Hogeschool van Amsterdam

sense maker samst erdam

ollama meetup 2

Michiel Bontenbal

Sensemakers Amsterdam

15 May 2024

Creating Tomorrow

Ollama meetup 2 15 May

First part - together

- Recap of first meetup & fresh start for newcomers
- Bring your own model
- Vision – Language Models like LLaVA
- Challenges, your ideas and show & tell

Second part – split group

Beginnners / new: Run Notebooks from meetup 1.

Advanced: “Chat with your document” – using Langchain (aka RAG)

ollama meetups 3rd wednesday

sense maker samst erdam

17 april : Introduction
- Starting with ollama
- Program with ollama and Python
- Challenges, your ideas and show & tell

15 May:
- Recap of first meetup & fresh start for newcomers
- Bring your own model
- Vision-Language Models (VLM's)
- "Chat with your document" – using Langchain (aka RAG)
- Challenges, your ideas and show & tell

19 June:
- Recap
- Chat with an avatar
- Advanced topics (to be determined)
- Challenges, your ideas and show & tell

First Wednesday of the month
1 May - 5 June
Do It Yourself Together
@ OBA

sense
maker
samst
erdam

Some questions… raise your hand!

- Who has worked with:
  - Python and Jupyter Notebook
  - Retrieval Augmented Generation

Recap first meetup

sense maker samst erdam

Why ollama?
Why running LLM’s on your laptop?

1. Data stays on your device
   - Privacy / no leakage of sensitive data / company policy
2. No longer dependent on internet connection
3. Lower costs with less data usage
4. Saves energy usage
5. Bring your own model (this meetup)

sense maker samst erdam

Models for Language, Vision and Coding

Chat

```
>>> What is the captial of the Netherlands?
 The capital of the Netherlands is Amsterdam.
```

Ask questions about images

Create and improve code

```
>>> create a python scriot to capture an image with the webcam
```python
import cv2

# Load the webcam
cap = cv2.VideoCapture(0)

# Check if the webcam is accessible
if not cap.isOpened():
    print("Error: Unable to open webcam.")
    exit()

# Capture an image
ret, frame = cap.read()

# Check if the image was captured successfully
if not ret:
    print("Error: Unable to capture image.")
    exit()

# Save the image
cv2.imwrite("webcam_image.jpg", frame)

# Release the webcam
cap.release()

print("Image captured successfully.")
```
```

sense maker samst erdam

Models come in different sizes – use the right size for your laptop

For example, Llama2 has 3 sizes: 7B, 13B and 70B model.

- 7B means 7 billion parameters which is ± 4 Gigabytes in size.

Check your hardware:

- 7b models need ± 8GB of RAM
- 13b models need ± 16GB of RAM
- 70b models need ± 64GB of RAM

Evaluate the performance of your model with:

```
ollama run tinyllama --verbose
```

sense maker samst erdam

run ollama

- Download and install via www.ollama.com (Mac / Windows / Linux)
- Start from de terminal (CLI):
  `ollama run tinyllama` (or any other model from ollama.com)
- Start chatting with the model!
- End ollama with `CTRL+C` or `/bye` or `/exit`
- You can then start another model.

Ollama challenges (tonight)

- **Beginnners: (guided)**
  - Try other language, vision or coding models
  - Program ollama with python & jupyter notebook

- **Experienced users (DIY)**
  - Ollama.ipynb
  - Ollama_llava_challenge.ipynb

https://tinyurl.com/ycyza52p

Work in pairs or teams!

First notebook only read code

Reorganising the room.

sense maker samst erdam

Challenges for next meetups

or better start tonight ☺

- Chat with your own documents or make a summary of your document with
- Make a front-end with Gradio for images
- Download a model from huggingface and run it with ollama
- Annotate my pictures / screenshots
- Talk with your laptop (text2speech & speech2text) (with whisper and ?? notebook)

Programming ollama with Python

sense maker samst erdam

Why program ollama with Python?

- Build a chatbot for in your browser (GUI) — Today!
- Integrate ollama in a webapplication: — Next month
  - Summarizing your own documents
  - Question answering
  - Visual question answering

sense maker samst erdam

Installing your programming tools

- Install Python: https://www.python.org/
- Install Visual Studio Code: https://code.visualstudio.com/
- Install the python VS code plugin, see:
  - https://code.visualstudio.com/docs/python/python-tutorial

sense maker samst erdam

Python programming ollama

Use the notebook to use python code from github:

`ollama.ipynb`

P.S. If you prefer JavaScrpt there is also a Node.js package:

`npm install ollama`

Run models from Huggingface

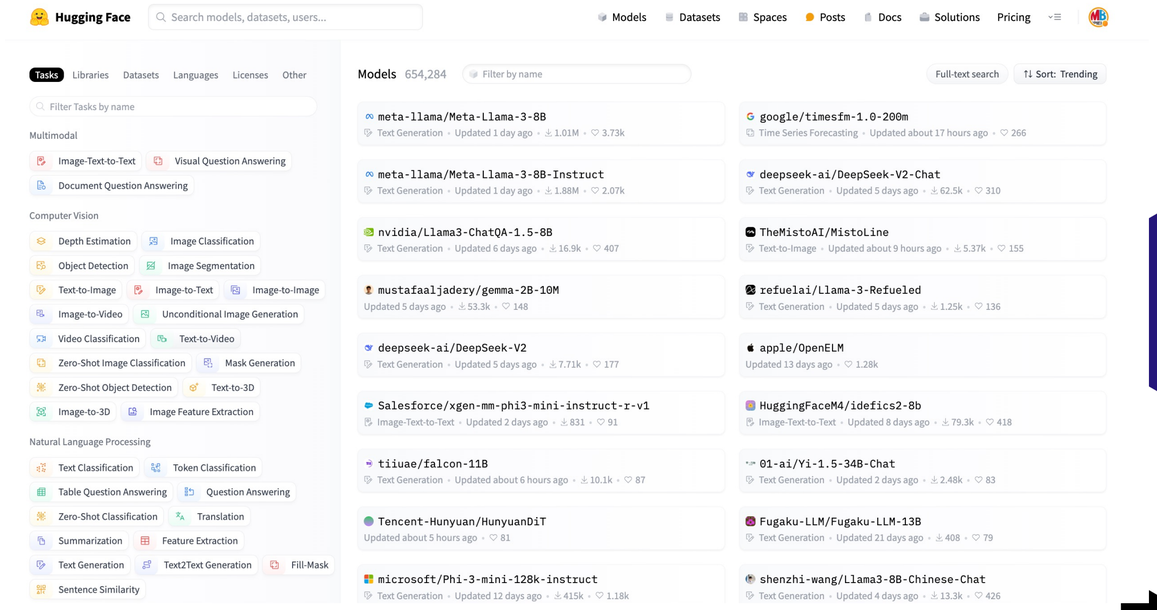
Huggingface Hub

- World’s biggest platform for AI models, datasets
- Individuals, research and Big Tech publish their work there
- Everything is open source

Huggingface also has

- Python packages like transformers, pipelines
- Courses
- Spaces

- Can we find a **quantized model** in Dutch?



Let’s try this with some Dutch models.

You might want to learn dutch or chat in Dutch, then use a Dutch model.

Two models have been developed for Dutch

- GEITje
- Fietje

```
ollama run bramvanroy/geitje-7b-ultra-gguf
```

- More: https://goingdutch.ai/nl/posts/why-geitje/

The .gguf file format indicates it’s a **quantized version** of a model

Basics of ‘quantization’

We reduce precision, like in the image of pres. Obama.

However in AI we work with numbers and computers work in bytes

Example: reduce precision from *floatingpoint32* (=32 bits) to *integer8* (=8 bits) values.
For LLM’s we call this ‘q8 quantization’

FP32: [33.623563422, 12.646104098, -51.583920991, …]
FP16: [33.6234, 12.6461, -51.5839, …]
INT8: [82, 44, -23, …]

It’s best known from the “llama.cpp project” that created the ggml and gguf file formats. More on this in the last meetup!

But there is more: other techniques are e.g. *pruning* (=removing unused layers) and optimizing for hardware.

Exercise

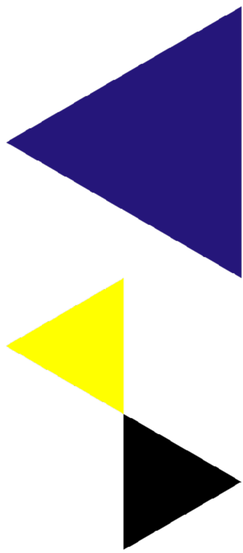
- Go to huggingface hub/models
- Filter on ‘GGUF’ and language
- Download a model from huggingface that you would like.

Sometimes you can run it directly in ollama:
`ollama run bramvanroy/geitje-7b-ultra-gguf`

Sometimes it links right back to ollama.com:
https://huggingface.co/BramVanroy/fietje-2b-chat-gguf

Sometimes: download the gguf manually, create a modelfile, run the model. See
https://youtu.be/fnvZJU5Fj3Q?t=153

Exercise: Use your python code with this new model.



A deeper look at Vision Language Models

- Intro
- CLIP
- Vision Language Models

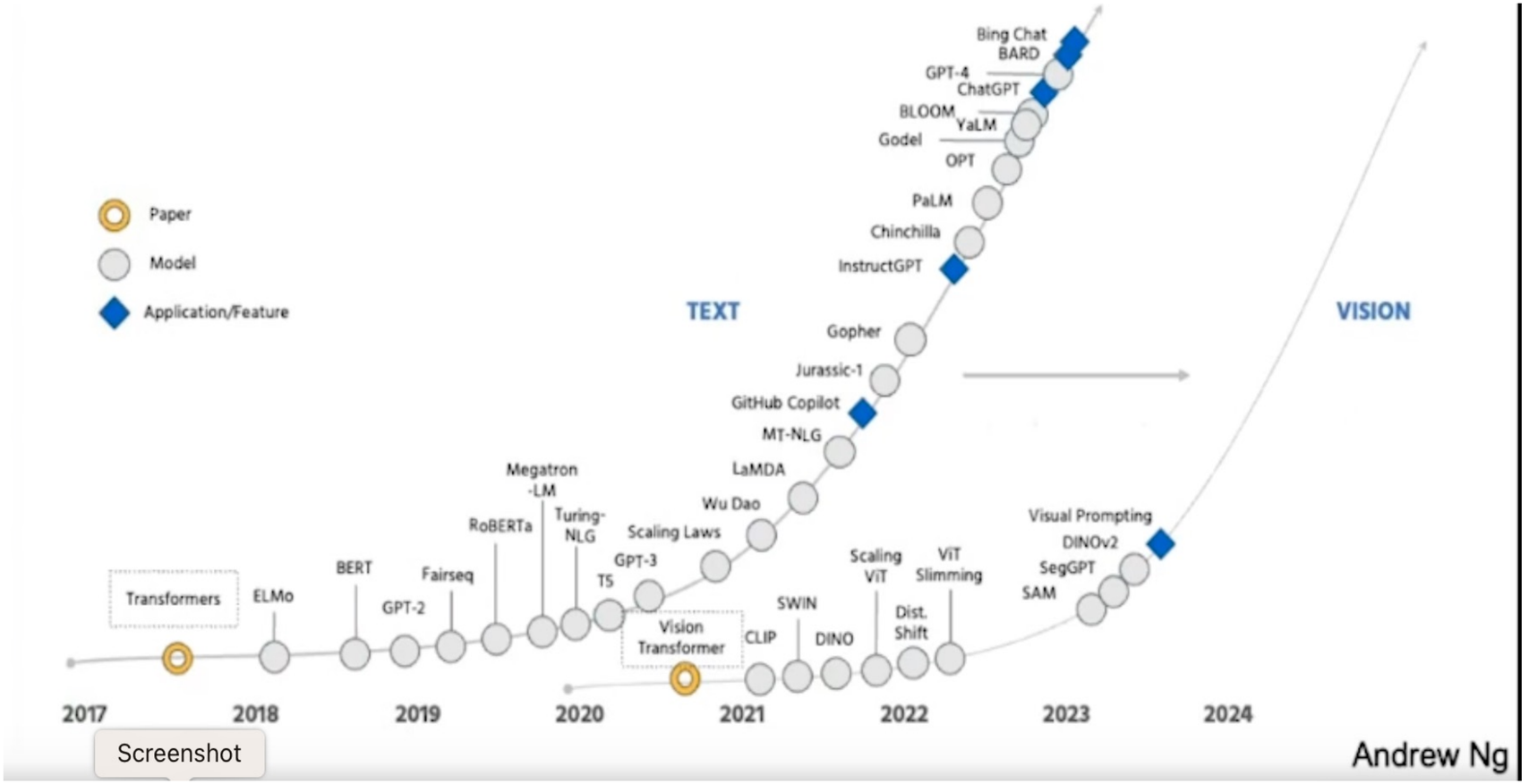
sense maker samst erdam

What you can do with Vision models

Talk with them using LLaVA or any other Vision – Language Model!

1. Human input with speech-2-text
2. Visual Question Answering with LLaVA
3. Text-2-Speech

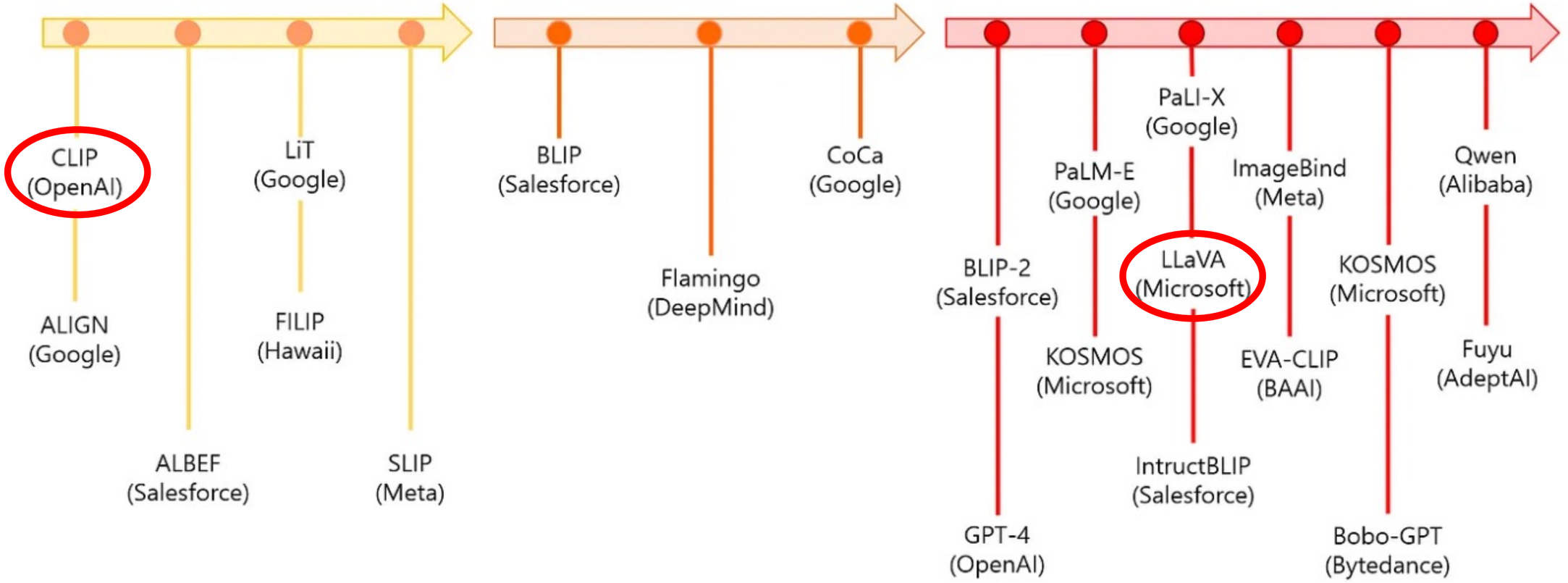
The ChatGPT revolution is coming for vision!



Screenshot

Large Vision Models =. Large Multimodal Models

Vision Language Models



How can computers find this relationship?

‘Paris’

As always in AI, we convert the text and image to numbers.
In this case they are called **‘vector embeddings’.**

Read more: https://www.geeksforgeeks.org/what-are-embeddings-in-machine-learning/

We use an ‘embedding model’ and compare the vectors. Vectors capture the meaning.

‘Paris’

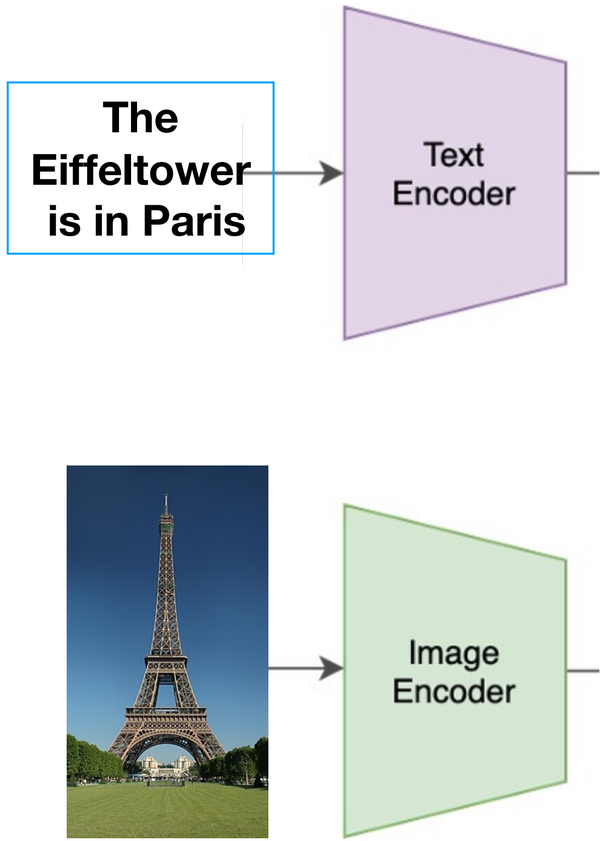
[0.5, 0.3, …, 0.7]

[0.4, 0.3, …, 0.6]

Compare vectors

We will use OpenAI’s CLIP with text-image pairs

Contrastive Language-Image Pre-training



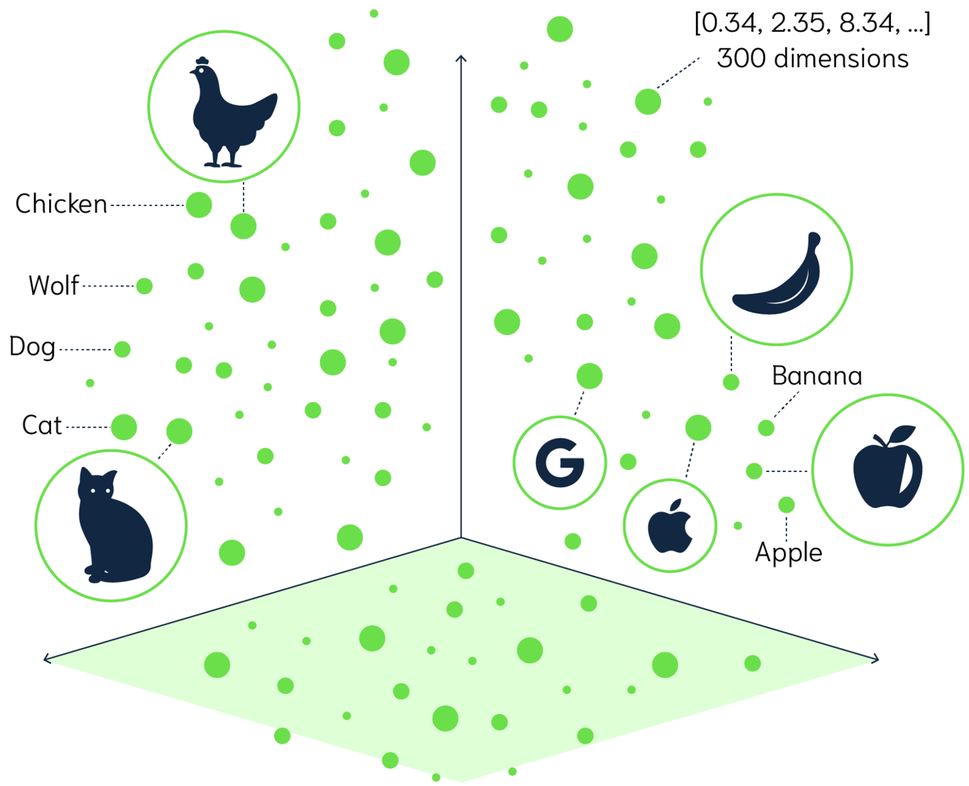
- Trained on 400 million pairs of images with text captions
- Gepubliceerd in 2021 tegelijk met Dall-e

- Use CLIP to describe images => ‘**image2text**’
- Dall-e is the reverse => **text2image**

Sources:

- https://github.com/OpenAI/CLIP
- https://openai.com/research/clip
- https://platform.openai.com/docs/guides/embeddings/what-are-embeddings

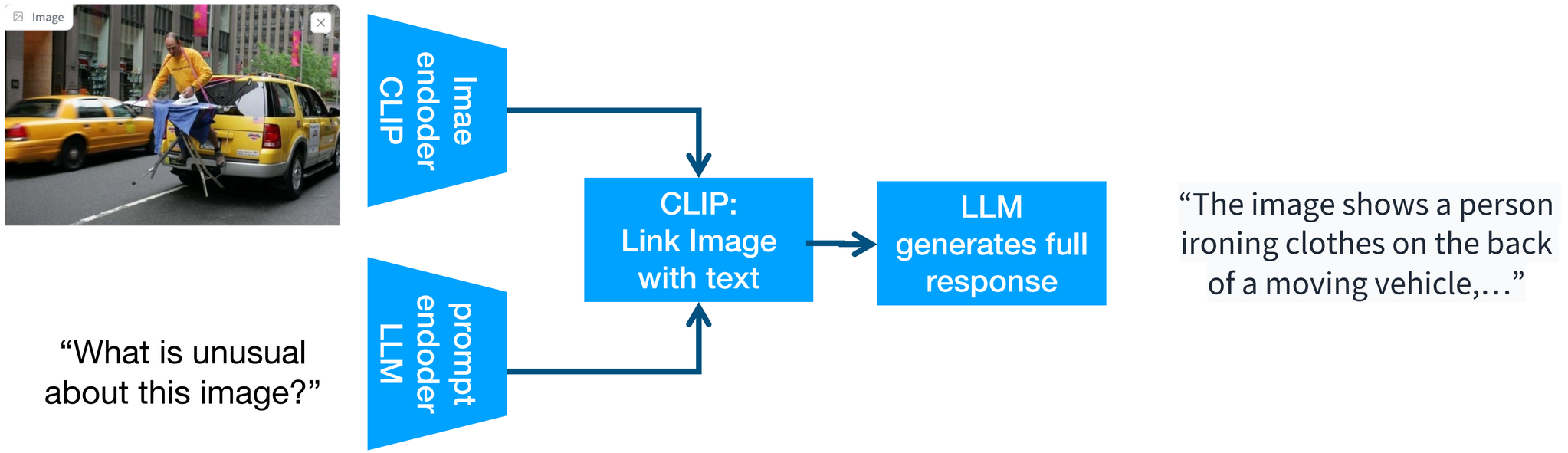
Texts and images in vector space



Words and images with same meaning are close in vector space.

LLaVA: Vision Language Model

Combines CLIP with a LLM



Exercises

Use LLAVA :

`ollama_llava_challenges.ipynb`

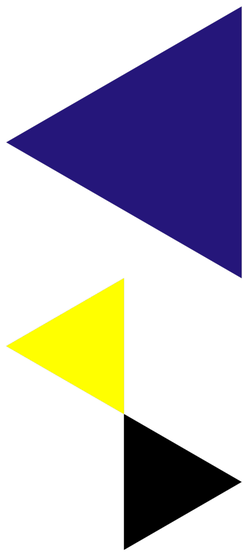
- Self driving car.ipynb

Learn more about CLIP with the following notebooks (CLIP is not in Ollama)

`Find_similar_images_using_CLIP.ipynb`

`Interacting_with_CLIP.ipynb`

Rename your screenshots with the following code



Chat with your document

Retrieval Augmented Generation

-

Retrieval Augmented Generation

- Hottest thing in AI right now
- Generate an augmented (=improved) answer from an LLM based on information retrieved from your document.

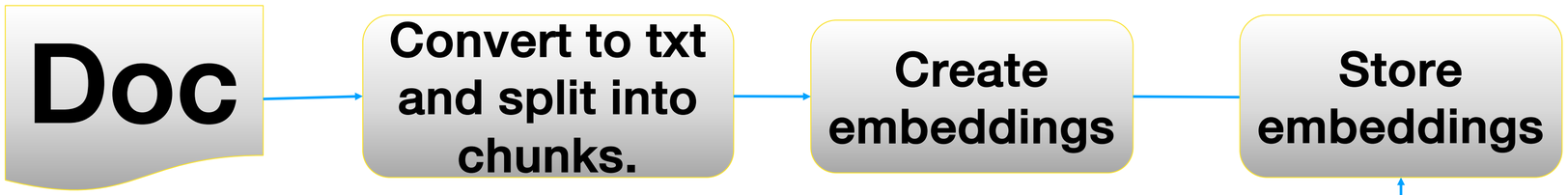
- Used for tasks as question – answering and summarizing

Most common python packages are Langchain and Llama_index

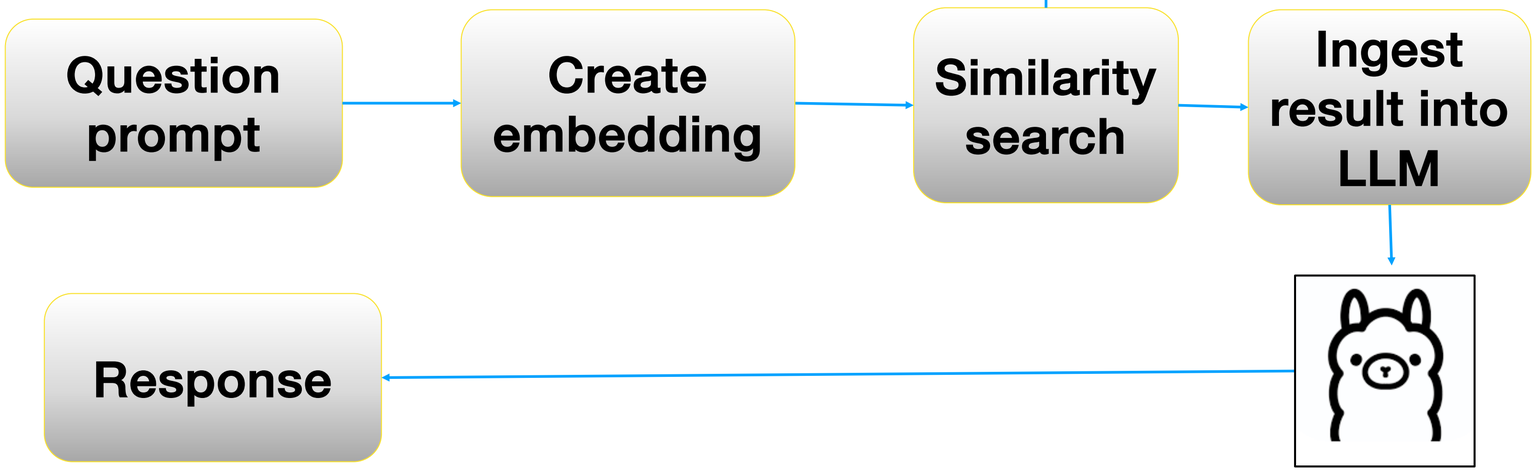
Many challenges in RAG: Optimal chunk length, hallucinations, prompting techniques, reranking, which embedding model, hybrid search (combine semantic search with traditional search), evaluation, retrieve info from webpages

Flow

1. Preparation

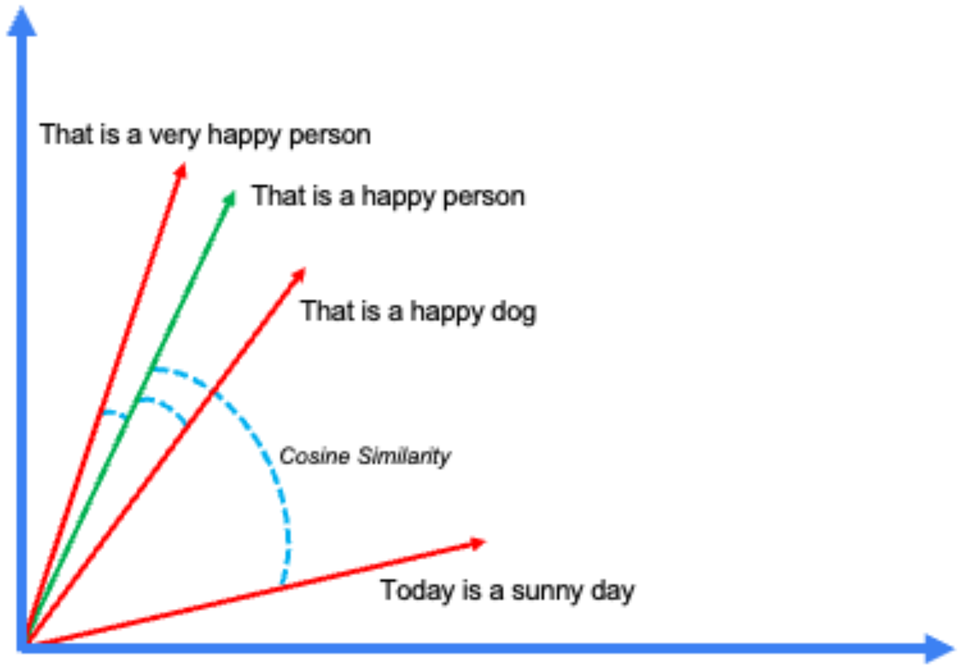


2. Inference



Similarity search

- Similarity search is done by comparing vector embeddings.
- Most often we use ‘cosine similarity’.
- It gives meaning to search – not just string search!
- Try it at www.perplexity.ai



RAG challenge + notebook

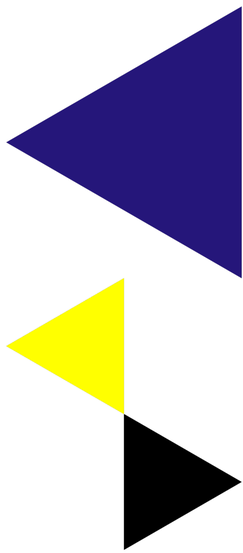
- Get your chatbot to talk with a document.

Use the notebooks to do RAG on Peter Pan book:

`RAG_basics_from_scratch_with_ollama.ipynb`

Bonus if you want to learn more about some theory behind RAG:

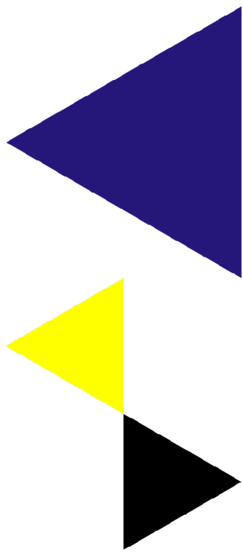
`RAG_exercises.ipynb`



Learn more RAG

Short courses with deeplearning.ai

- https://www.deeplearning.ai/short-courses/building-multimodal-search-and-rag/
- https://learn.deeplearning.ai/courses/preprocessing-unstructured-data-for-llm-applications



Bonus slides

-

sense maker samst erdam

Ollama alternatives

- GPT4all
- LM-studio
- LocalGPT (also does RAG natively)
- Jan.ai (my favorite)

Comparison of Local LLM frameworks

Framework	GUI	# Available models	API	Python package	Vision models	Also online models
ollama	No		Yes	Yes	Yes	No
GPT4all						
LM-studio	Yes					?
LocalGPT	Yes					?
Jan	Yes					Yes

Future of running LLM's: Browser, super fast inference, iPhone

Run LLM's in your browser:

- https://huggingface.co/spaces/Xenova/experimental-phi3-webgpu

Groq super fast inference:

- Groq have developed a new type of processor called LPU:
- It's extremely fast (like 300 token/sec) and you can try it for free at
- www.groq.com (make sure you use a large model like 70B params!)

LLMFarm app to run LLM's on iPhone

- An app has been developed, you still need to run it with Testflight
- LinkedIn post of the announcement:
- Instruction video: https://www.youtube.com/watch?v=5QEDNZlDf-c