ScaDS.AI
DRESDEN LEIPZIG

CENTER FOR SCALABLE DATA ANALYTICS AND ARTIFICIAL INTELLIGENCE

Large Language Models for Bio-image Analysis

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.

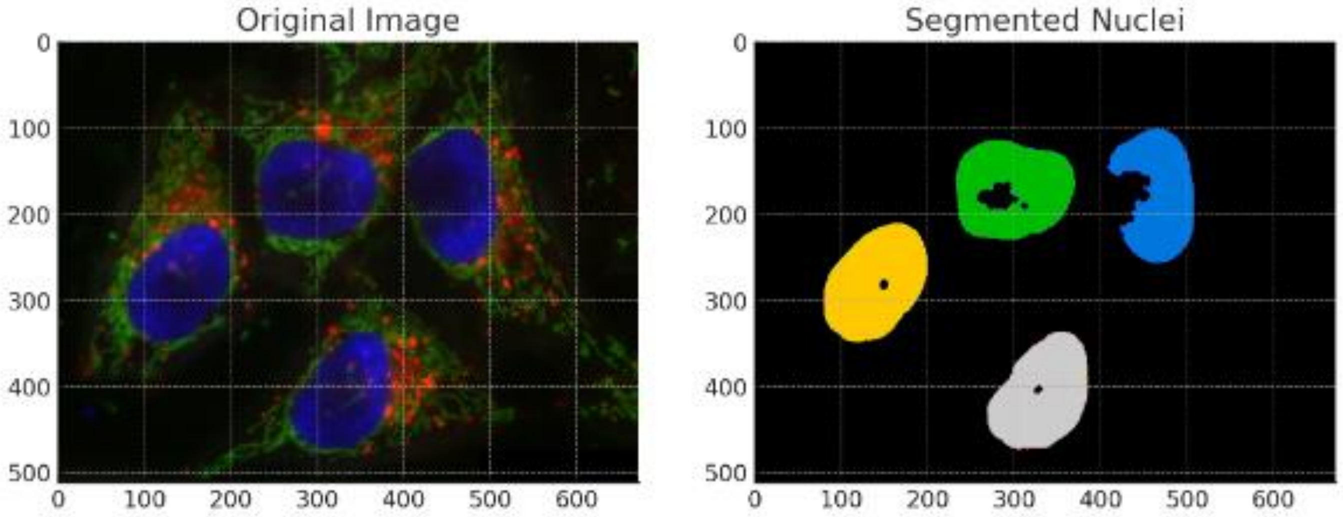
These slides can be reused under the terms of the CC-BY 4.0 license unless mentioned otherwise.

TECHNISCHE UNIVERSITÄT DRESDEN

UNIVERSITÄT LEIPZIG

Bio-image Analysis

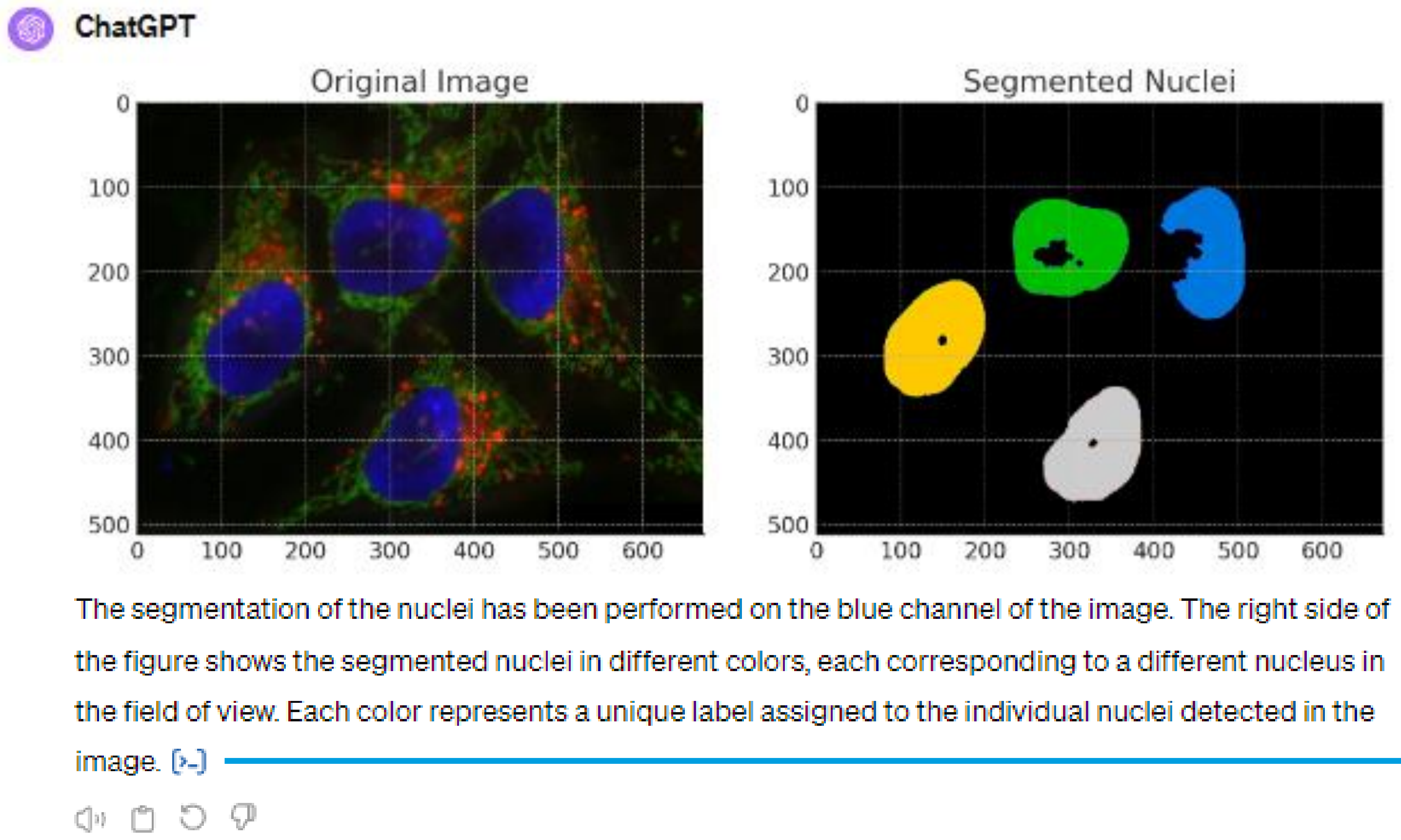
My job …



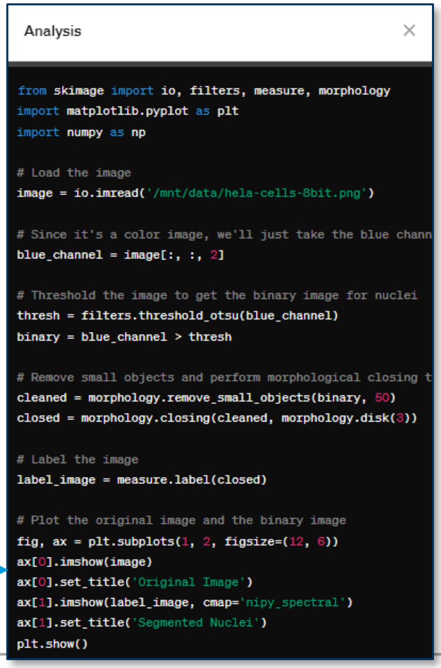
Bio-image Analysis using Large-Language Models

My job is changing, since we have ChatGPT

Prompt:
“Please segment the nuclei in the blue channel of the image”



The segmentation of the nuclei has been performed on the blue channel of the image. The right side of the figure shows the segmented nuclei in different colors, each corresponding to a different nucleus in the field of view. Each color represents a unique label assigned to the individual nuclei detected in the image. [>_]



```
from skimage import io, filters, measure, morphology
import matplotlib.pyplot as plt
import numpy as np

# Load the image
image = io.imread('/mnt/data/hela-cells-8bit.png')

# Since it's a color image, we'll just take the blue chann
blue_channel = image[:, :, 2]

# Threshold the image to get the binary image for nuclei
thresh = filters.threshold_otsu(blue_channel)
binary = blue_channel > thresh

# Remove small objects and perform morphological closing t
cleaned = morphology.remove_small_objects(binary, 50)
closed = morphology.closing(cleaned, morphology.disk(3))

# Label the image
label_image = measure.label(closed)

# Plot the original image and the binary image
fig, ax = plt.subplots(1, 2, figsize=(12, 6))
ax[0].imshow(image)
ax[0].set_title('Original Image')
ax[1].imshow(label_image, cmap='nipy_spectral')
ax[1].set_title('Segmented Nuclei')
plt.show()
```

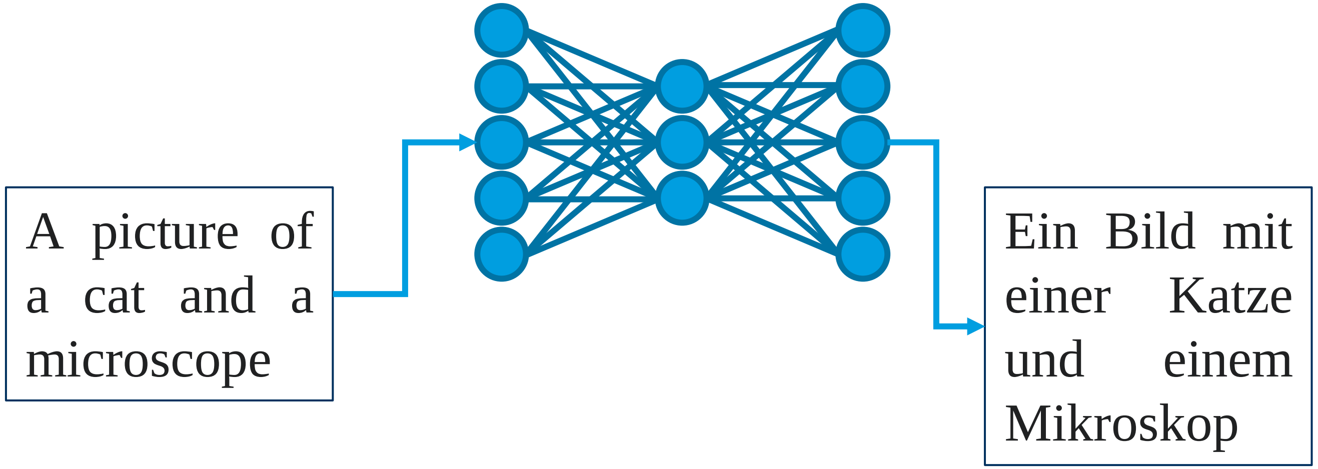
Generative Artificial Intelligence

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
- Image interpretation / analysis



A picture of a cat and a microscope

Ein Bild mit einer Katze und einem Mikroskop

---

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

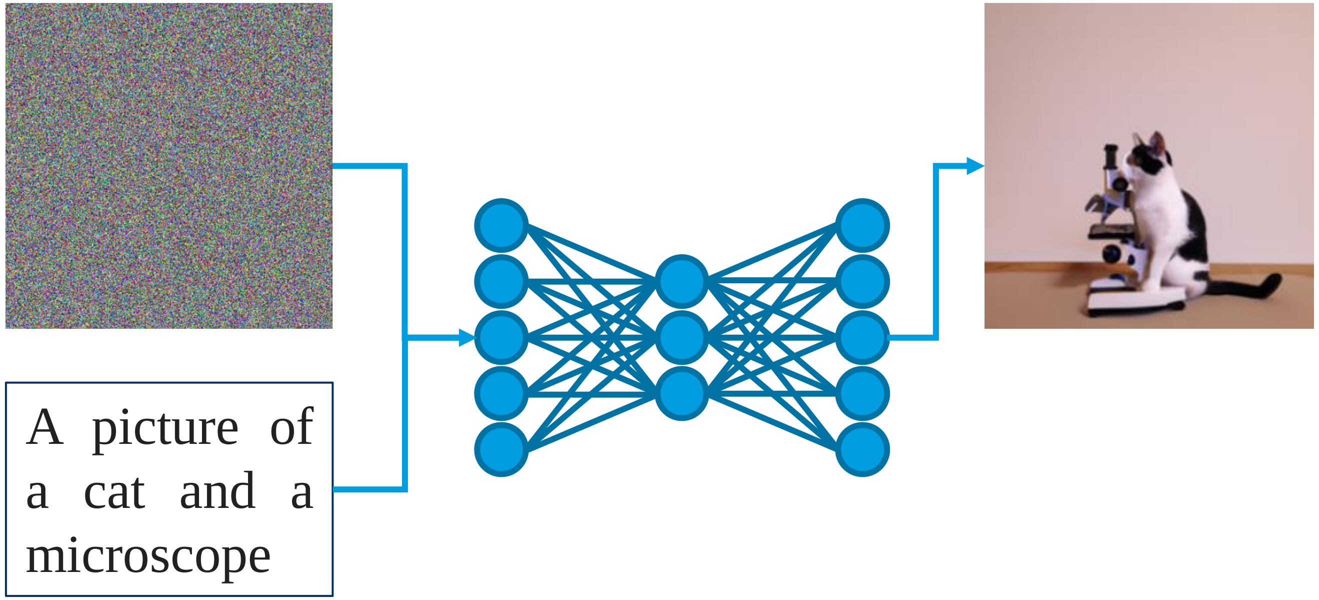
Generative Artificial Intelligence

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
- Image interpretation / analysis



A picture of a cat and a microscope

---

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

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]

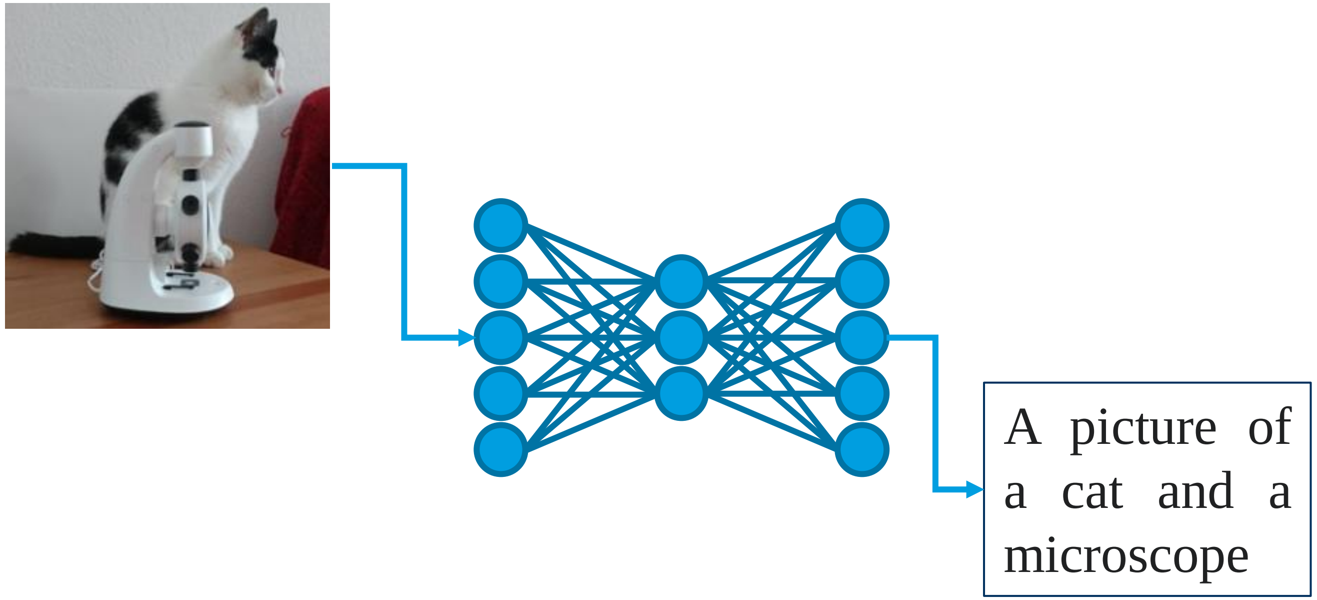
Commonly based on Neural Networks

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
- Image interpretation / analysis



A picture of a cat and a microscope

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

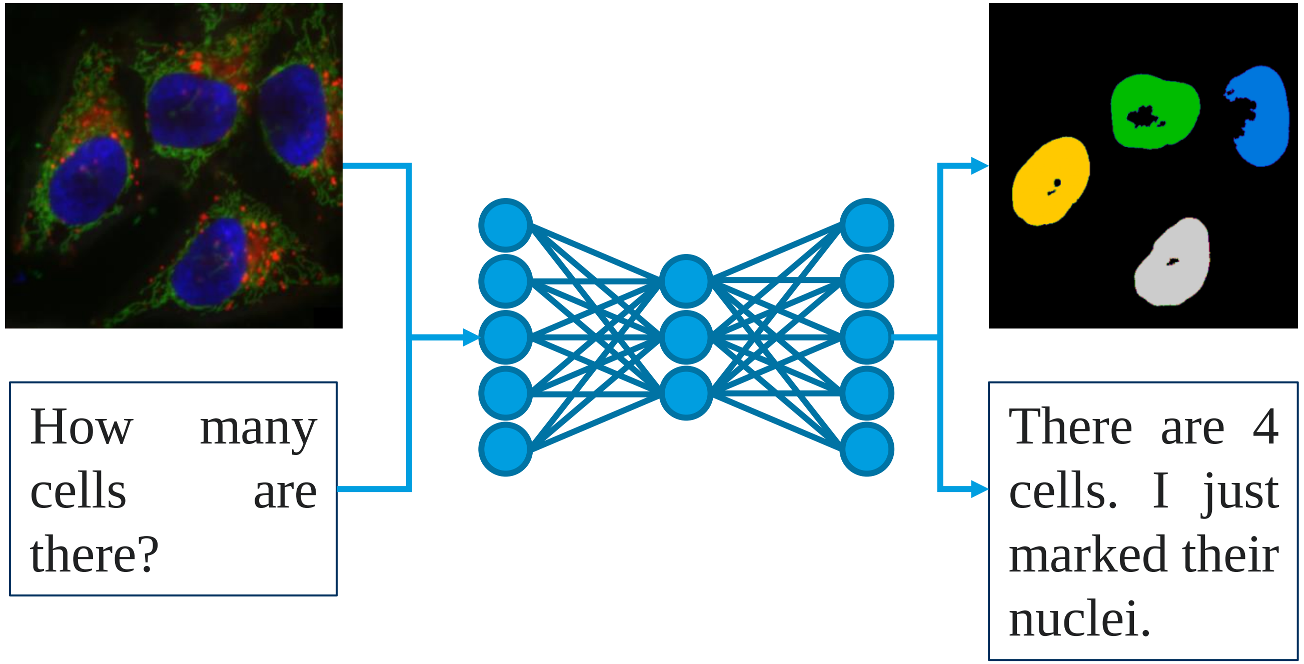
Generative Artificial Intelligence

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
- Image interpretation / analysis



How many cells are there?

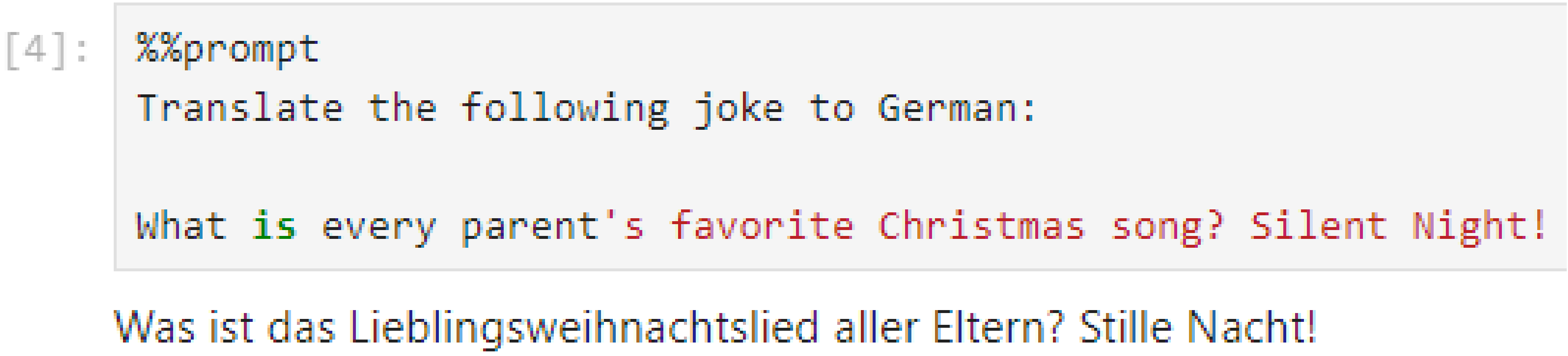
There are 4 cells. I just marked their nuclei.

---

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

What are large language models good in?

Translation tasks

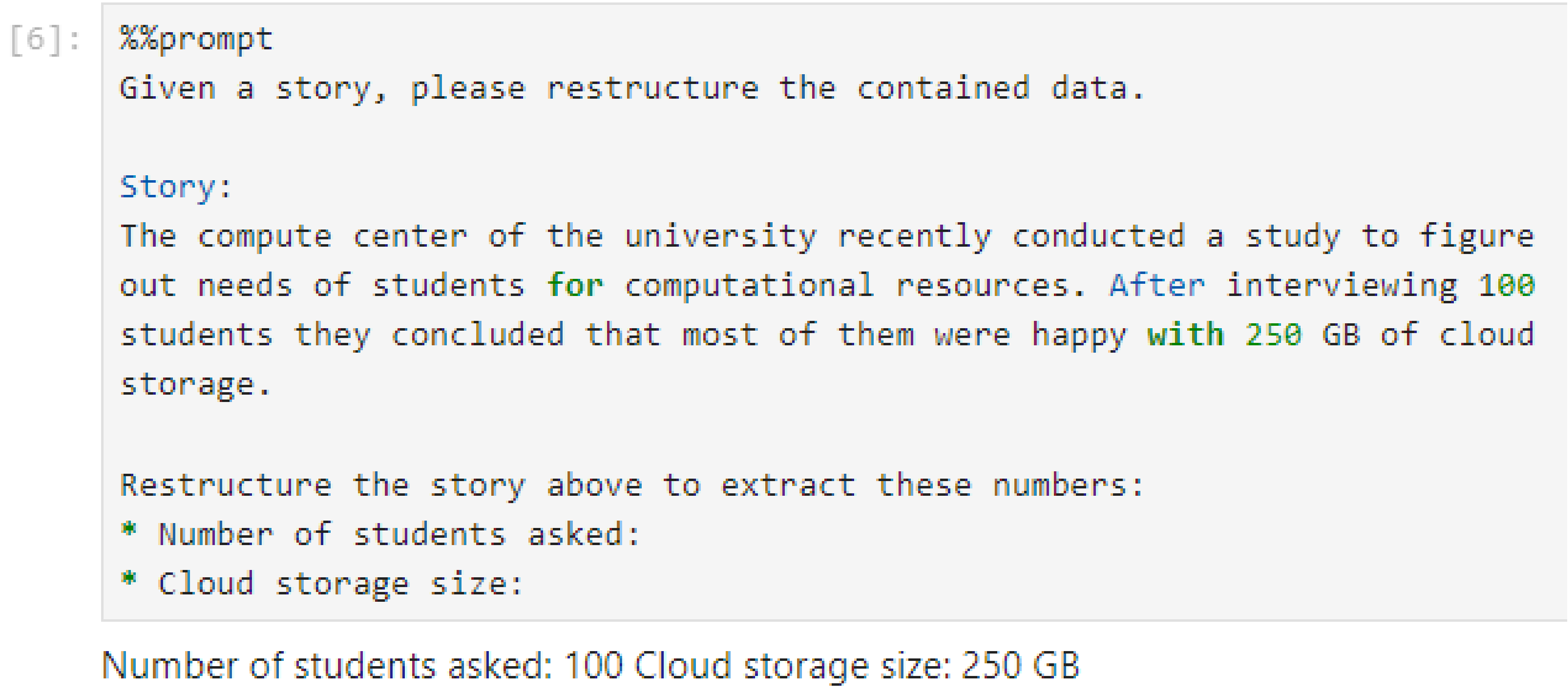


```
[4]: %%prompt
Translate the following joke to German:

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

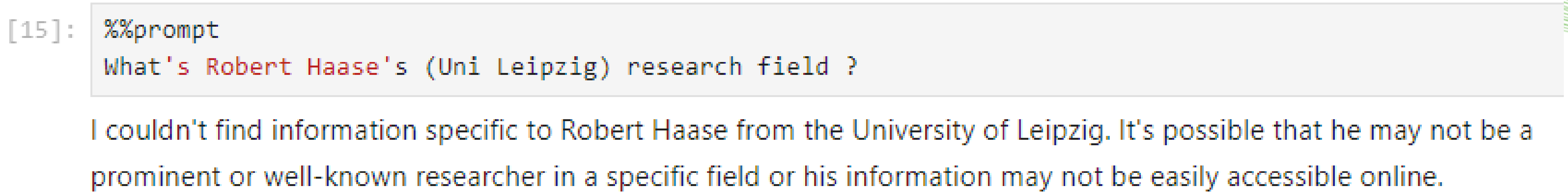


```
[6]: %%prompt
Given a story, please restructure the contained data.

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
```

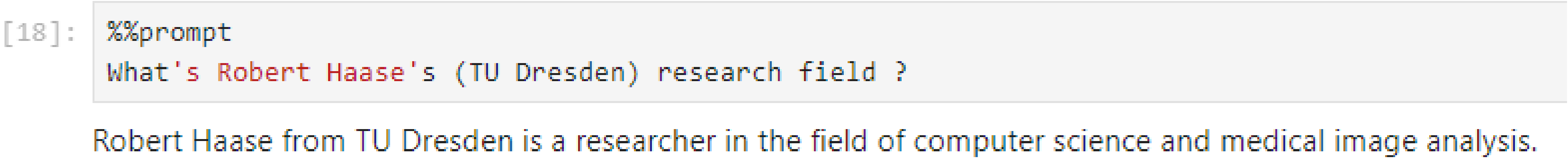
What are large language models not so good in?

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.

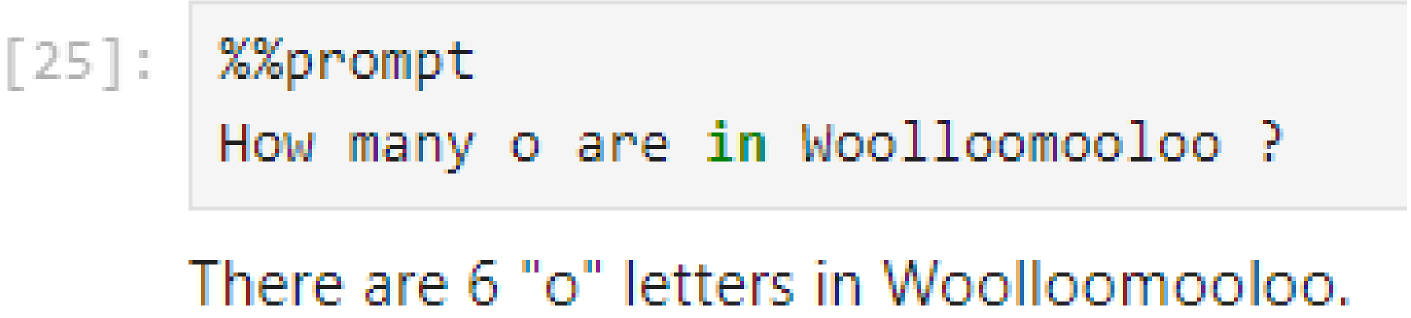


```
[18]: %%prompt
      What's Robert Haase's (TU Dresden) research field ?
```

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

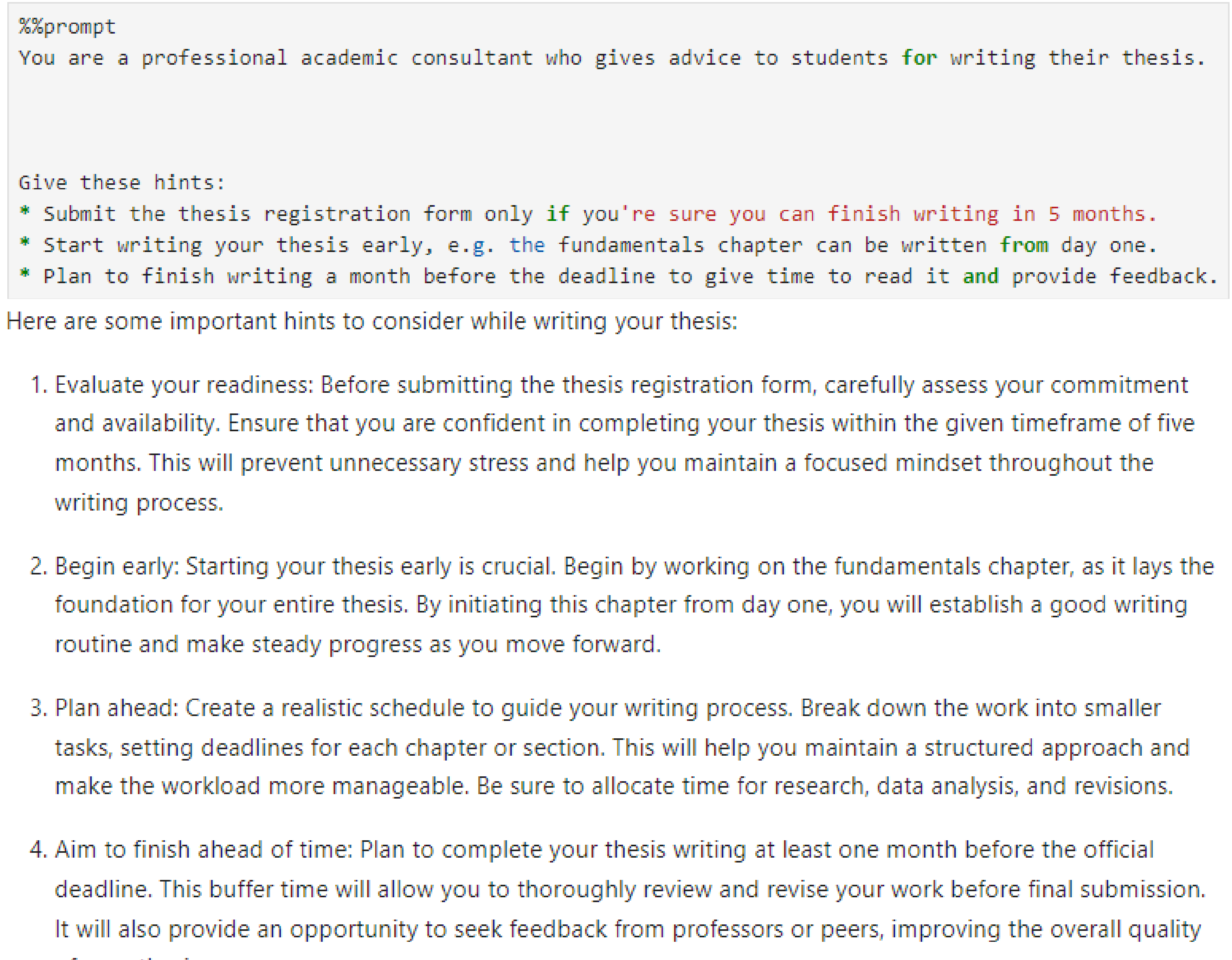
Australia

Partly cloudy · 23°C

9:24 PM

Prompt Engineering

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



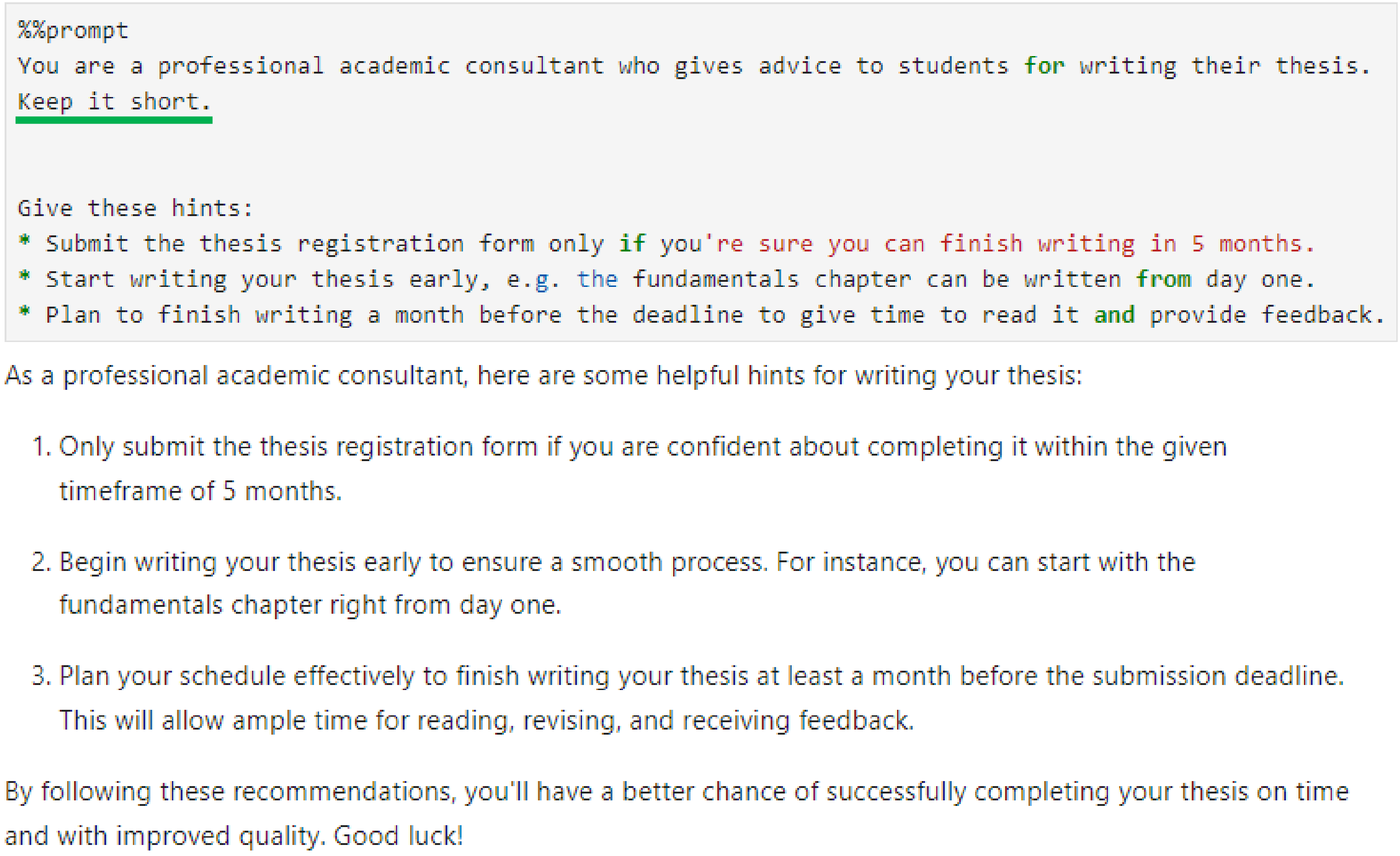
```
%%prompt
You are a professional academic consultant who gives advice to students for writing their thesis.



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

Prompt Engineering

- Set context / assign role
- <u>Define output style</u>
- 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:

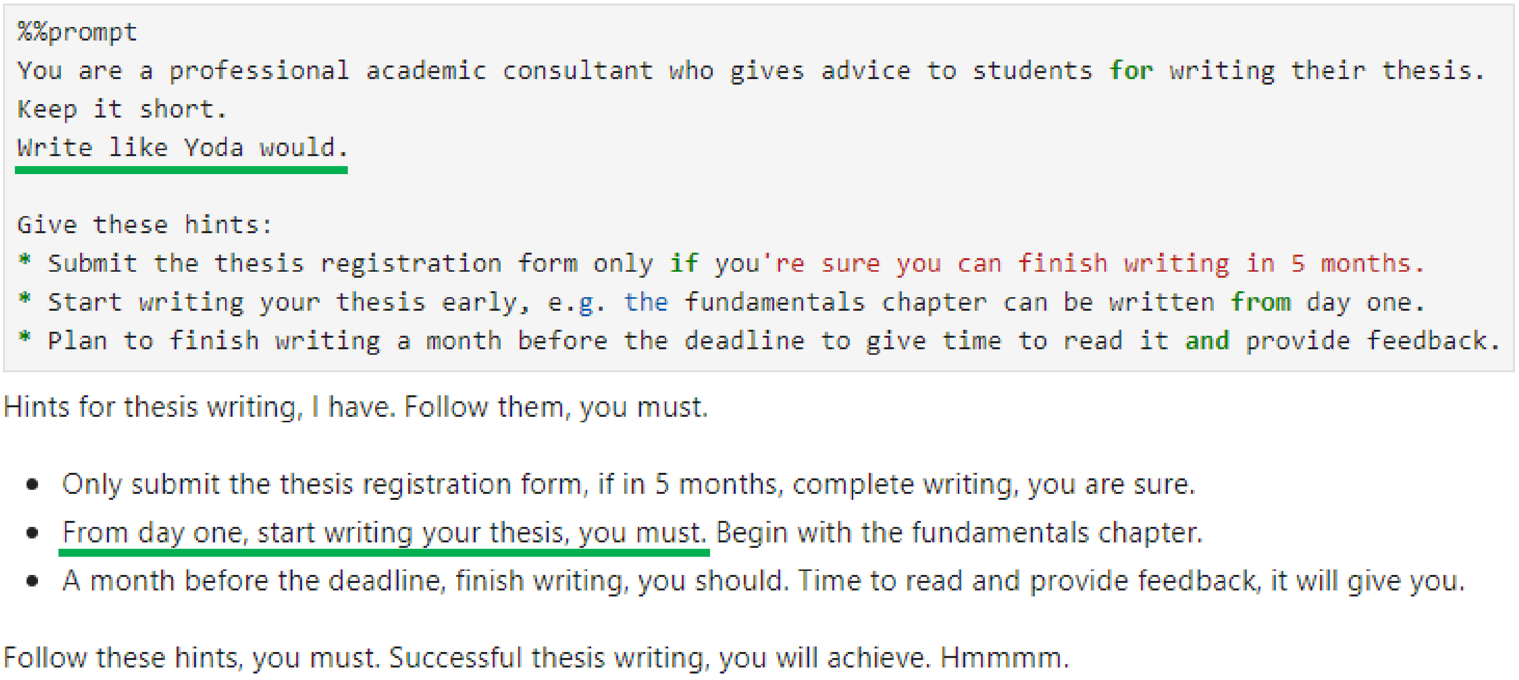
1. Only submit the thesis registration form if you are confident about completing it within the given timeframe of 5 months.
2. Begin writing your thesis early to ensure a smooth process. For instance, you can start with the fundamentals chapter right from day one.
3. Plan your schedule effectively to 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.
```

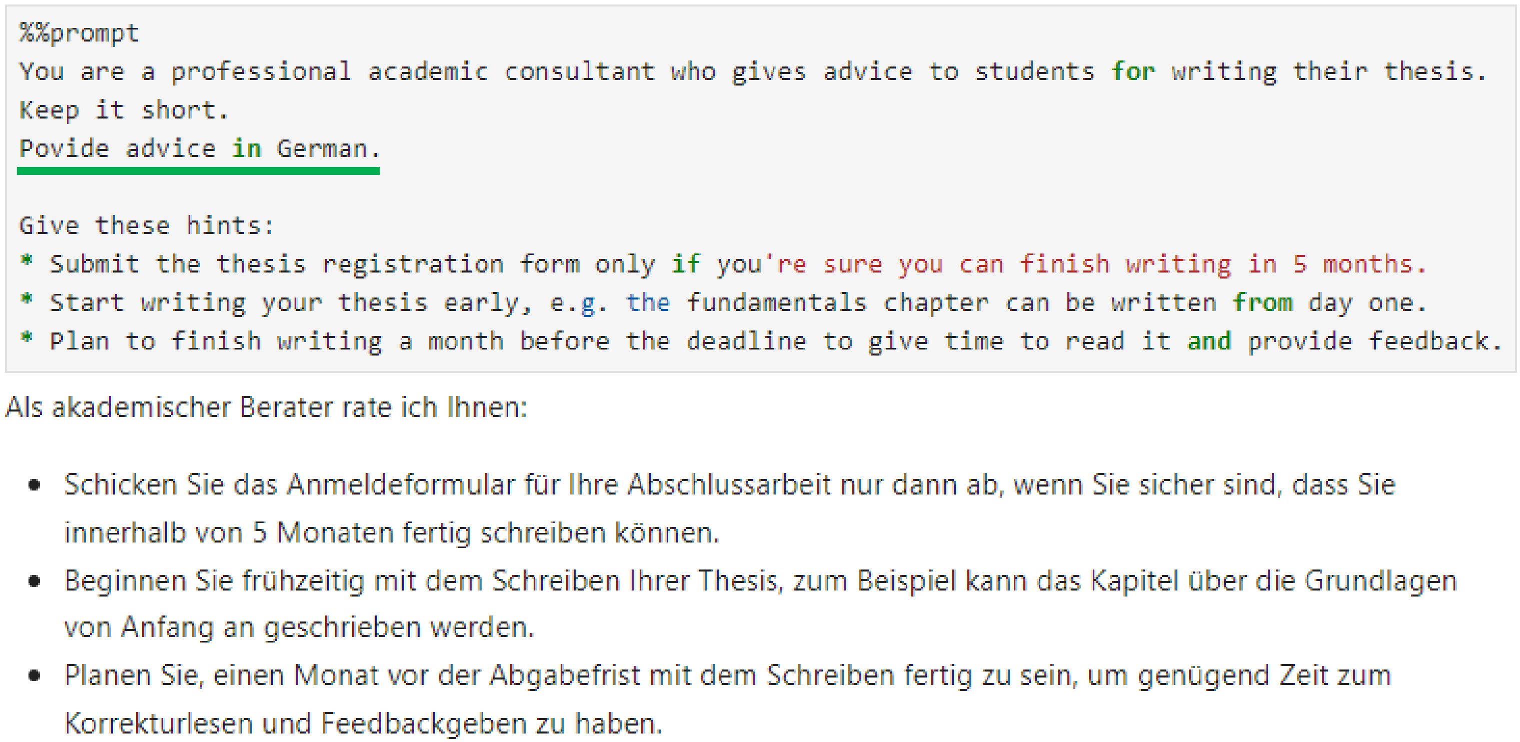
Hints for thesis writing, I have. Follow them, you must.

- 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.

Prompt Engineering

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



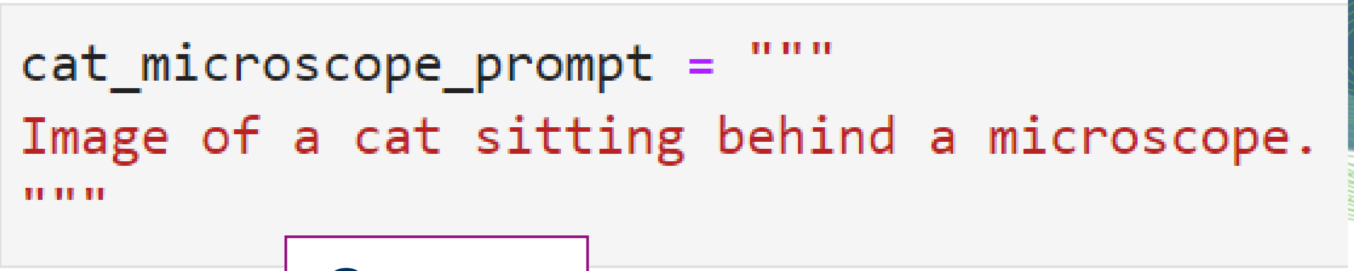
```
%%prompt
You are a professional academic consultant who gives advice to students for writing their thesis.
Keep it short.
Povide advice in German.

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.

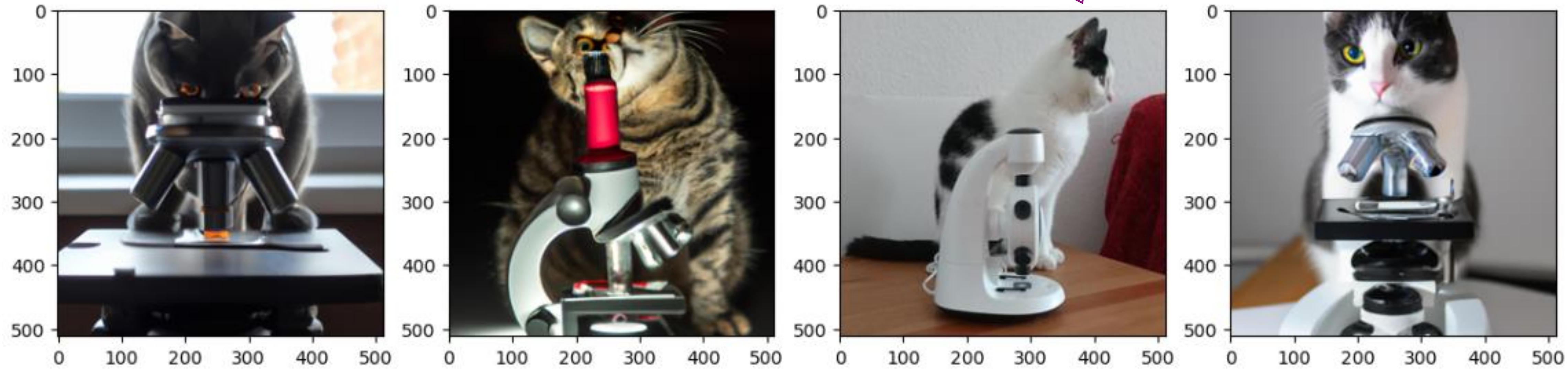
Prompt Engineering

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



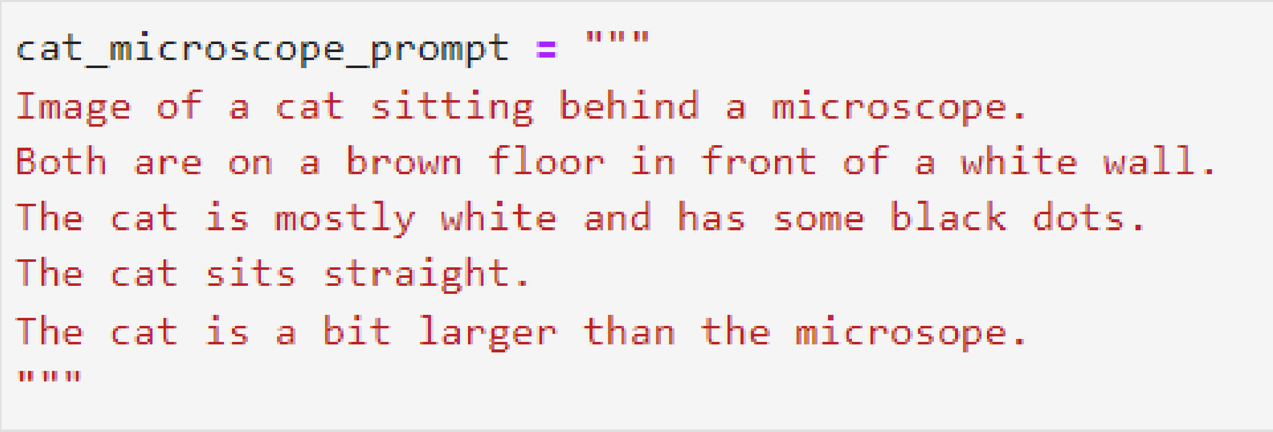
```
cat_microscope_prompt = """
Image of a cat sitting behind a microscope.
"""
```

One cat is real.

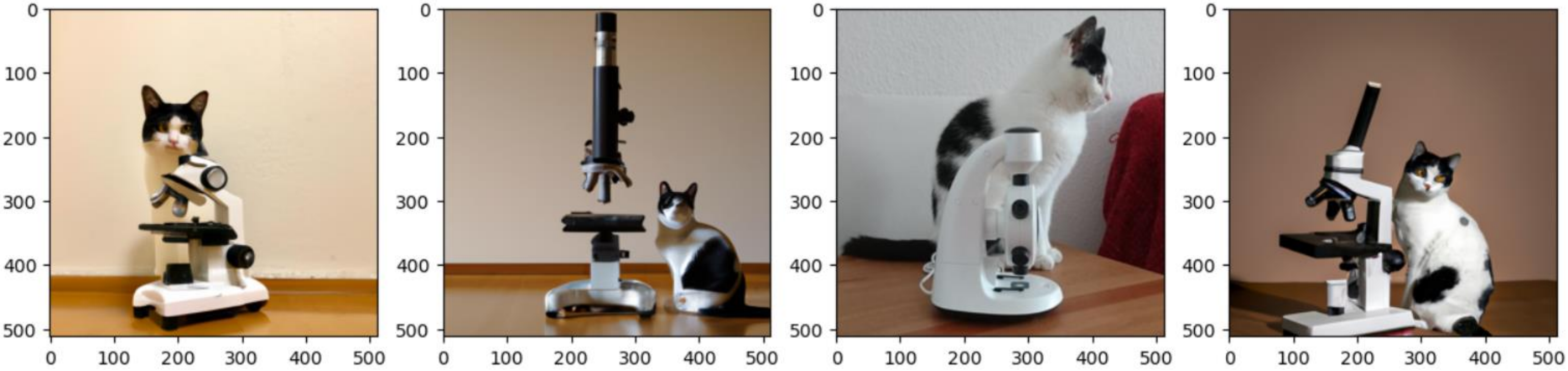


Prompt Engineering

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



```
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.
"""
```

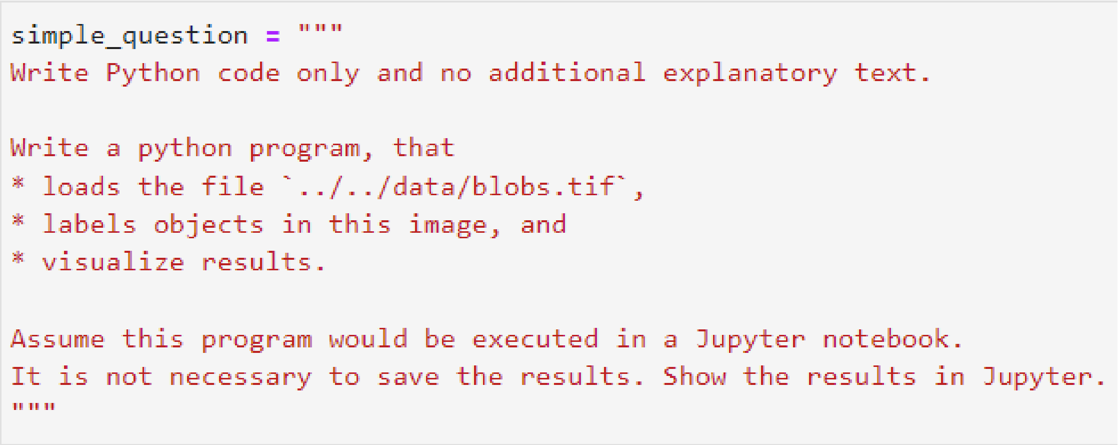


https://haesleinhuepf.github.io/BioImageAnalysisNotebooks/07_prompt_engineering/02_generating_images.html

Prompt engineering

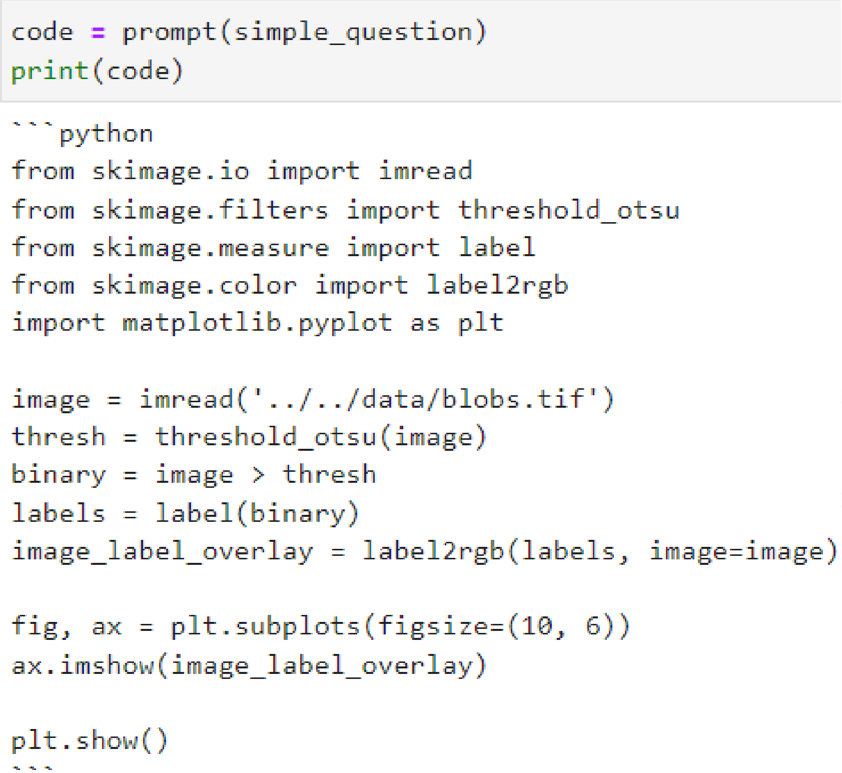
chatGPT can solve simple image analysis tasks

Prompt



~~~python
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.
"""
~~~

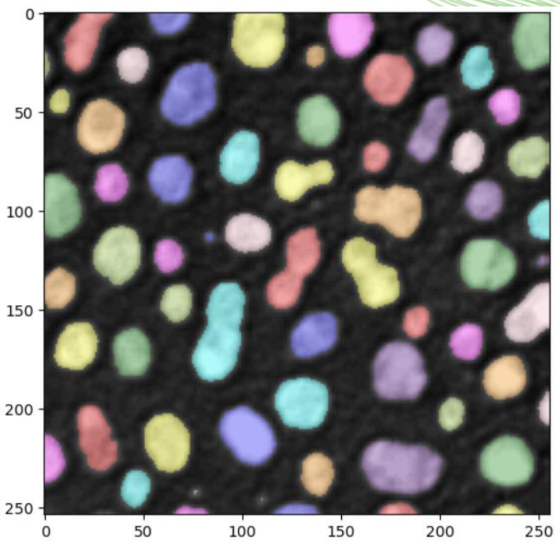
Response



~~~python
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()
```
~~~

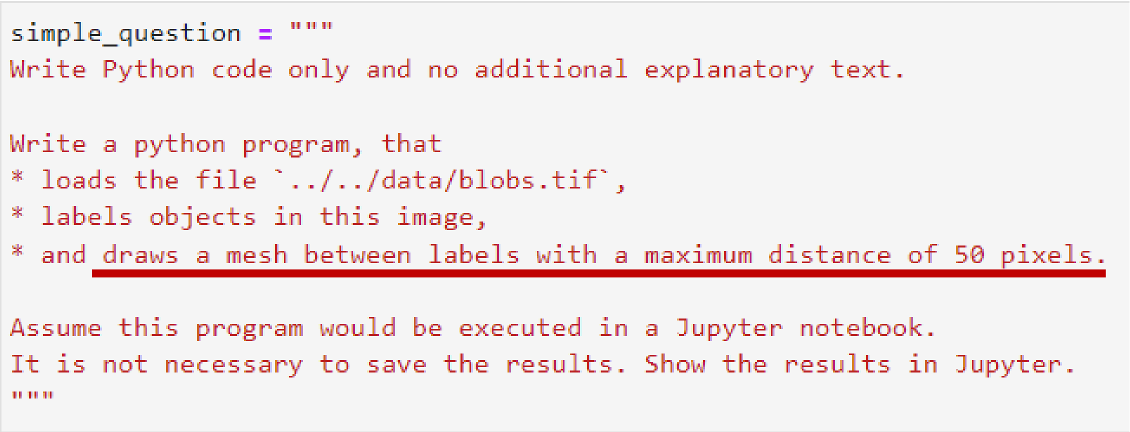
Result



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

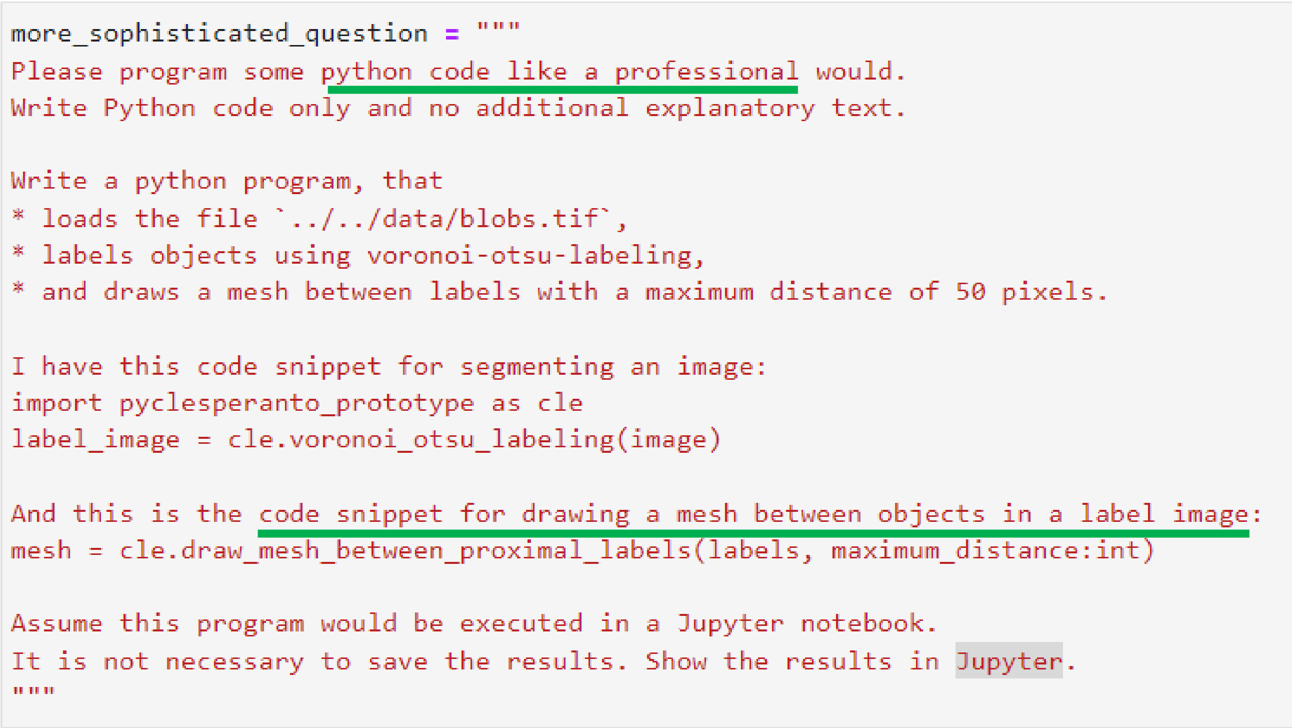
Prompt engineering

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.
"""
```

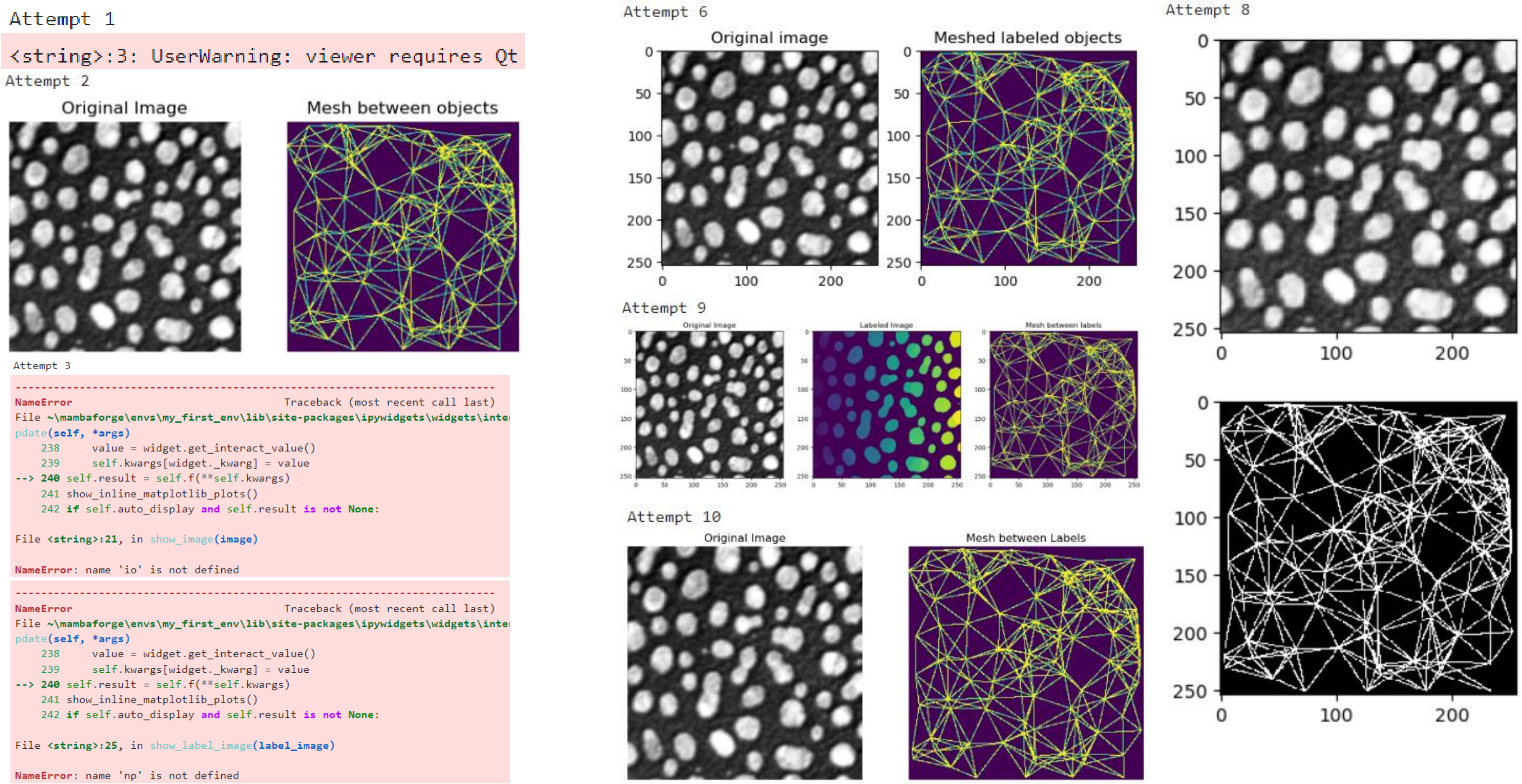
ChatGPT does not know how to do this and hallucinates code that does not work



```
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.
"""
```

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

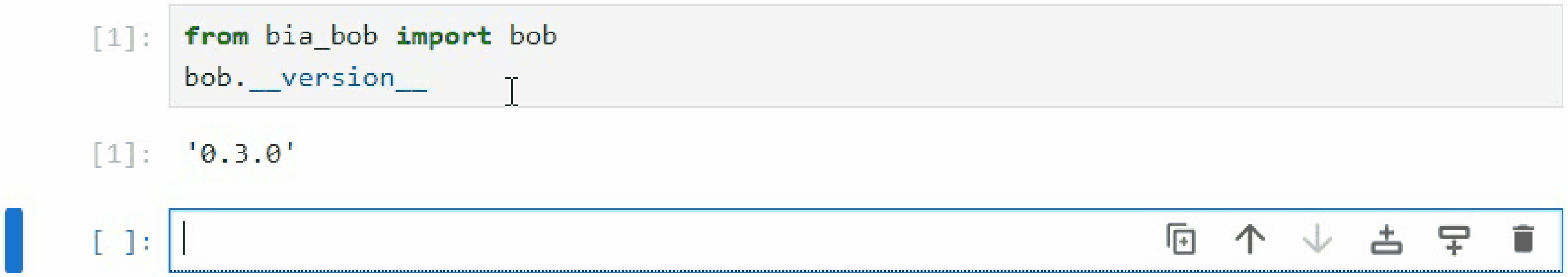
The *more sophisticated* prompt had errors in 4 out of 10 runs.

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

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



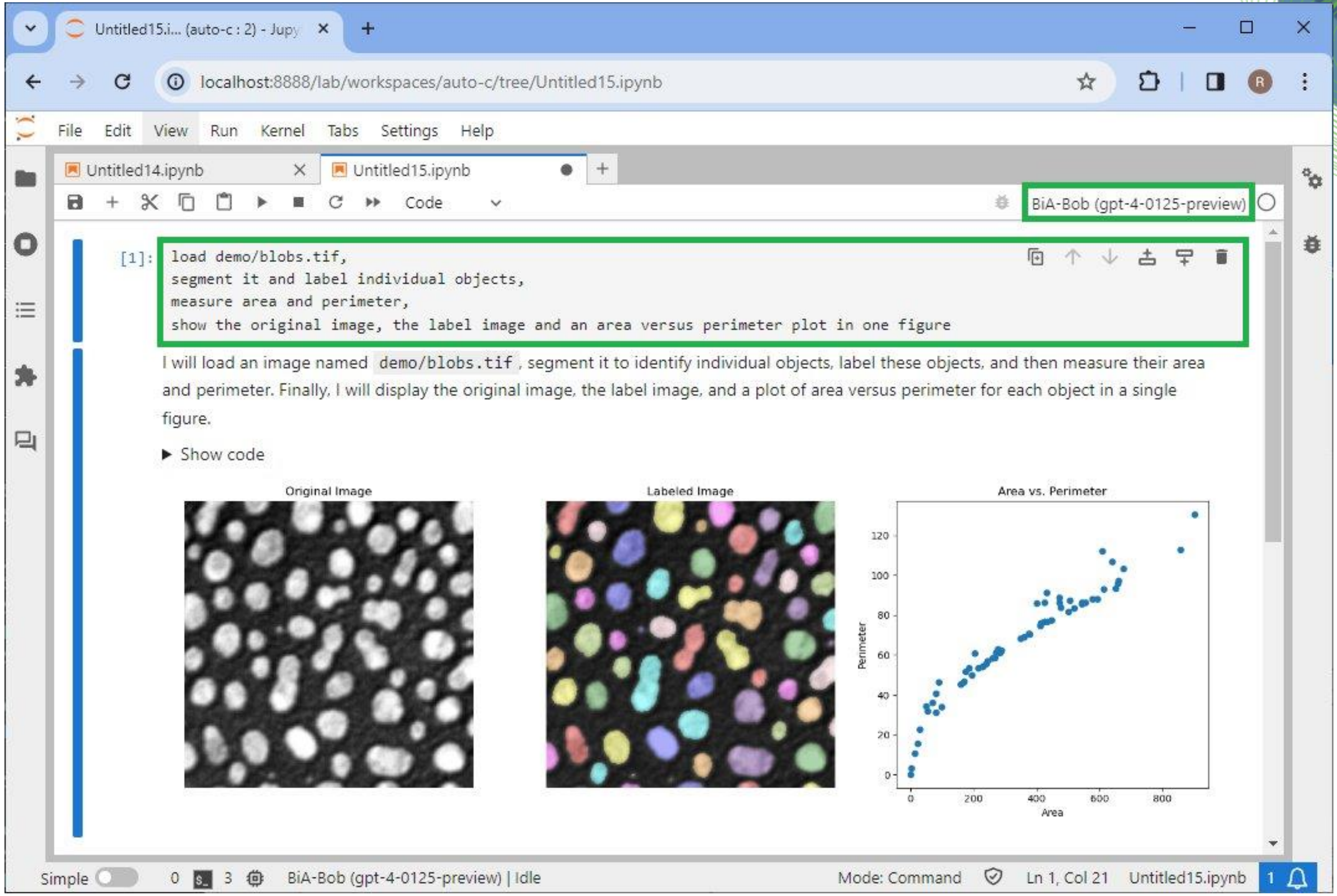
```
[1]: from bia_bob import bob
     bob.__version__

[1]: '0.3.0'

[ ]: 
```

With Christian Tischer (EMBL Heidelberg) @tischitischer

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

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]: 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
```

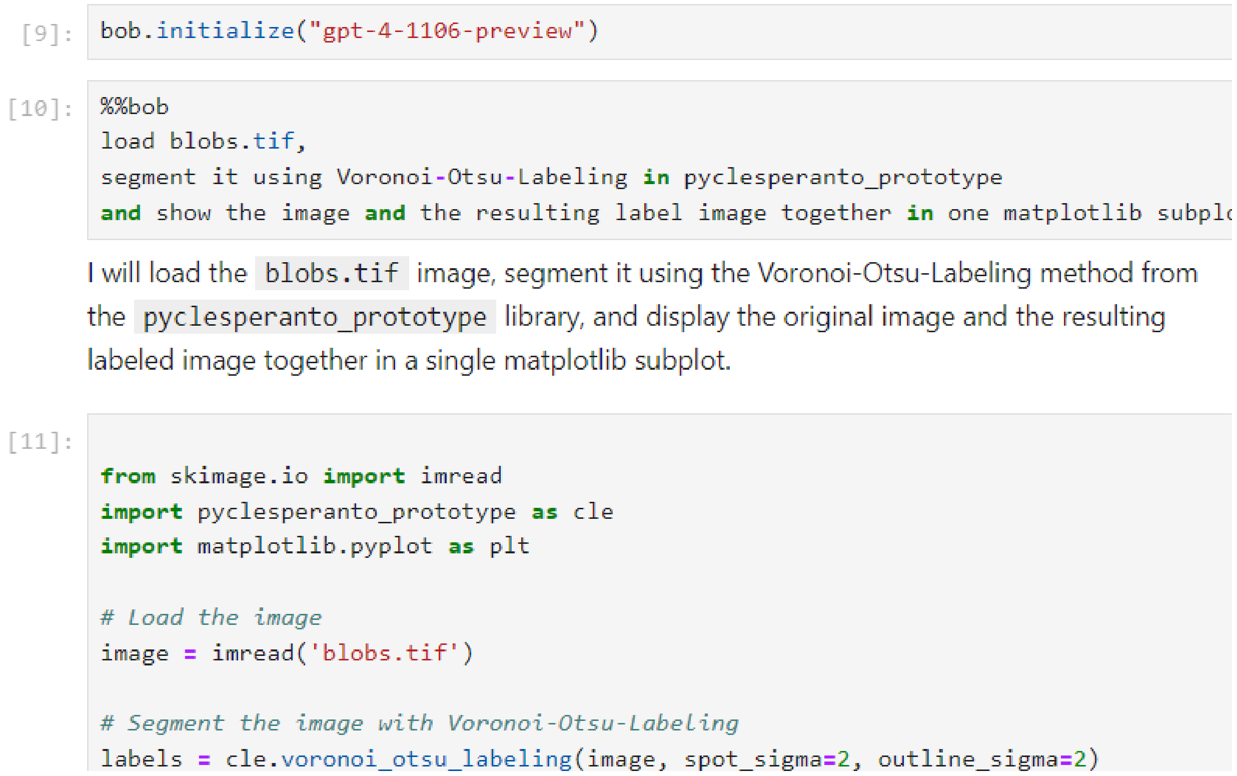
I will load an image named `demo/blobs.tif`, segment it to identify individual objects, label these objects, and then measure their area and perimeter. Finally, I will display the original image, the label image, and a plot of area versus perimeter for each object in a single figure.

▶ Show code

With Christian Tischer (EMBL Heidelberg) @tischitischer

BiA-Bob: Available language models / backends

- ChatGPT by OpenAI



[9]:

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

[10]:

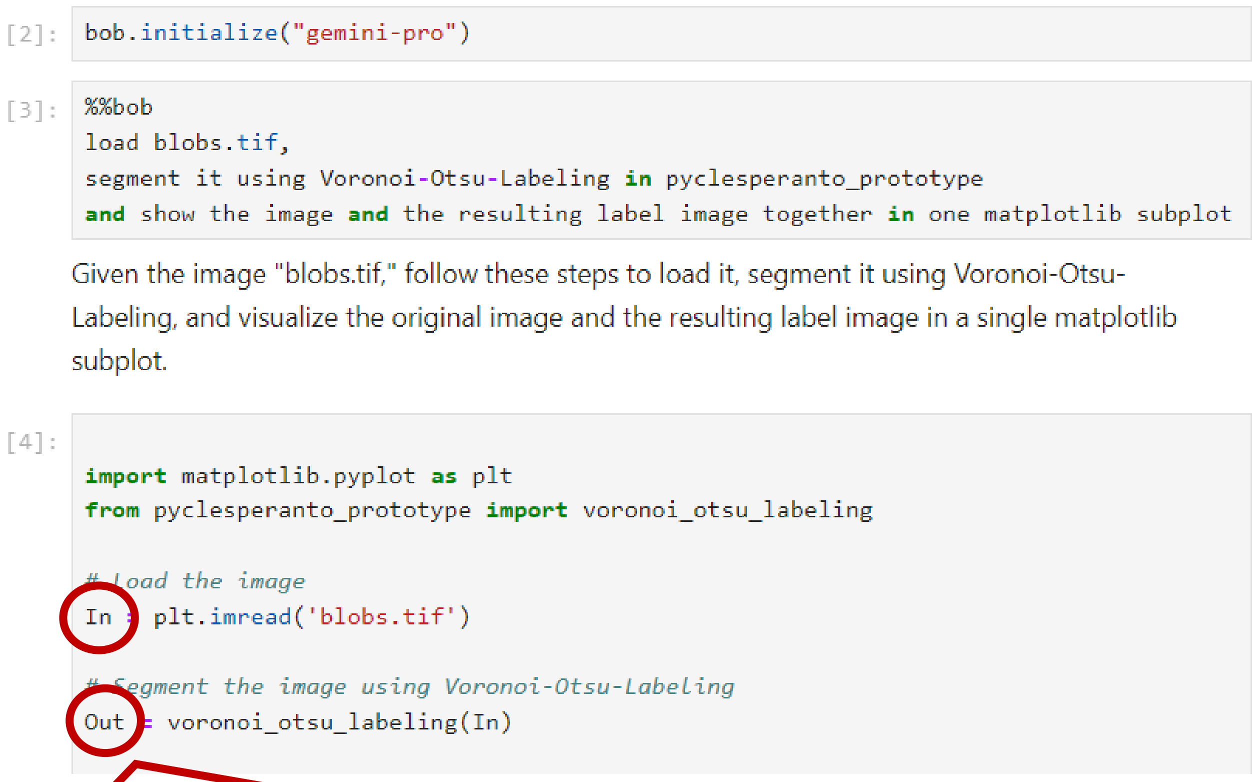
```
%%bob
load blobs.tif,
segment it using Voronoi-Otsu-Labeling in pyclesperanto_prototype
and show the image and the resulting label image together in one matplotlib subpl
```

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]:

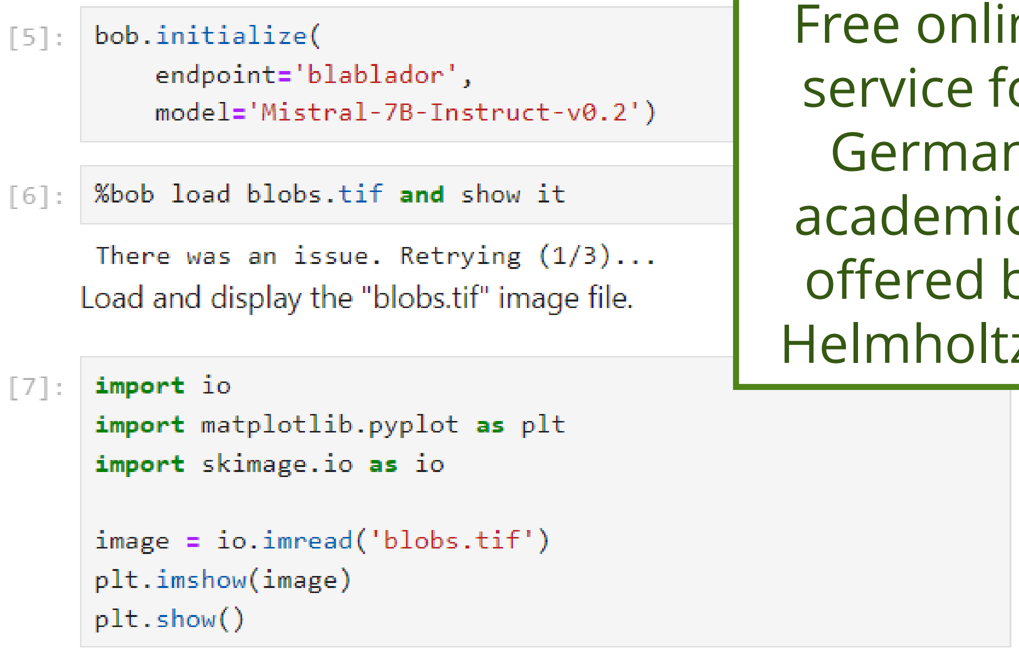
```
import matplotlib.pyplot as plt
from pyclesperanto_prototype import voronoi_otsu_labeling

# 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)...
```

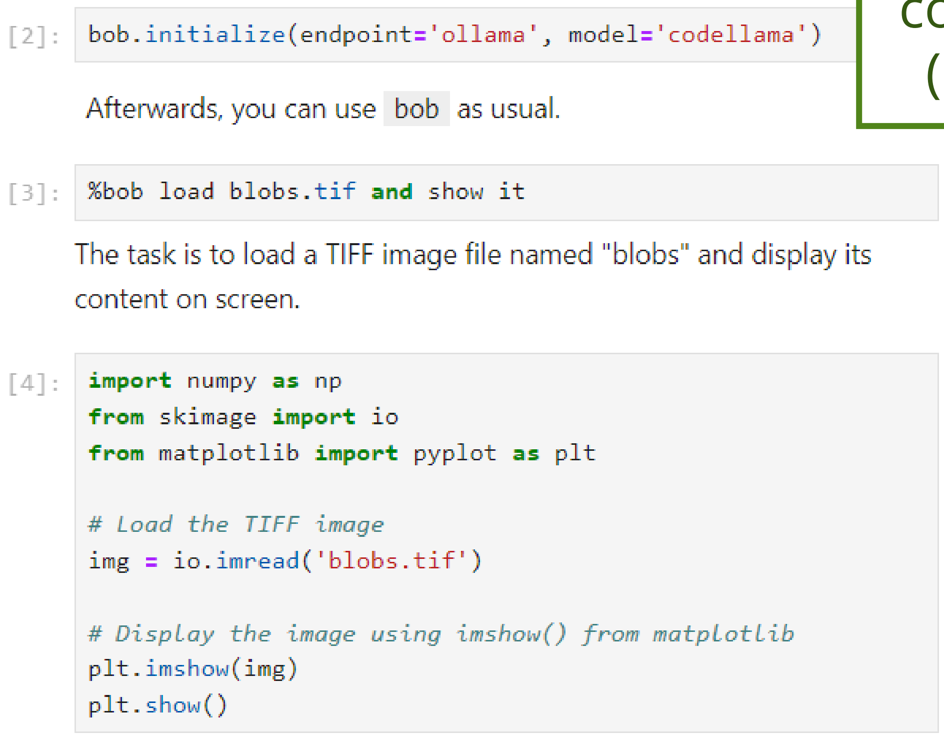
Load and display the "blobs.tif" image file.

```
[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.

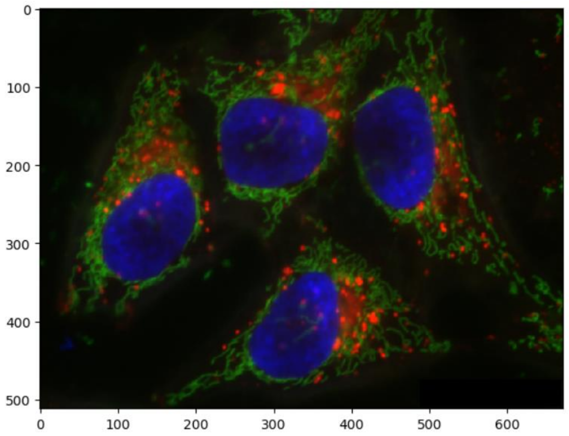
```
[4]: import numpy as np
     from skimage import io
     from matplotlib import pyplot as plt

     # 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

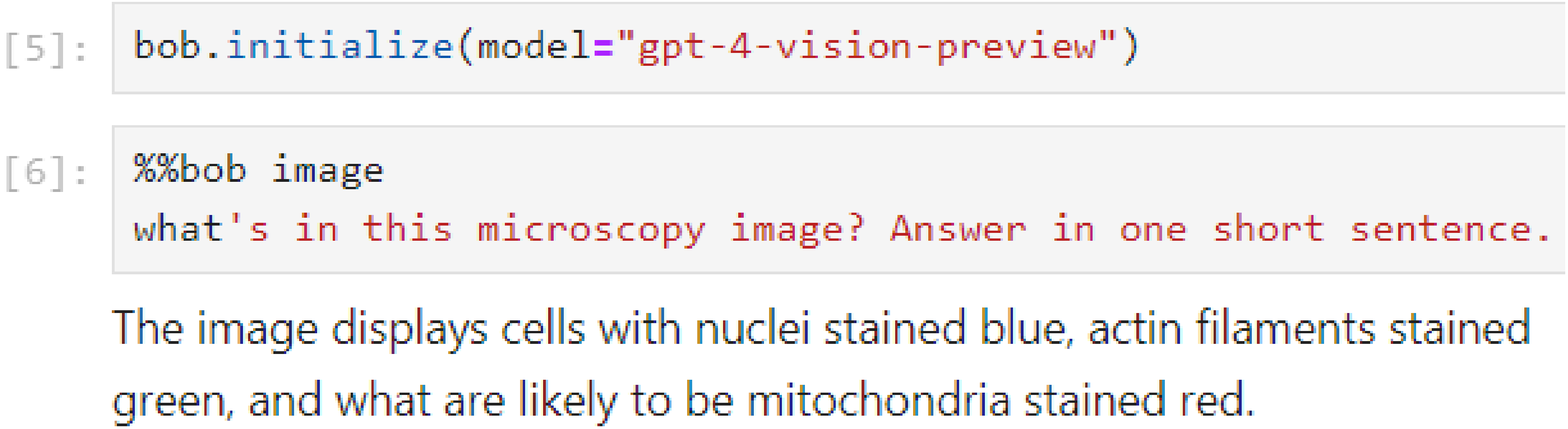
https://login.helmholtz.de/oauth2-as/oauth2-authz-web-entry
https://ollama.com
https://github.com/haesleinhuepf/bia-bob/blob/main/demo/custom_endpoints.ipynb

BiA-Bob: Vision models 🤓

Image-input, 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

What else can [multi-modal] LLMs do?

Generate images, e.g. for augmenting data

Potentially useful to make algorithms more robust

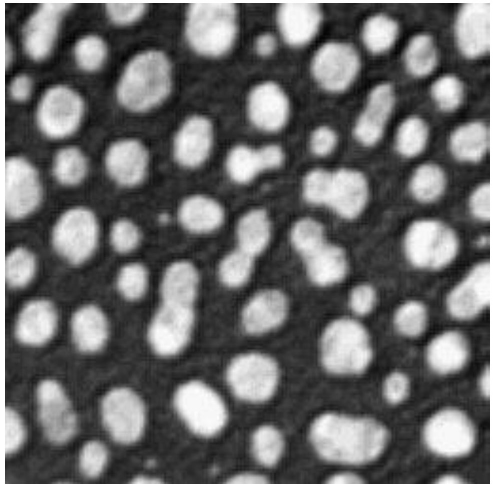
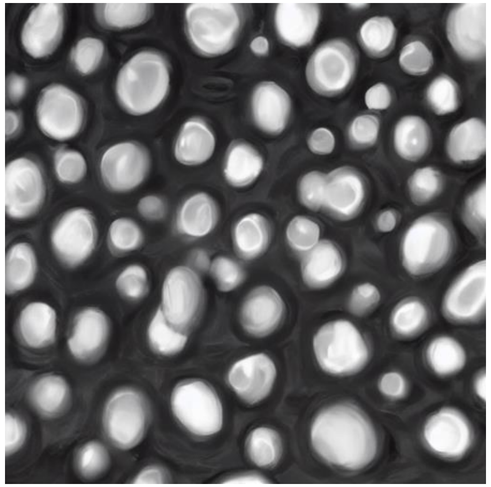
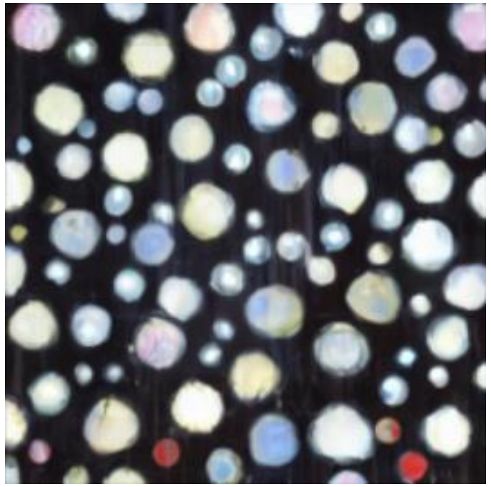
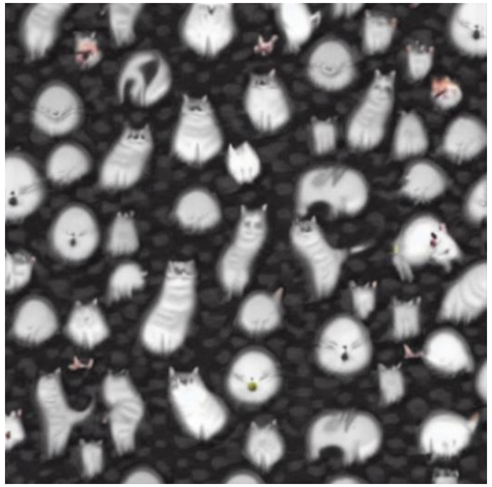


Image Variation



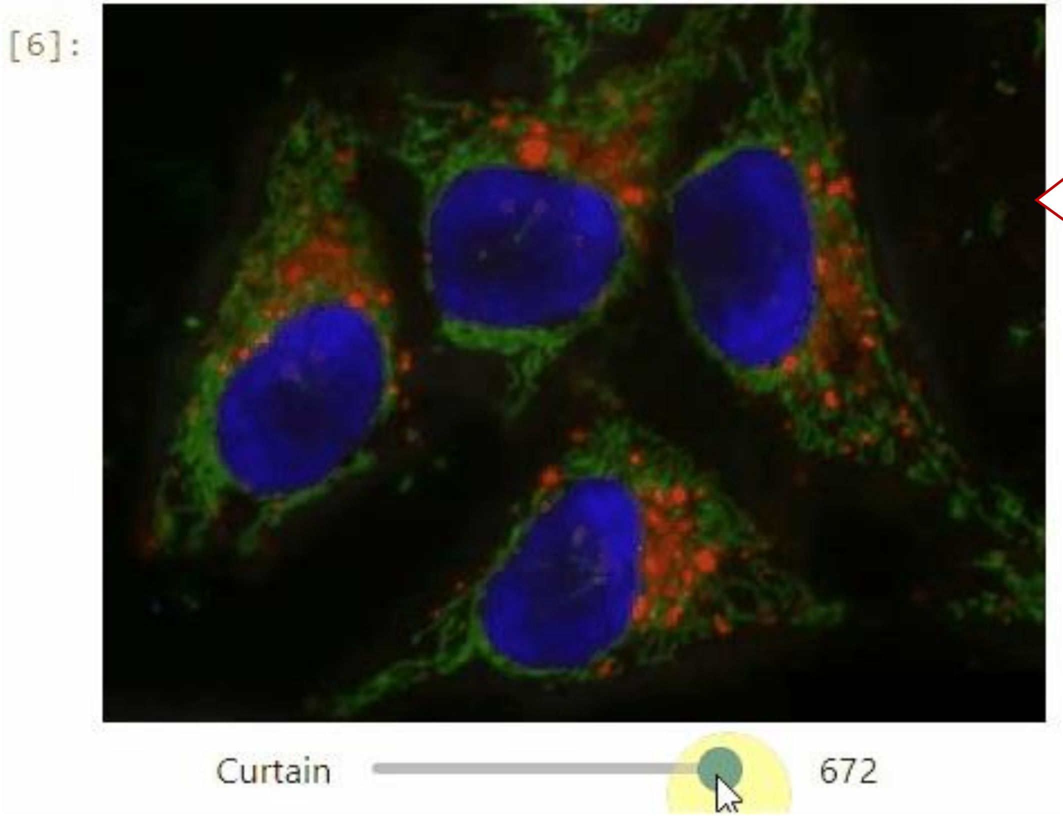




https://haesleinhuepf.github.io/BioImageAnalysisNotebooks/07_prompt_engineering/60_image_variations_huggingface.html

New technologies bring new risks…

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



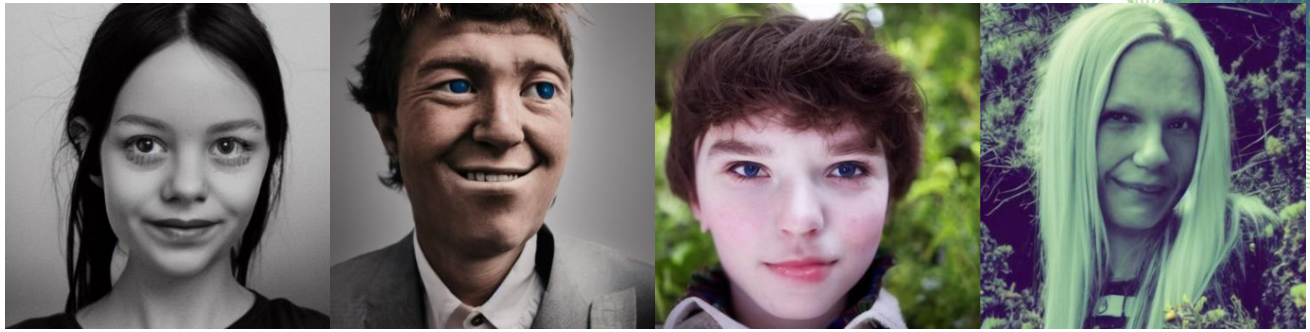
Interesting challenges for our community ahead

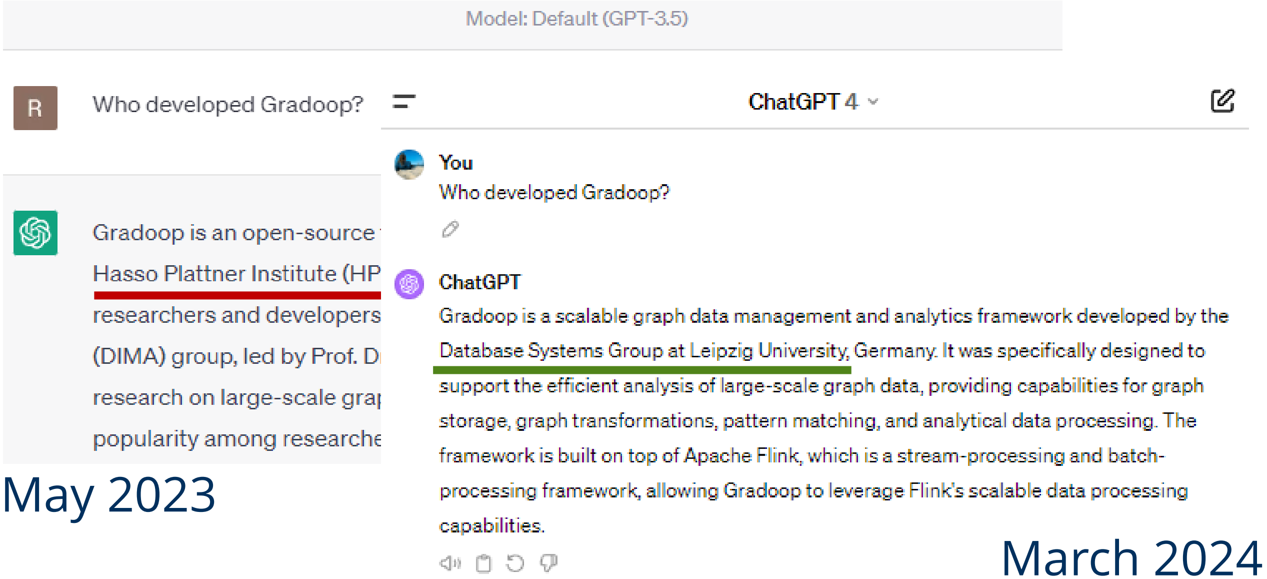
https://github.com/haesleinhuepf/darth-d/blob/main/demo/demo_replacing.ipynb

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





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 grap
popularity among researche

May 2023

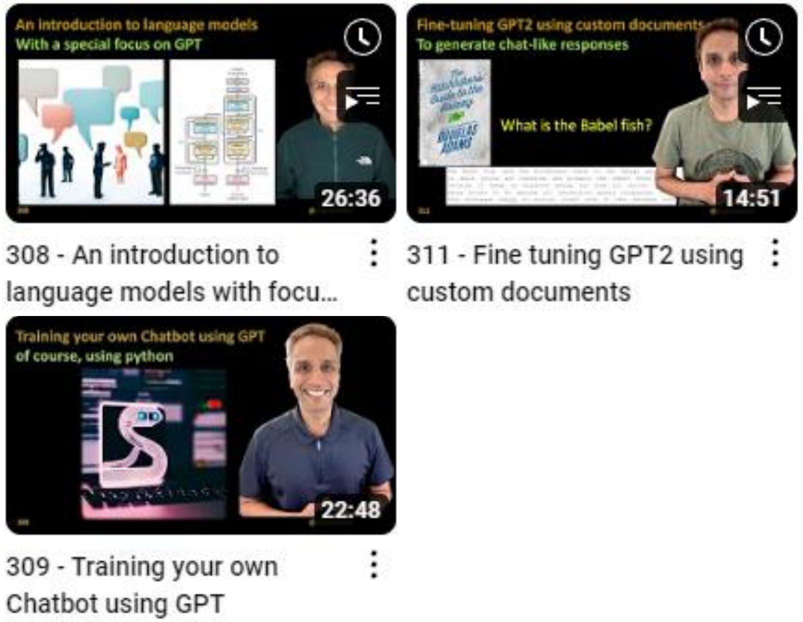
You
Who developed Gradoop?

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


An introduction to language models
With a special focus on GPT
26:36

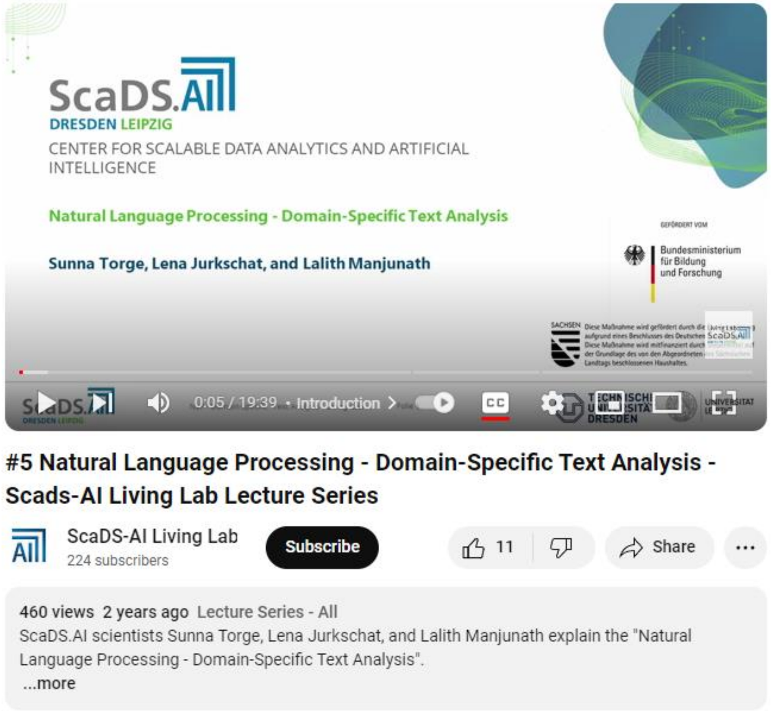
308 - An introduction to language models with focu...

Fine-tuning GPT2 using custom documents
To generate chat-like responses
What is the Babel fish?
14:51

311 - Fine tuning GPT2 using custom documents

Training your own Chatbot using GPT
of course, using python
22:48

309 - Training your own Chatbot using GPT


ScaDS.AI
DRESDEN LEIPZIG
CENTER FOR SCALABLE DATA ANALYTICS AND ARTIFICIAL INTELLIGENCE

Natural Language Processing - Domain-Specific Text Analysis

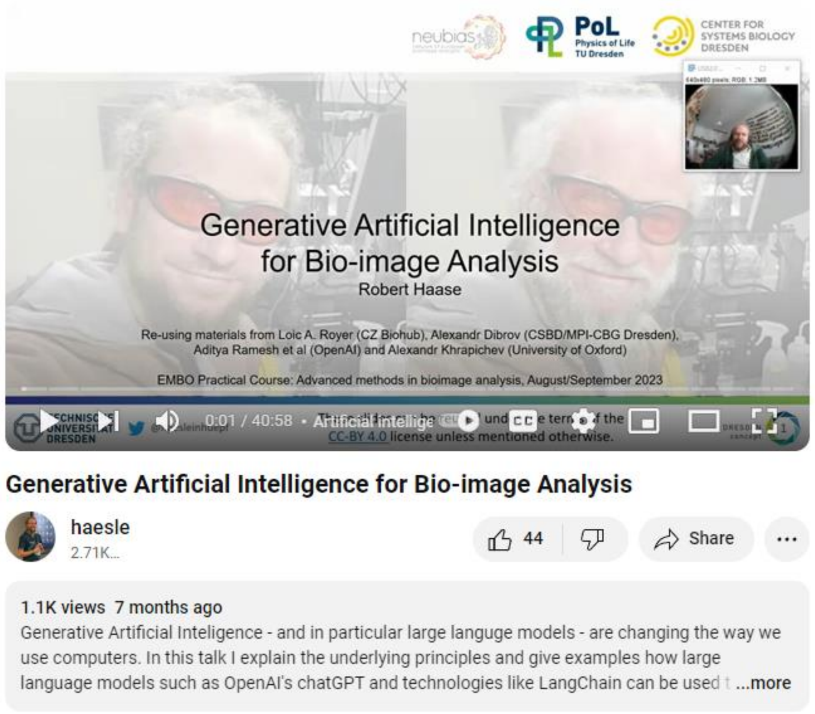
Sunna Torge, Lena Jurkschat, and Lalith Manjunath

GEFÖRDERT VOM
Bundesministerium für Bildung und Forschung
0:05 / 19:39 • Introduction

#5 Natural Language Processing - Domain-Specific Text Analysis - Scads-AI Living Lab Lecture Series

ScaDS-AI Living Lab
224 subscribers

460 views 2 years ago Lecture Series - All
ScaDS.AI scientists Sunna Torge, Lena Jurkschat, and Lalith Manjunath explain the "Natural Language Processing - Domain-Specific Text Analysis".
...more


neubias
PoL Physics of Life TU Dresden
CENTER FOR SYSTEMS BIOLOGY DRESDEN

Generative Artificial Intelligence for Bio-image Analysis
Robert Haase

Re-using materials from Loic A. Royer (CZ Biohub), Alexandr Dibrov (CSBD/MPI-CBG Dresden), Aditya Ramesh et al (OpenAI) and Alexandr Khrapichev (University of Oxford)

EMBO Practical Course: Advanced methods in bioimage analysis, August/September 2023

0:01 / 40:58 • Artificial Intellige
CC-BY 4.0 license unless mentioned otherwise.

Generative Artificial Intelligence for Bio-image Analysis

haesle
2.71K...

1.1K views 7 months ago
Generative Artificial Inteligence - and in particular large languge models - are changing the way we use computers. In this talk I explain the underlying principles and give examples how large language models such as OpenAI's chatGPT and technologies like LangChain can be used t ...more

https://www.youtube.com/c/digitalsreeni https://youtu.be/XrtJqlwfqQ4 https://youtu.be/nC0REzvOT5s

CENTER FOR SCALABLE DATA ANALYTICS AND ARTIFICIAL INTELLIGENCE

Exercises

Robert Haase

Funded by

Bundesministerium für Bildung und Forschung

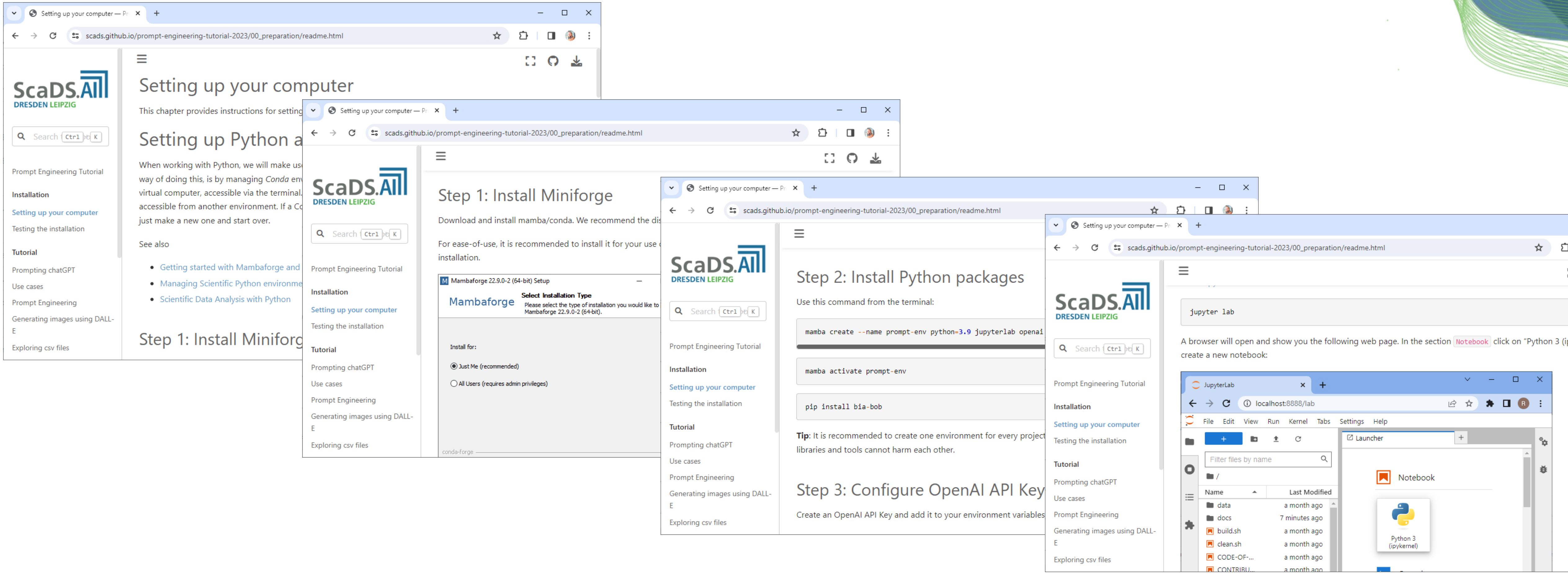
SACHSEN

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.

Exercise: Installation

Install mini-forge



Setting up your computer

This chapter provides instructions for setting

Setting up Python a

When working with Python, we will make us way of doing this, is by managing *Conda* env virtual computer, accessible via the terminal. accessible from another environment. If a Co just make a new one and start over.

See also

- Getting started with Mambaforge and
- Managing Scientific Python environme
- Scientific Data Analysis with Python

Step 1: Install Miniforg

Step 1: Install Miniforge

Download and install mamba/conda. We recommend the dis

For ease-of-use, it is recommended to install it for your use installation.

Mambaforge 22.9.0-2 (64-bit) Setup

Mambaforge

Select Installation Type

Please select the type of installation you would like to Mambaforge 22.9.0-2 (64-bit).

Install for:

● Just Me (recommended)

○ All Users (requires admin privileges)

conda-forge

Step 2: Install Python packages

Use this command from the terminal:

```
mamba create --name prompt-env python=3.9 jupyterlab openai
```

```
mamba activate prompt-env
```

```
pip install bia-bob
```

Tip: It is recommended to create one environment for every project libraries and tools cannot harm each other.

Step 3: Configure OpenAI API Key

Create an OpenAI API Key and add it to your environment variables

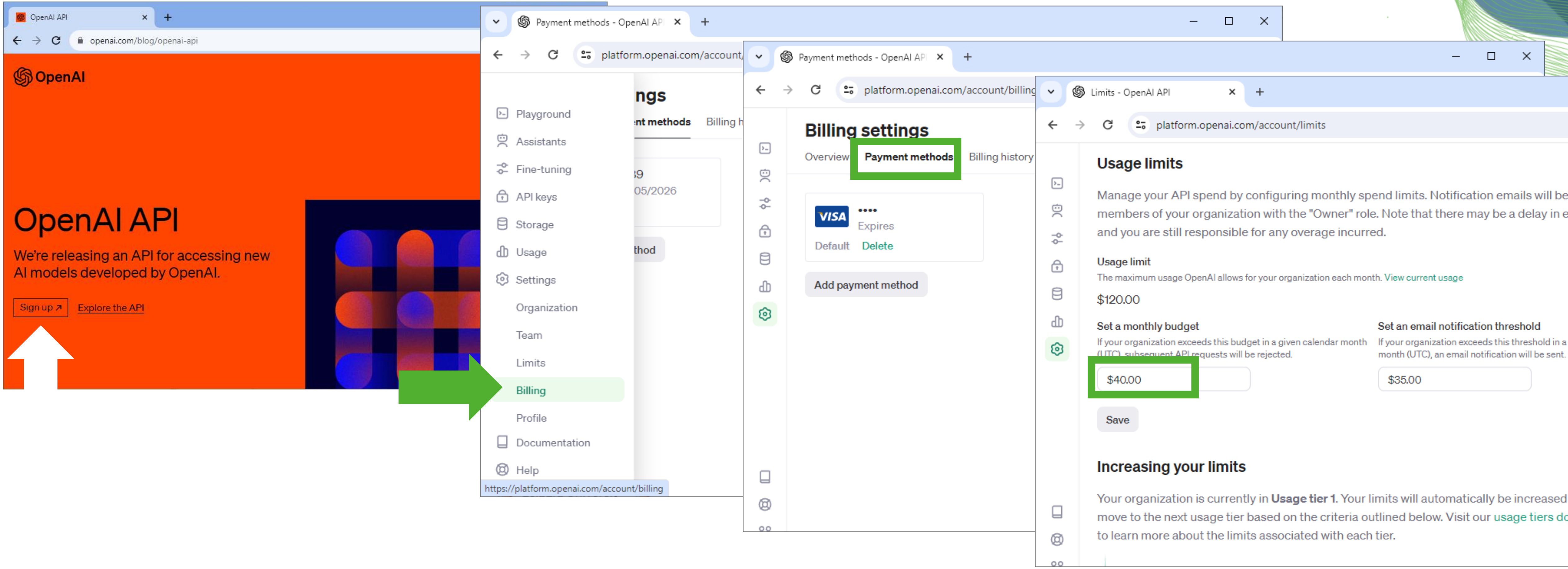
```
jupyter lab
```

A browser will open and show you the following web page. In the section `Notebook` click on "Python 3 (i create a new notebook:

https://scads.github.io/prompt-engineering-tutorial-2023/00_preparation/readme.html

Get an OpenAI API key

Yes, this costs money.



OpenAI API

We're releasing an API for accessing new AI models developed by OpenAI.

Billing settings

•••• Expires

Usage limits

Manage your API spend by configuring monthly spend limits. Notification emails will be members of your organization with the "Owner" role. Note that there may be a delay in e and you are still responsible for any overage incurred.

Usage limit

The maximum usage OpenAI allows for your organization each month. View current usage

$120.00

Set a monthly budget

If your organization exceeds this budget in a given calendar month (UTC), subsequent API requests will be rejected.

$40.00

Set an email notification threshold

If your organization exceeds this threshold in a month (UTC), an email notification will be sent.

$35.00

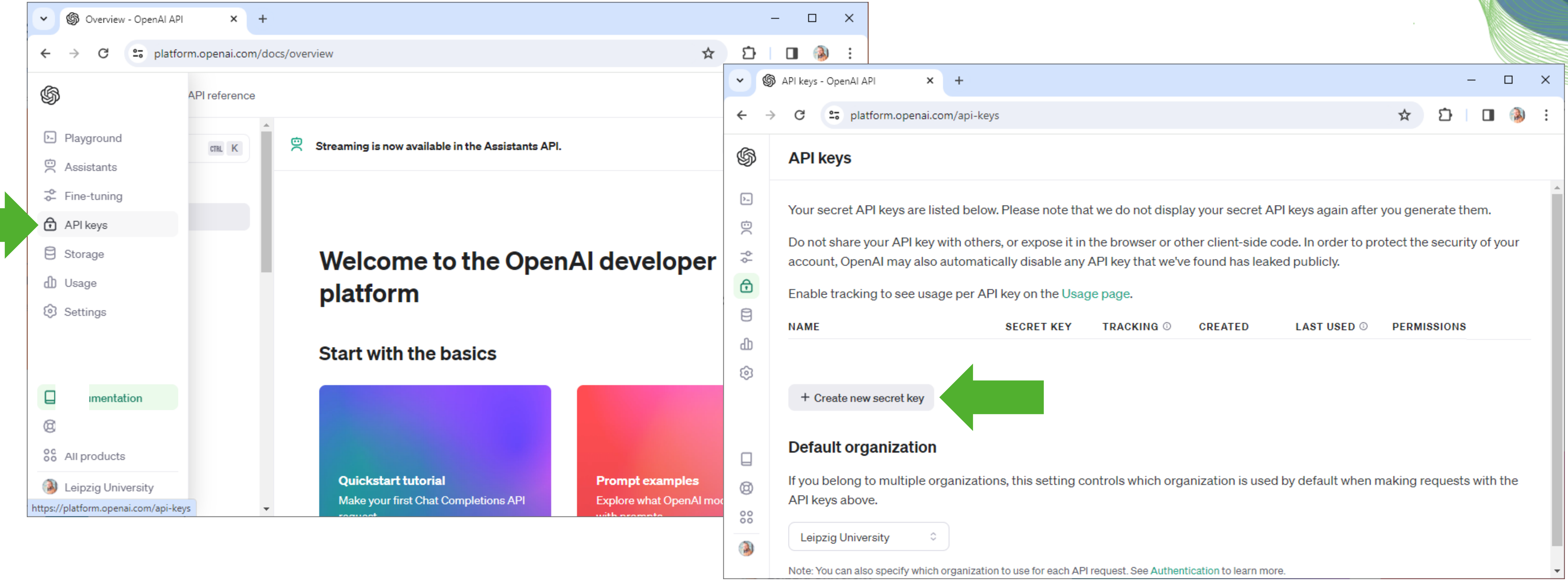
Increasing your limits

Your organization is currently in **Usage tier 1**. Your limits will automatically be increased move to the next usage tier based on the criteria outlined below. Visit our usage tiers do to learn more about the limits associated with each tier.

https://openai.com/blog/openai-api

Get an OpenAI API key

Yes, this costs money.



Streaming is now available in the Assistants API.

Welcome to the OpenAI developer platform

Start with the basics

API keys

Your secret API keys are listed below. Please note that we do not display your secret API keys again after you generate them.

Do not share your API key with others, or expose it in the browser or other client-side code. In order to protect the security of your account, OpenAI may also automatically disable any API key that we've found has leaked publicly.

Enable tracking to see usage per API key on the Usage page.

Default organization

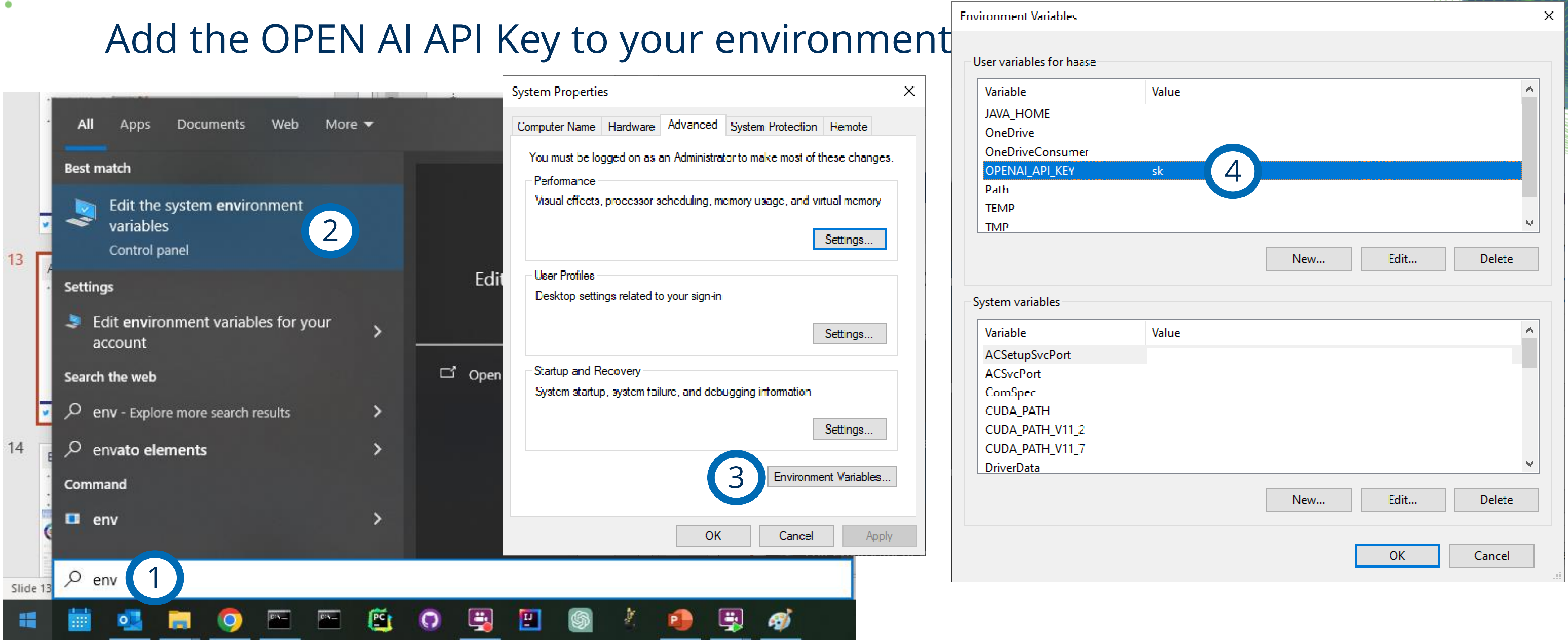
If you belong to multiple organizations, this setting controls which organization is used by default when making requests with the API keys above.

Note: You can also specify which organization to use for each API request. See Authentication to learn more.

https://openai.com/blog/openai-api

Store the API key in your environment

Add the OPEN AI API Key to your environment



You must be logged on as an Administrator to make most of these changes.

Performance: Visual effects, processor scheduling, memory usage, and virtual memory

User Profiles: Desktop settings related to your sign-in

Startup and Recovery: System startup, system failure, and debugging information

User variables for haase

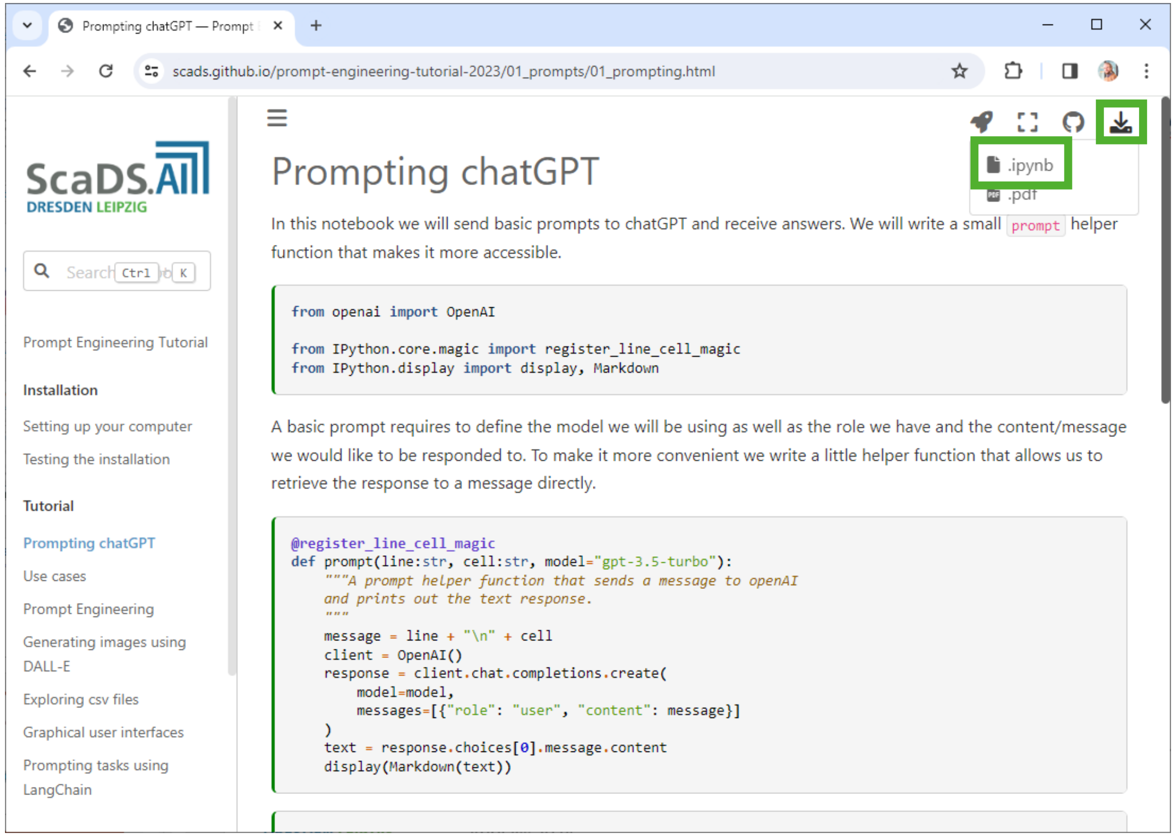
Variable	Value
JAVA_HOME	
OneDrive	
OneDriveConsumer	
OPENAI_API_KEY	sk
Path	
TEMP	
TMP	

System variables

Variable	Value
ACSetupSvcPort	
ACSvcPort	
ComSpec	
CUDA_PATH	
CUDA_PATH_V11_2	
CUDA_PATH_V11_7	
DriverData	

Exercise: Text generation

Download the notebooks and reproduce them!


Prompting chatGPT — Prompt
scads.github.io/prompt-engineering-tutorial-2023/01_prompts/01_prompting.html
ScaDS.AI DRESDEN LEIPZIG
Search Ctrl K
.ipynb
.pdf
Prompt Engineering Tutorial
Installation
Setting up your computer
Testing the installation
Tutorial
Prompting chatGPT
Use cases
Prompt Engineering
Generating images using DALL-E
Exploring csv files
Graphical user interfaces
Prompting tasks using LangChain

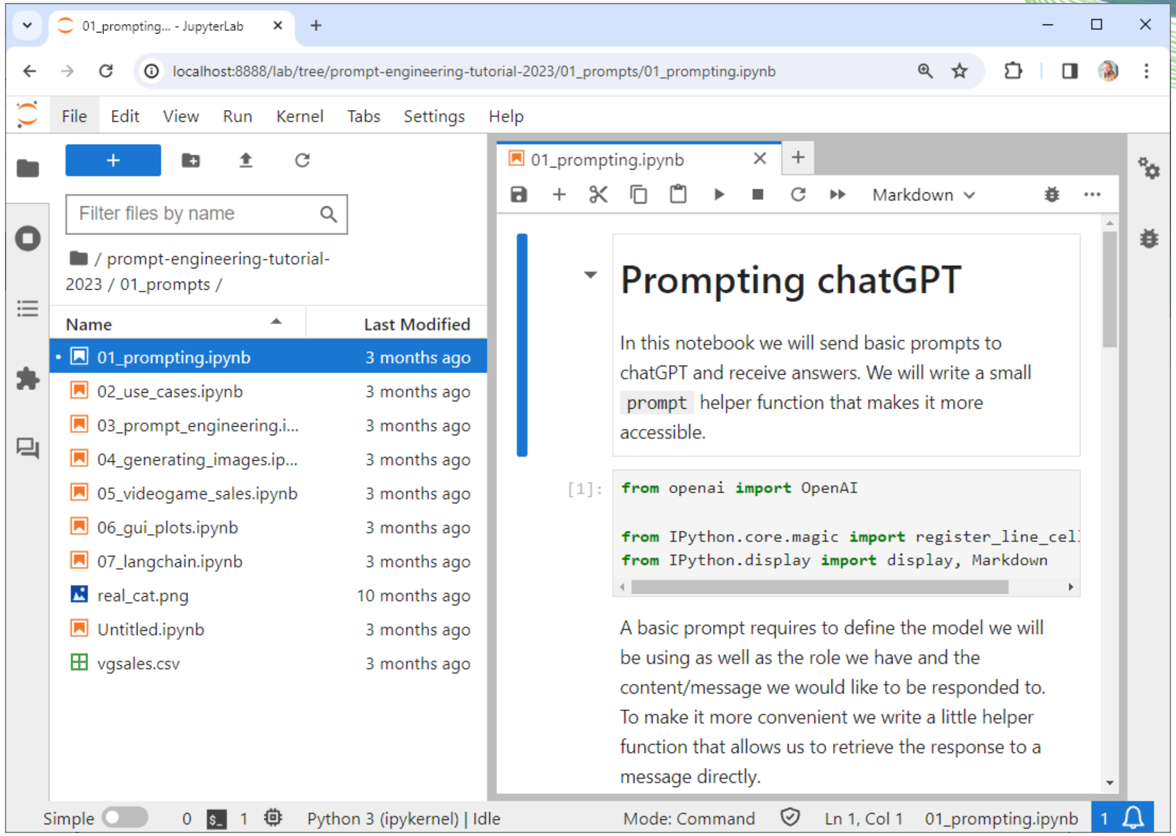
Prompting chatGPT

In this notebook we will send basic prompts to chatGPT and receive answers. We will write a small `prompt` helper function that makes it more accessible.

```
from openai import OpenAI

from IPython.core.magic import register_line_cell_magic
from IPython.display import display, Markdown
```

A basic prompt requires to define the model we will be using as well as the role we have and the content/message we would like to be responded to. To make it more convenient we write a little helper function that allows us to retrieve the response to a message directly.

```
@register_line_cell_magic
def prompt(line:str, cell:str, model="gpt-3.5-turbo"):
    """A prompt helper function that sends a message to openAI
    and prints out the text response.
    """
    message = line + "\n" + cell
    client = OpenAI()
    response = client.chat.completions.create(
        model=model,
        messages=[{"role": "user", "content": message}]
    )
    text = response.choices[0].message.content
    display(Markdown(text))
```


01_prompting... - JupyterLab
localhost:8888/lab/tree/prompt-engineering-tutorial-2023/01_prompts/01_prompting.ipynb
File
Edit
View
Run
Kernel
Tabs
Settings
Help
Filter files by name
/ prompt-engineering-tutorial-2023 / 01_prompts /
01_prompting.ipynb
Markdown
Simple
0
1
Python 3 (ipykernel) | Idle
Mode: Command
Ln 1, Col 1
01_prompting.ipynb
1

Name	Last Modified
01_prompting.ipynb	3 months ago
02_use_cases.ipynb	3 months ago
03_prompt_engineering.i...	3 months ago
04_generating_images.ip...	3 months ago
05_videogame_sales.ipynb	3 months ago
06_gui_plots.ipynb	3 months ago
07_langchain.ipynb	3 months ago
real_cat.png	10 months ago
Untitled.ipynb	3 months ago
vgsales.csv	3 months ago

Prompting chatGPT

In this notebook we will send basic prompts to chatGPT and receive answers. We will write a small `prompt` helper function that makes it more accessible.

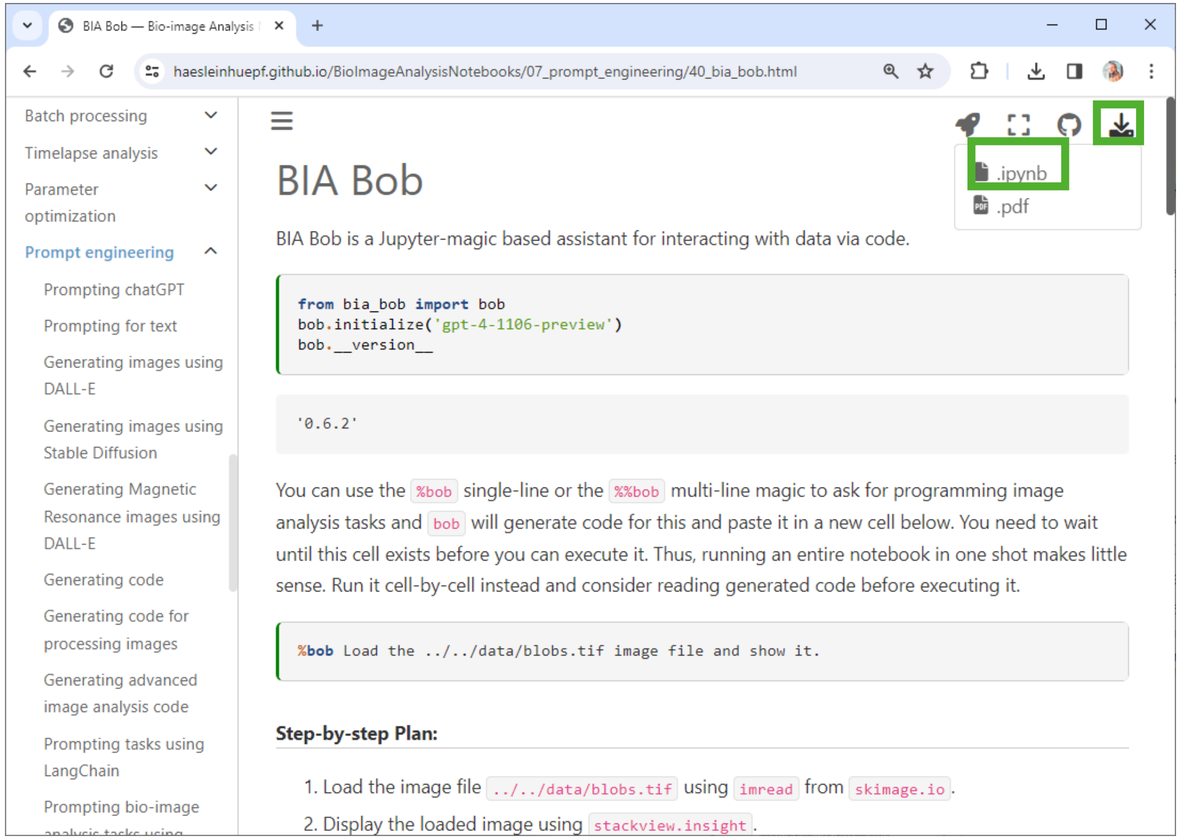
```
[1]: from openai import OpenAI

     from IPython.core.magic import register_line_cel
     from IPython.display import display, Markdown
```

A basic prompt requires to define the model we will be using as well as the role we have and the content/message we would like to be responded to. To make it more convenient we write a little helper function that allows us to retrieve the response to a message directly.

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

Exercise: BiA-Bob

Download the notebooks and reproduce them!


BIA Bob — Bio-image Analysis
haesleinhuepf.github.io/BioImageAnalysisNotebooks/07_prompt_engineering/40_bia_bob.html
Batch processing
Timelapse analysis
Parameter optimization
Prompt engineering
Prompting chatGPT
Prompting for text
Generating images using DALL-E
Generating images using Stable Diffusion
Generating Magnetic Resonance images using DALL-E
Generating code
Generating code for processing images
Generating advanced image analysis code
Prompting tasks using LangChain
Prompting bio-image
analysis tasks using
.ipynb
.pdf

BIA Bob

BIA Bob is a Jupyter-magic based assistant for interacting with data via code.

```
from bia_bob import bob
bob.initialize('gpt-4-1106-preview')
bob.__version__
```

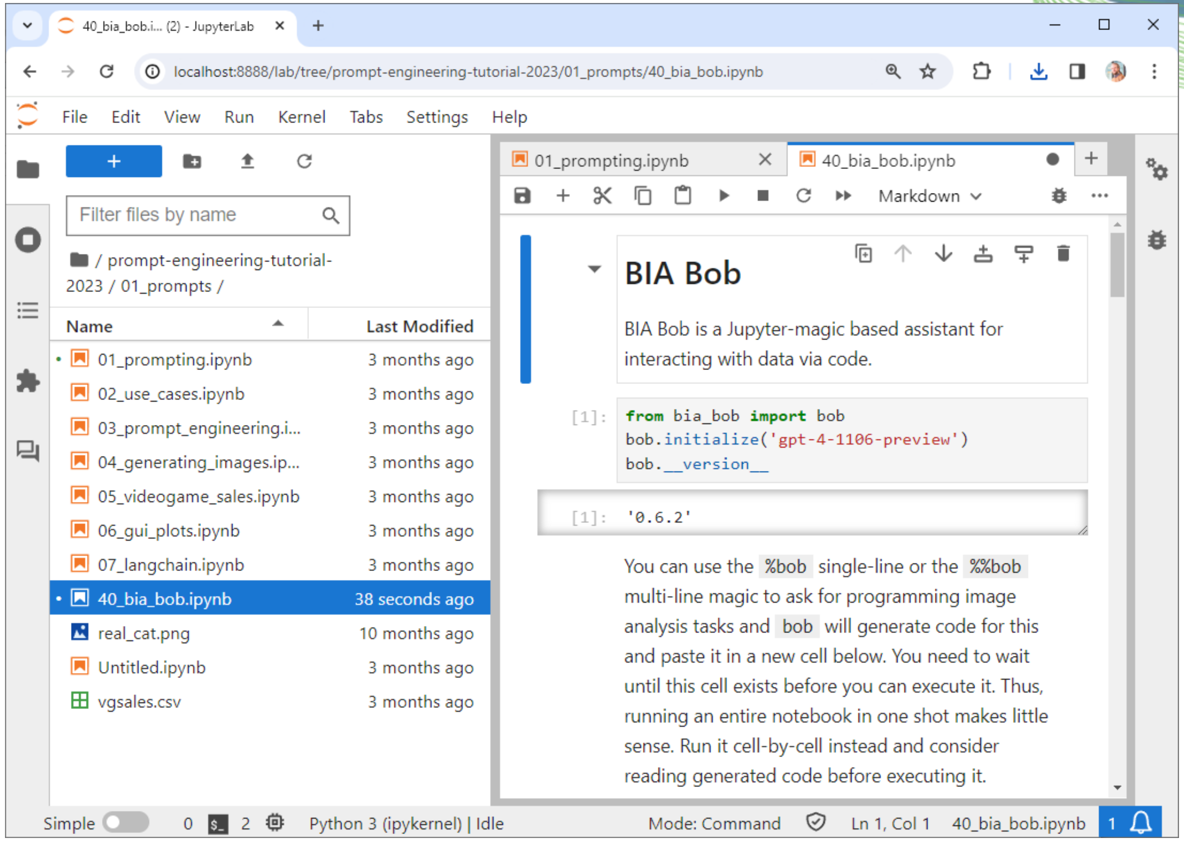
```
'0.6.2'
```

You can use the `%bob` single-line or the `%%bob` multi-line magic to ask for programming image analysis tasks and `bob` will generate code for this and paste it in a new cell below. You need to wait until this cell exists before you can execute it. Thus, running an entire notebook in one shot makes little sense. Run it cell-by-cell instead and consider reading generated code before executing it.

```
%bob Load the ../../data/blobs.tif image file and show it.
```

Step-by-step Plan:

1. Load the image file `../../data/blobs.tif` using `imread` from `skimage.io`.
2. Display the loaded image using `stackview.insight`.


40_bia_bob.i... (2) - JupyterLab
localhost:8888/lab/tree/prompt-engineering-tutorial-2023/01_prompts/40_bia_bob.ipynb
File
Edit
View
Run
Kernel
Tabs
Settings
Help
Filter files by name
/ prompt-engineering-tutorial-2023 / 01_prompts /
01_prompting.ipynb
40_bia_bob.ipynb
Markdown
Simple
0
2
Python 3 (ipykernel) | Idle
Mode: Command
Ln 1, Col 1
40_bia_bob.ipynb
1

Name	Last Modified
01_prompting.ipynb	3 months ago
02_use_cases.ipynb	3 months ago
03_prompt_engineering.i...	3 months ago
04_generating_images.ip...	3 months ago
05_videogame_sales.ipynb	3 months ago
06_gui_plots.ipynb	3 months ago
07_langchain.ipynb	3 months ago
40_bia_bob.ipynb	38 seconds ago
real_cat.png	10 months ago
Untitled.ipynb	3 months ago
vgsales.csv	3 months ago

BIA Bob

BIA Bob is a Jupyter-magic based assistant for interacting with data via code.

```
[1]: from bia_bob import bob
     bob.initialize('gpt-4-1106-preview')
     bob.__version__
```

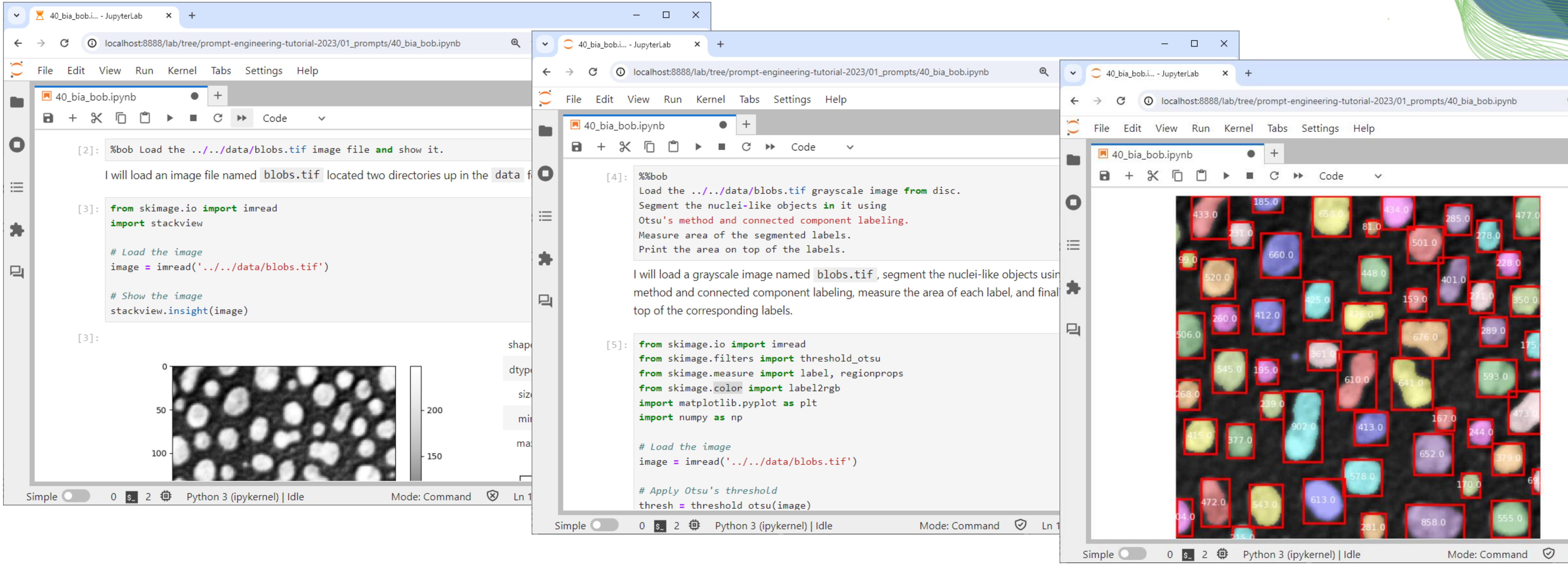
```
[1]: '0.6.2'
```

You can use the `%bob` single-line or the `%%bob` multi-line magic to ask for programming image analysis tasks and `bob` will generate code for this and paste it in a new cell below. You need to wait until this cell exists before you can execute it. Thus, running an entire notebook in one shot makes little sense. Run it cell-by-cell instead and consider reading generated code before executing it.

https://haesleinhuepf.github.io/BioImageAnalysisNotebooks/07_prompt_engineering/40_bia_bob.html

Exercise: BiA-Bob

Download the notebooks and reproduce them!


40_bia_bob.i... - JupyterLab
localhost:8888/lab/tree/prompt-engineering-tutorial-2023/01_prompts/40_bia_bob.ipynb
File Edit View Run Kernel Tabs Settings Help
40_bia_bob.ipynb
Code
Simple 0 2 Python 3 (ipykernel) | Idle Mode: Command

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

I will load an image file named `blobs.tif` located two directories up in the `data` f

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

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

     # Show the image
     stackview.insight(image)
```

[3]:
0
50
100
200
150
shap
dtyp
siz
mi
ma

```
[4]: %%bob
     Load the ../../data/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.
```

I will load a grayscale image named `blobs.tif`, segment the nuclei-like objects usin method and connected component labeling, measure the area of each label, and final top of the corresponding labels.

```
[5]: from skimage.io import imread
     from skimage.filters import threshold_otsu
     from skimage.measure import label, regionprops
     from skimage.color import label2rgb
     import matplotlib.pyplot as plt
     import numpy as np

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

     # Apply Otsu's threshold
     thresh = threshold_otsu(image)
```

https://haesleinhuepf.github.io/BioImageAnalysisNotebooks/07_prompt_engineering/40_bia_bob.html

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