_langchain.html

Voice Assistance

Combining voice recognition with large language models



```
[*]: microscopist.discuss()
```

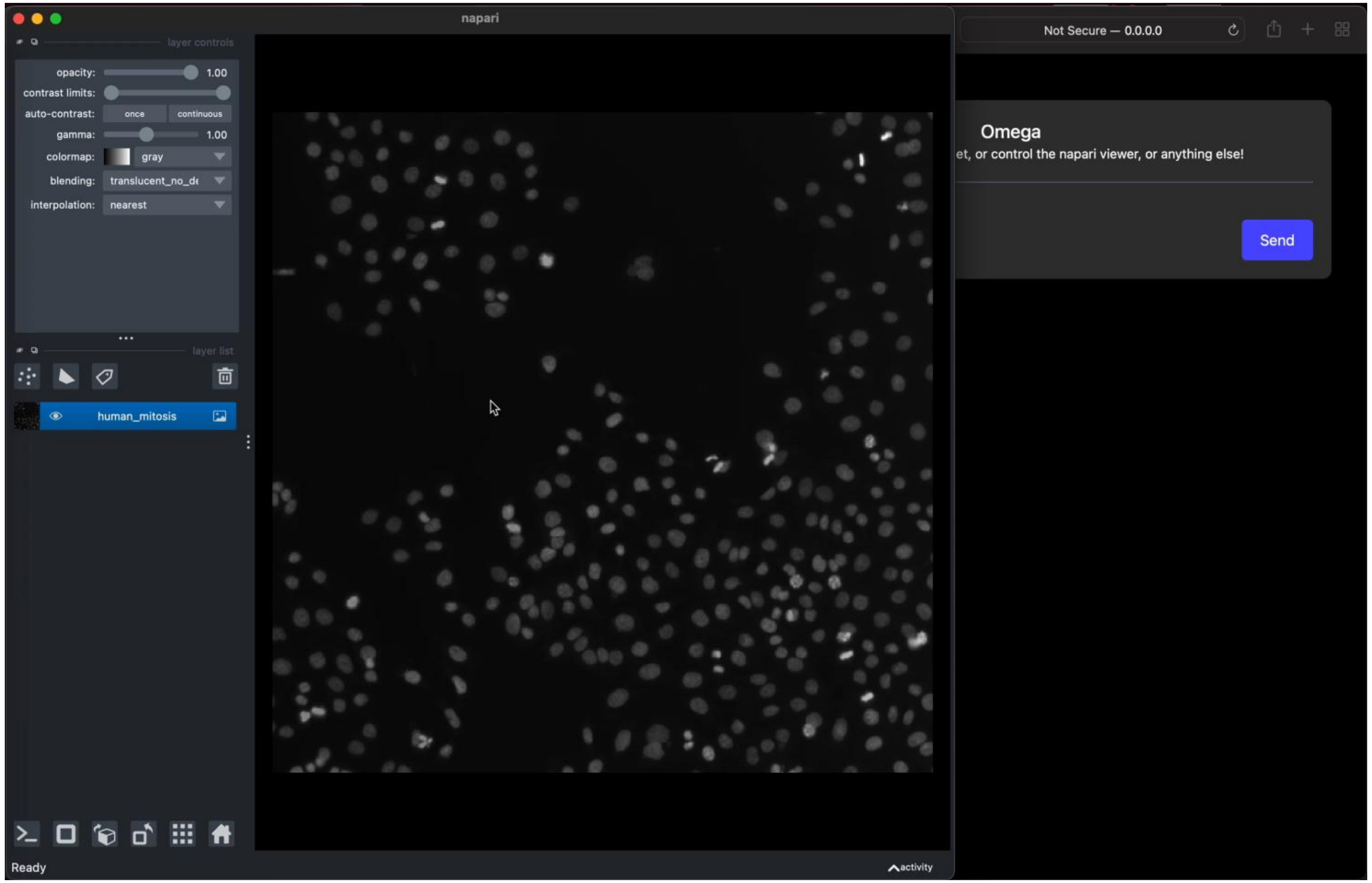
```
Listening...
You said: show the current view
```

```
The current view is shown.
```

https://github.com/haesleinhuepf/blablado
https://github.com/haesleinhuepf/blablado/blob/main/demo/microscope_stage_demo.ipynb

napari-chatGPT

Napari-chatGPT can automate programming plugins / “widgets”.


napari

opacity: 1.00
contrast limits:
auto-contrast: once continuous
gamma: 1.00
colormap: gray
blending: translucent_no_de
interpolation: nearest

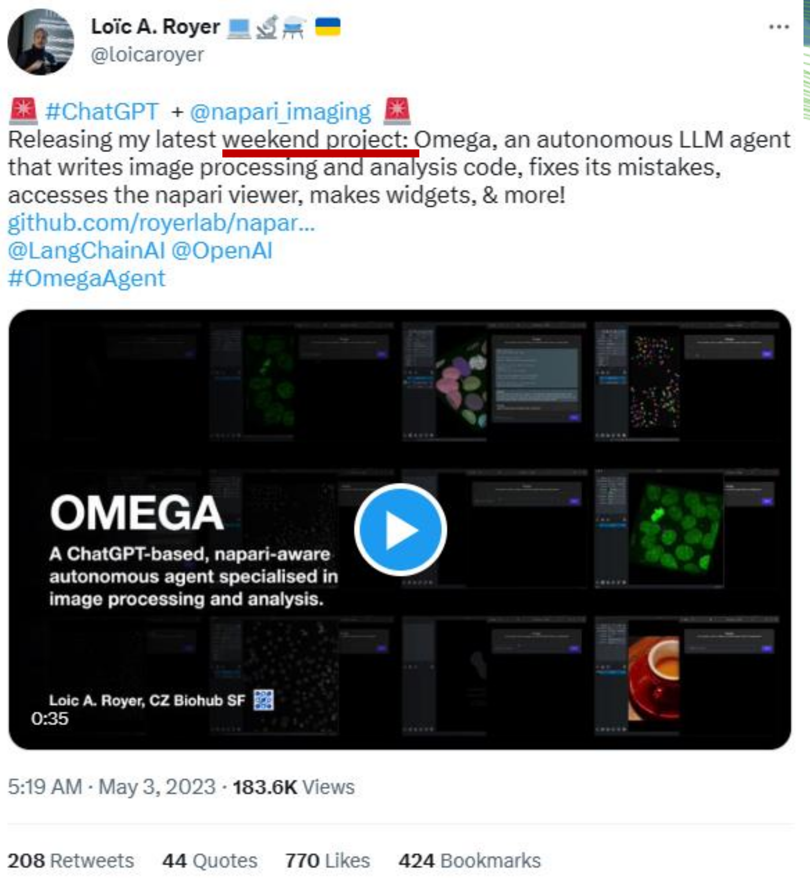
human_mitosis

Ready

Not Secure — 0.0.0.0

Omega

et, or control the napari viewer, or anything else!


Loïc A. Royer 💻🔬🪑
@loicaroyer

🚨 #ChatGPT + @napari_imaging 🚨
Releasing my latest weekend project: Omega, an autonomous LLM agent that writes image processing and analysis code, fixes its mistakes, accesses the napari viewer, makes widgets, & more!
github.com/royerlab/napar...
@LangChainAI @OpenAI
#OmegaAgent

OMEGA
A ChatGPT-based, napari-aware autonomous agent specialised in image processing and analysis.
Loic A. Royer, CZ Biohub SF
0:35

5:19 AM · May 3, 2023 · 183.6K Views

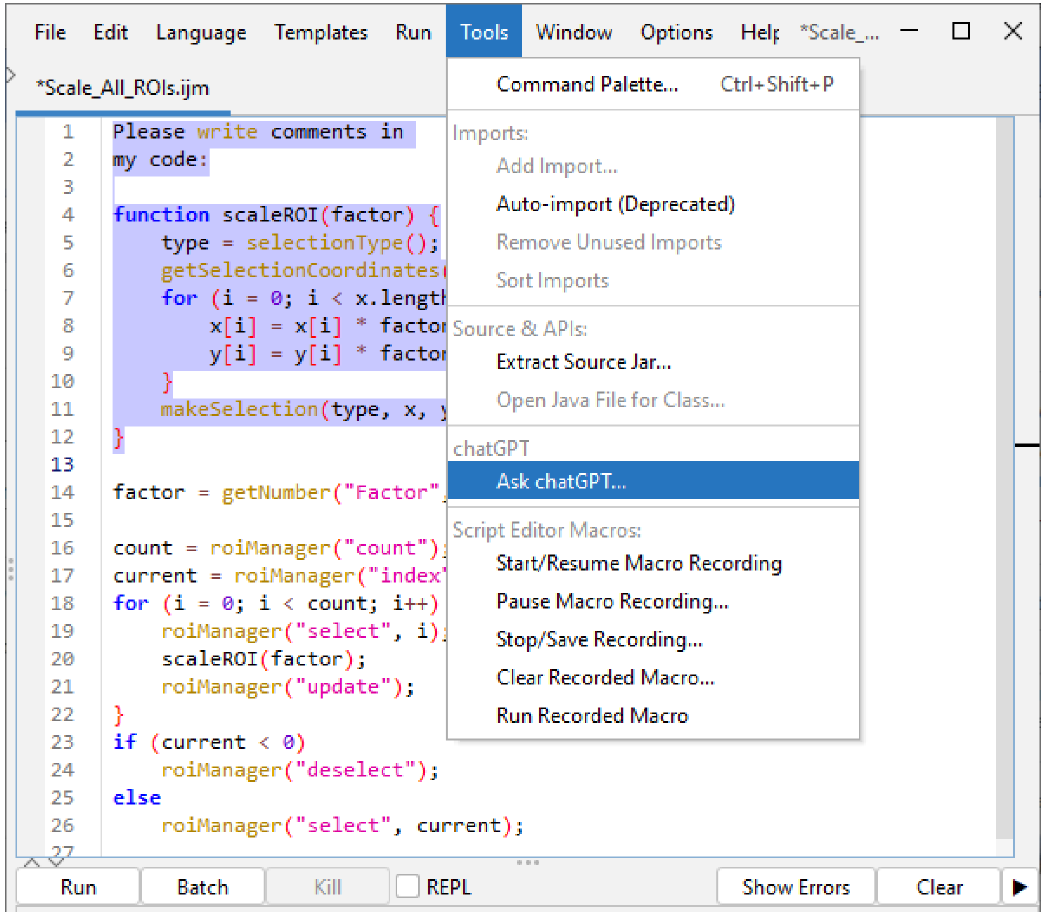
208 Retweets 44 Quotes 770 Likes 424 Bookmarks

Source: https://github.com/royerlab/napari-chatgpt (BSD3 license)
https://twitter.com/loicaroyer/status/1653600252807757824

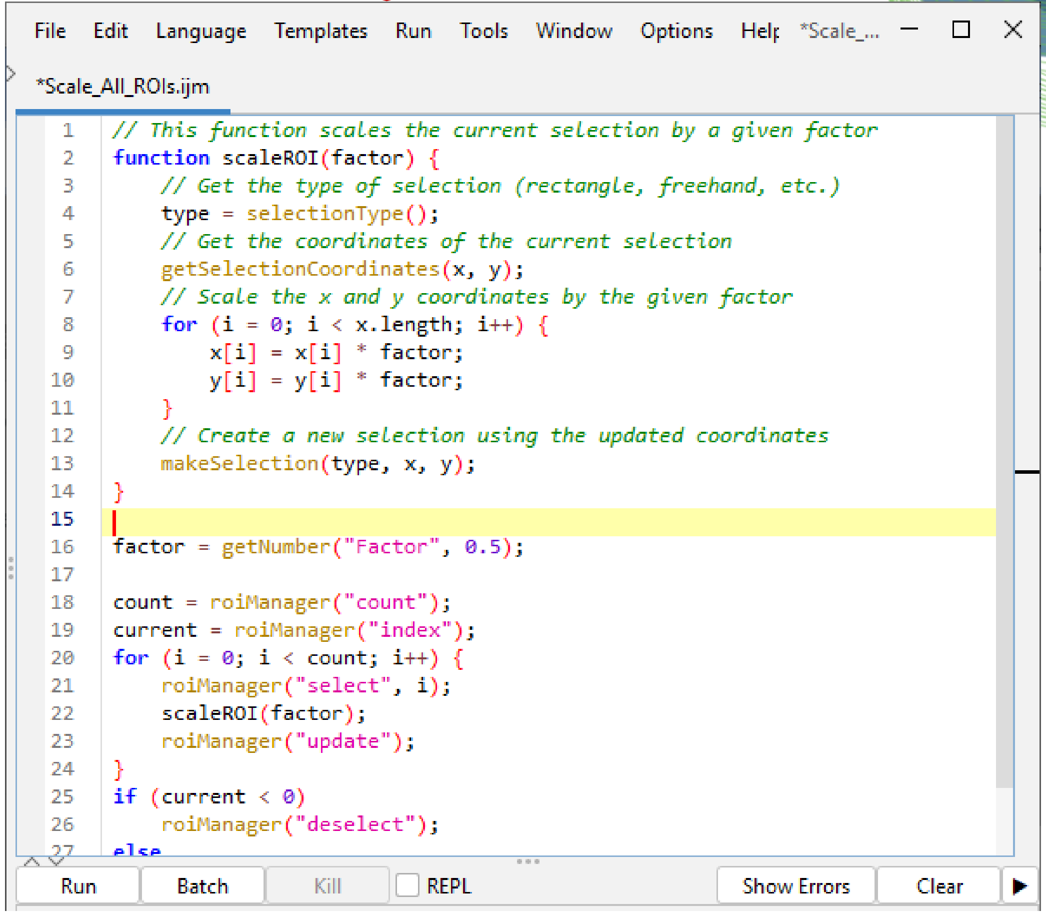
ChatGPT @Fiji's Script-Editor

... will also come at some point

ChatGPT's knowledge about ImageJ and Macro is limited



```
Please write comments in
my code:

function scaleROI(factor) {
    type = selectionType();
    getSelectionCoordinates(
    for (i = 0; i < x.length
        x[i] = x[i] * factor
        y[i] = y[i] * factor
    }
    makeSelection(type, x, y
}

factor = getNumber("Factor"

count = roiManager("count")
current = roiManager("index"
for (i = 0; i < count; i++)
    roiManager("select", i)
    scaleROI(factor);
    roiManager("update");
}
if (current < 0)
    roiManager("deselect");
else
    roiManager("select", current);
```



```
// This function scales the current selection by a given factor
function scaleROI(factor) {
    // Get the type of selection (rectangle, freehand, etc.)
    type = selectionType();
    // Get the coordinates of the current selection
    getSelectionCoordinates(x, y);
    // Scale the x and y coordinates by the given factor
    for (i = 0; i < x.length; i++) {
        x[i] = x[i] * factor;
        y[i] = y[i] * factor;
    }
    // Create a new selection using the updated coordinates
    makeSelection(type, x, y);
}

factor = getNumber("Factor", 0.5);

count = roiManager("count");
current = roiManager("index");
for (i = 0; i < count; i++) {
    roiManager("select", i);
    scaleROI(factor);
    roiManager("update");
}
if (current < 0)
    roiManager("deselect");
else
```

https://github.com/scijava/script-editor/pull/67

What else can [multi-modal] LLMs do?

Generate images, e.g. for augmenting data

Potentially useful to make algorithms more robust

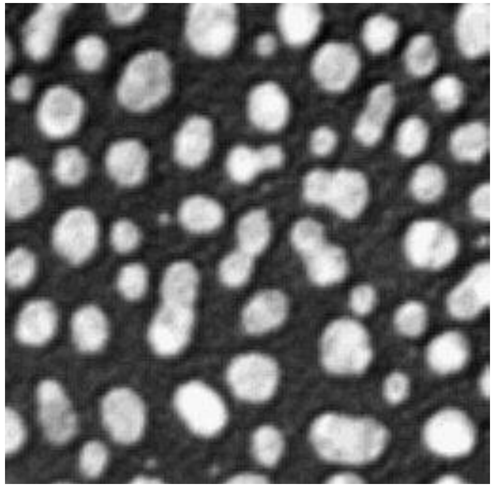
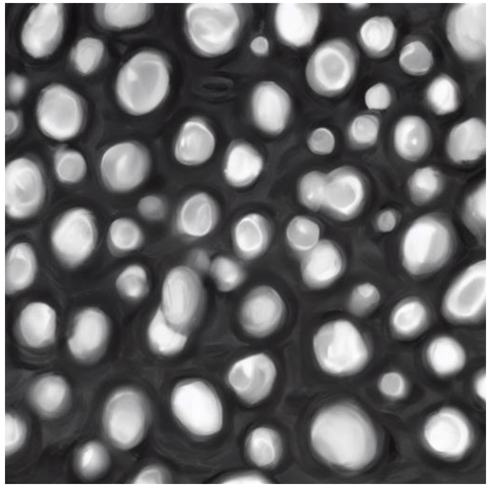
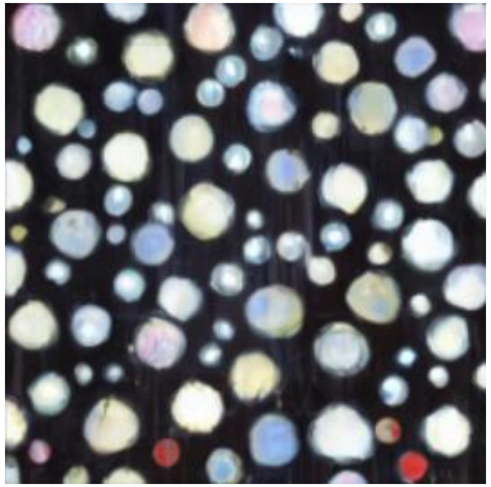
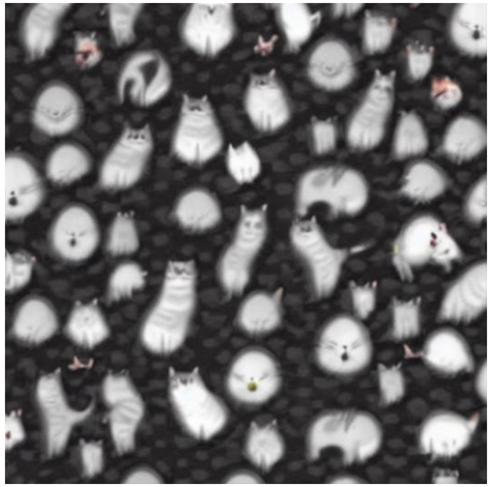


Image Variation



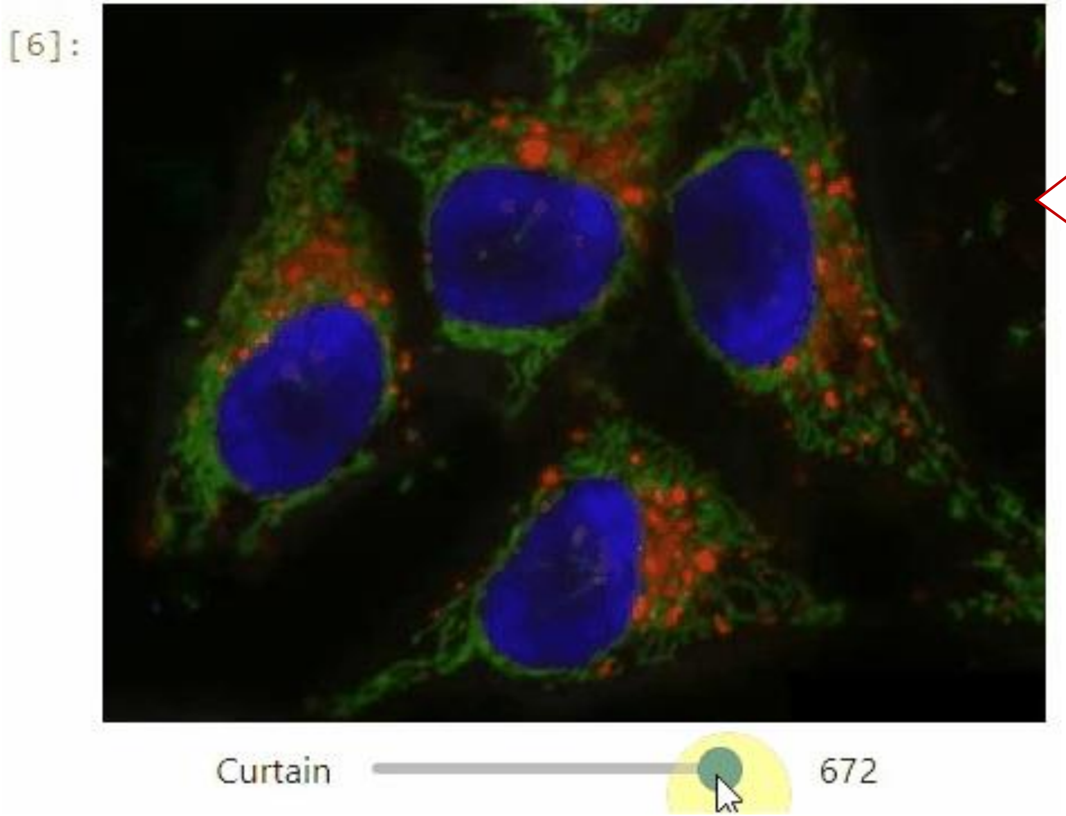




https://haesleinhuepf.github.io/BioImageAnalysisNotebooks/07_prompt_engineering/60_image_variations_huggingface.html

New technologies bring new risks...

If you can generate images,
you can also generate parts of images....



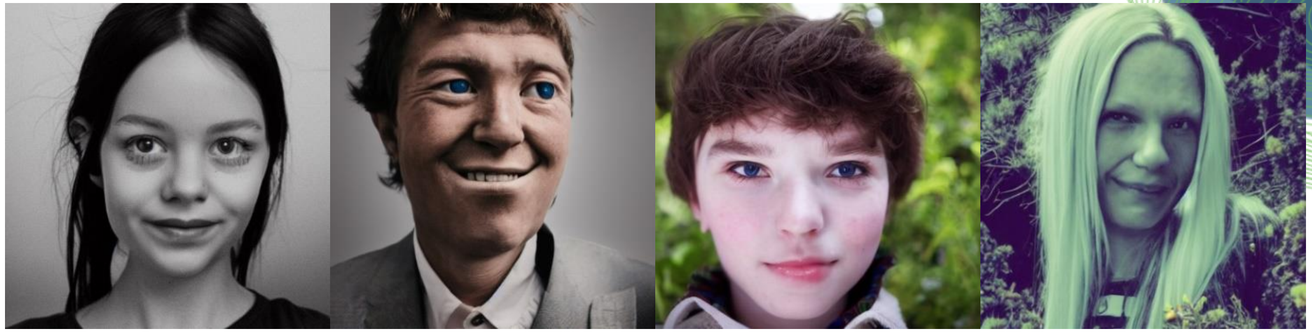
Interesting challenges for our community ahead

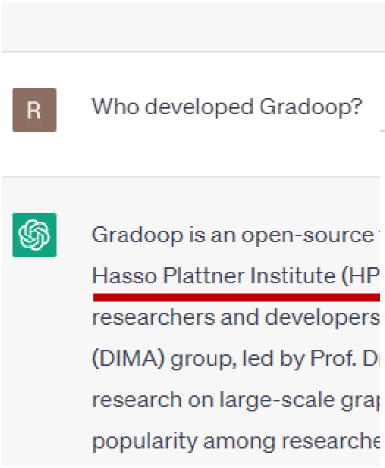
https://github.com/haesleinhuepf/darth-d/blob/main/demo/demo_replacing.ipynb

Generative Artificial Intelligence

Challenges

- Data safety / security / integrity
- Computational cost of training of neural networks
  - CO2-footprint/climate change
  - Accessibility
- Bias: “a nice photo of a human”
- Hallucinations
- False information / fake news
- Glitch tokens
- Debugging is hard



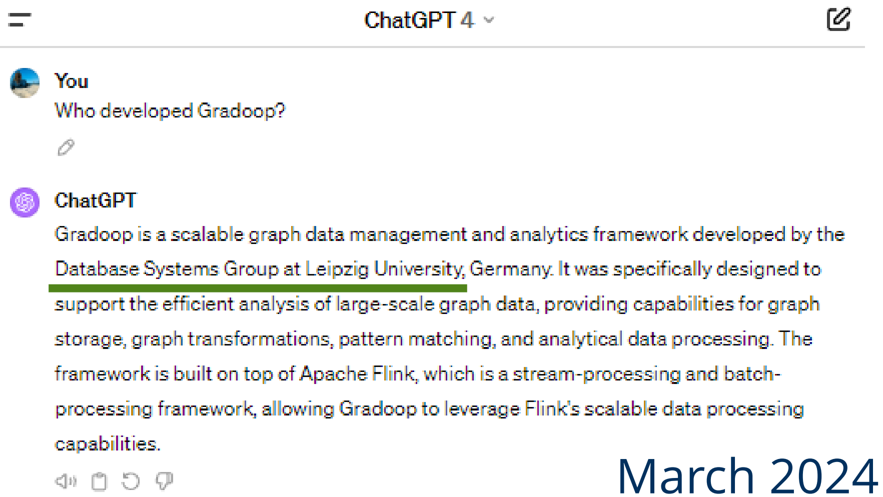


Model: Default (GPT-3.5)

R Who developed Gradoop?

Gradoop is an open-source
Hasso Plattner Institute (HP
researchers and developers
(DIMA) group, led by Prof. D
research on large-scale gra
popularity among researche

May 2023



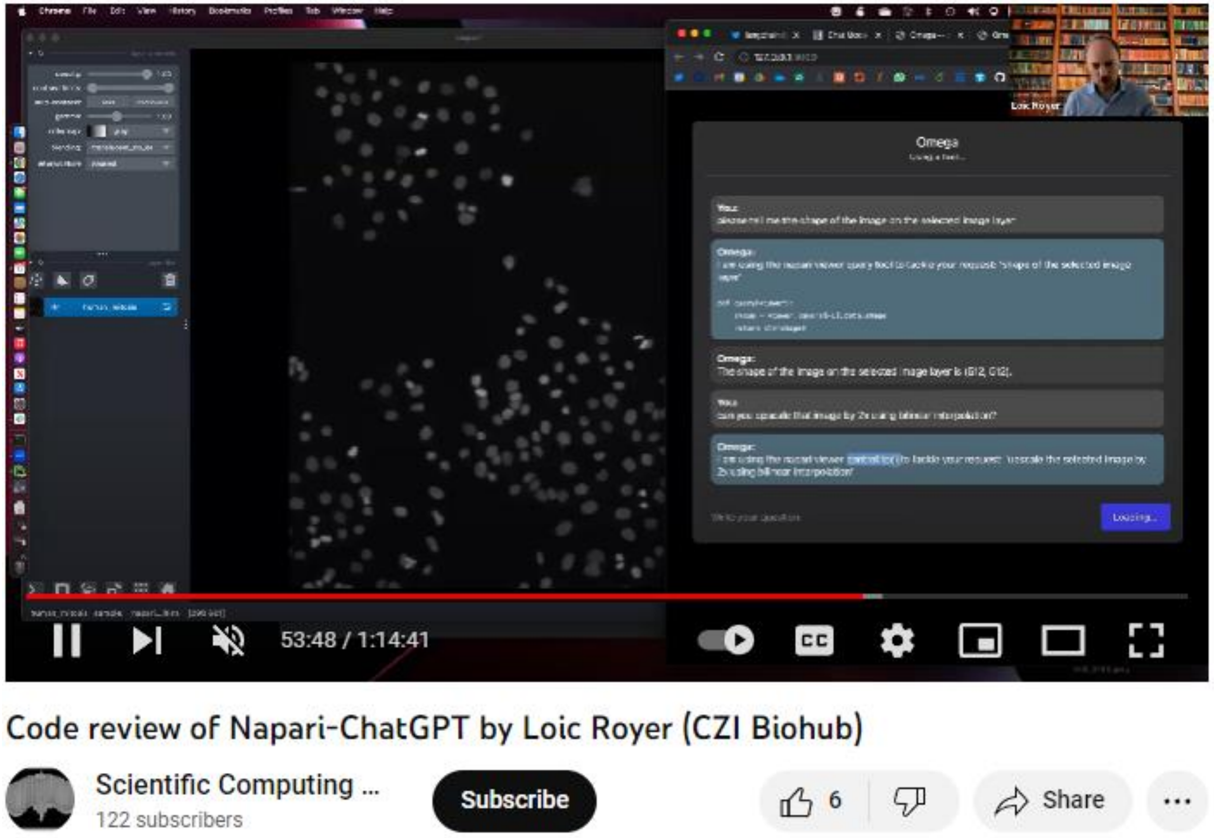
You
Who developed Gradoop?

ChatGPT
Gradoop is a scalable graph data management and analytics framework developed by the Database Systems Group at Leipzig University, Germany. It was specifically designed to support the efficient analysis of large-scale graph data, providing capabilities for graph storage, graph transformations, pattern matching, and analytical data processing. The framework is built on top of Apache Flink, which is a stream-processing and batch-processing framework, allowing Gradoop to leverage Flink's scalable data processing capabilities.

March 2024

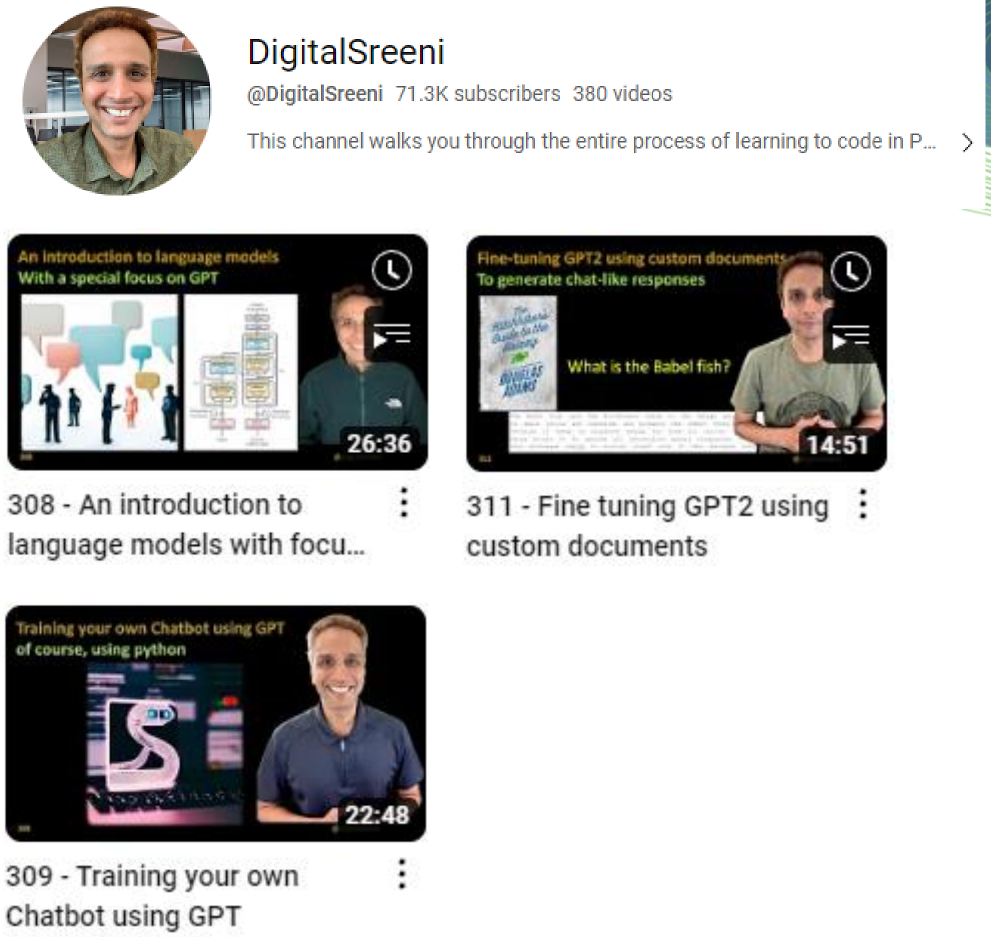
Sources: https://replicate.com/stability-ai/stable-diffusion https://chat.openai.com/

Further reading / watching



Code review of Napari-ChatGPT by Loic Royer (CZI Biohub)

Scientific Computing ...
122 subscribers


An introduction to language models
With a special focus on GPT
26:36
Fine-tuning GPT2 using custom documents
To generate chat-like responses
What is the Babel fish?
14:51
Training your own Chatbot using GPT
of course, using python
22:48

DigitalSreeni
@DigitalSreeni 71.3K subscribers 380 videos
This channel walks you through the entire process of learning to code in P...

308 - An introduction to language models with focu...

311 - Fine tuning GPT2 using custom documents

309 - Training your own Chatbot using GPT

https://www.youtube.com/watch?v=JMo6Sn-L_j4

https://www.youtube.com/c/digitalsreeni

Acknowledgements

Communities &
platforms

image.sc

NFDI4
BIOIMAGE

GloBIAS
global bioimage analysts' society

neubias
network of european
bioimage analysts

FocalPlane
Where biology meets microscopy

BiAPoL team

- Marcelo Zoccoler
- Johannes Soltwedel
- Maleeha Hassan
- Stefan Hahmann
- Somashekhar Kulkarni

Former lab members:

- Ryan George Savill
- Laura Zigutyte
- Mara Lampert
- Allyson Ryan
- Conni Wetzker
- Till Korten

Funding

Bundesministerium
für Bildung
und Forschung

SACHSEN

Diese Maßnahme wird gefördert durch die Bundesregierung aufgrund eines Beschlusses des Deutschen Bundestages. Diese Maßnahme wird mitfinanziert durch Steuermittel auf der Grundlage des von den Abgeordneten des Sächsischen Landtags beschlossenen Haushaltes.

Chan
Zuckerberg
Initiative

DFG Deutsche
Forschungsgemeinschaft