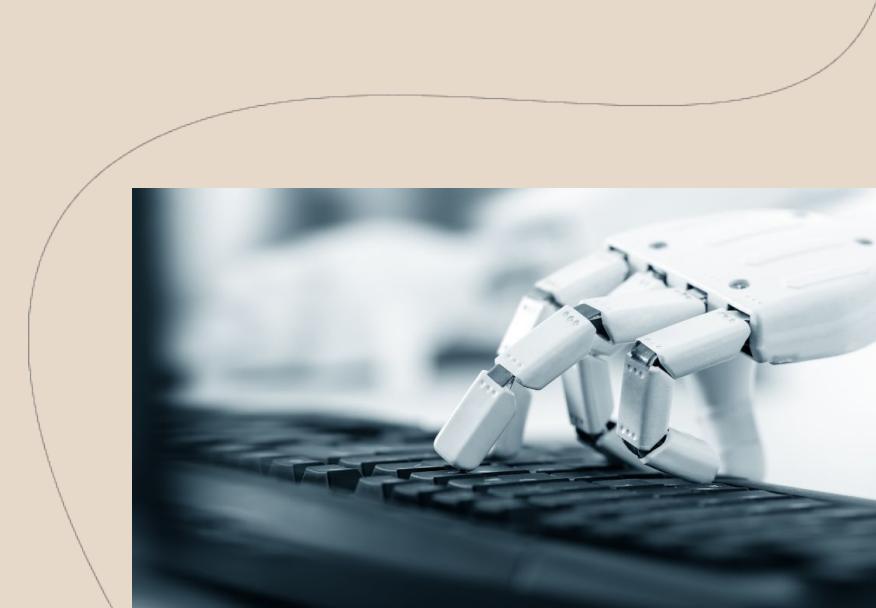




Postgraduaat

AI Technical Architect

academiejaar '23-'24



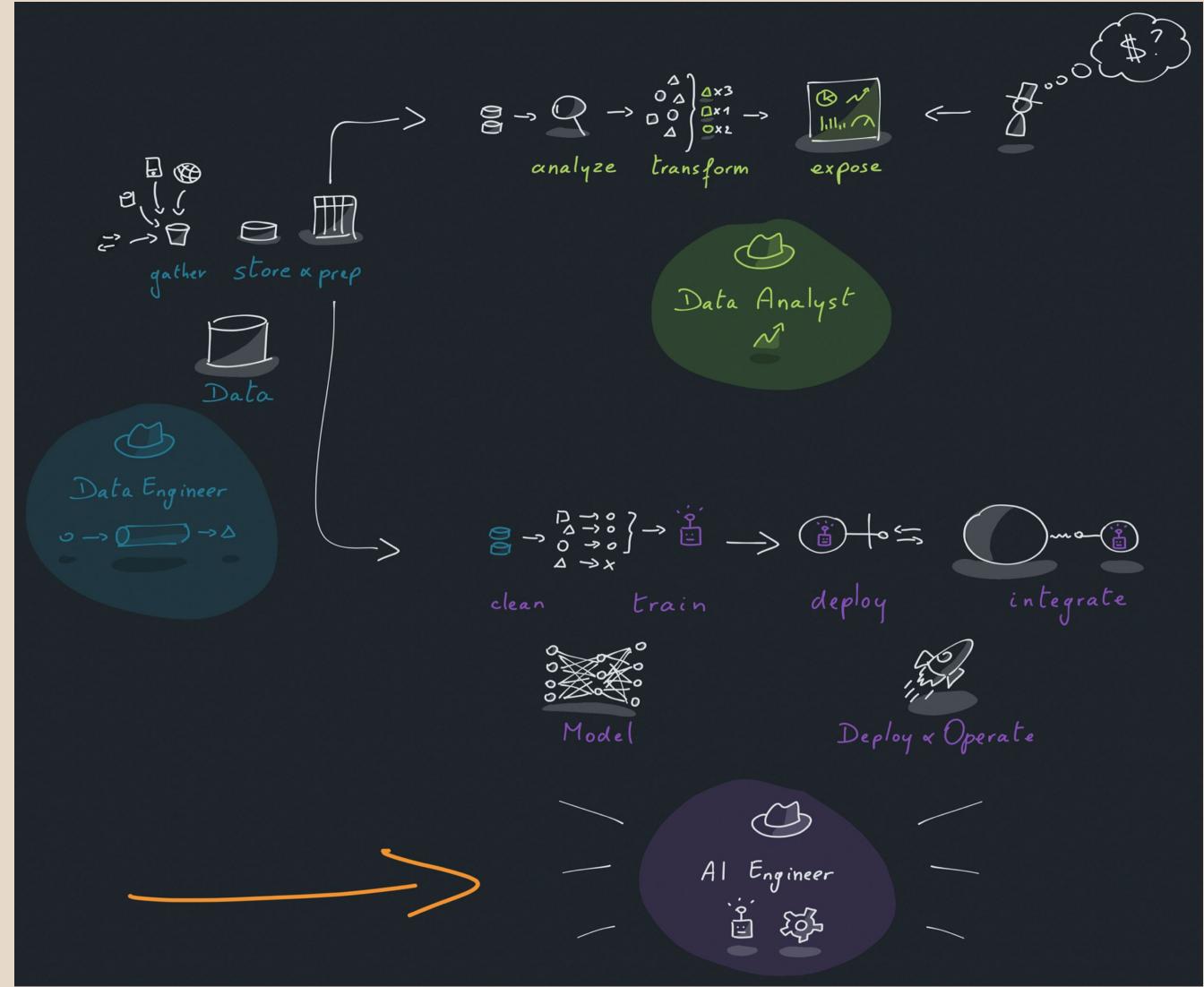


AI Technical Architect

Technologies
Hands on day 2

Tim Dupont
Sam Van Rijn

17 jan '24





N e X T

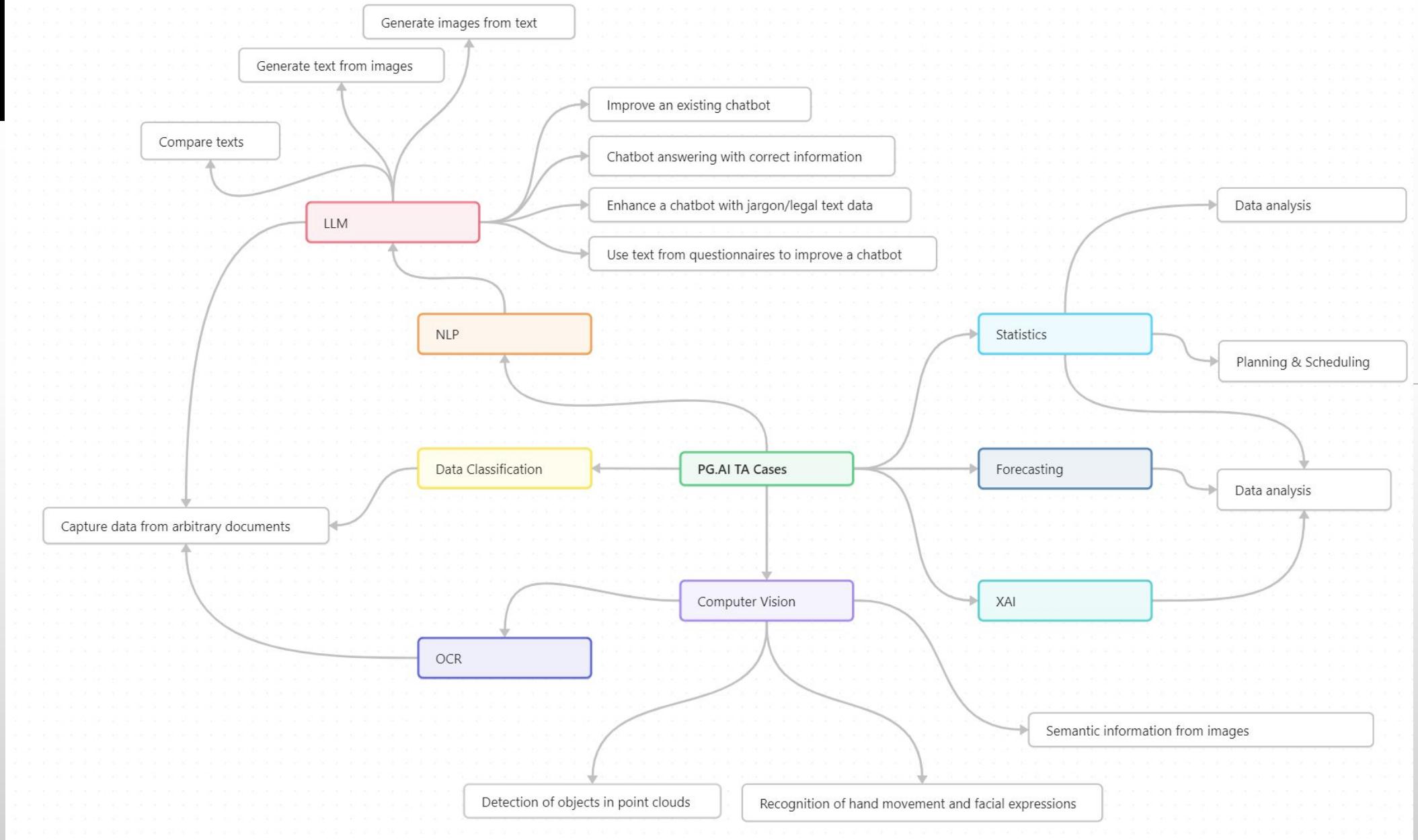


Table of contents

1. Frameworks & libraries: OCR
2. Frameworks & libraries: Backend
3. Frameworks & libraries: Python dependency handling + Docker
4. AI in the Cloud
5. Model deployment
6. Practical example
7. Large Language Models



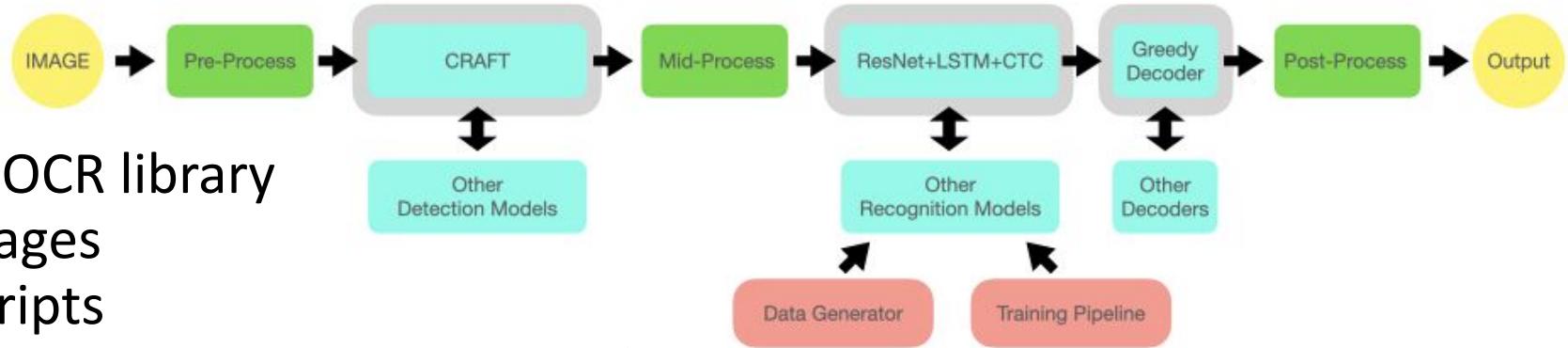
N e x T

Frameworks & libraries

OCR

EasyOCR

EasyOCR Framework



- Open Source Python OCR library
- 80+ supported languages
- all popular writing scripts
(Latin, Chinese, Arabic, Devanagari, Cyrillic, ...)
- In the (near) future: handwritten text support

image	<p>Reduce your risk of coronavirus infection:</p> <ul style="list-style-type: none">Clean hands with soap and water or alcohol-based hand rubCover nose and mouth when coughing and sneezing with tissue or flexed elbowAvoid close contact with anyone with cold or flu-like symptomsThoroughly cook meat and eggsNo unprotected contact with live wild or farm animals <p>World Health Organization</p>					
result	<p>'Reduce your risk of coronavirus infection:', 'Clean hands with soap and water', 'or alcohol based hand rub', 'Cover nose and mouth when coughing and', 'sneezing with tissue or flexed elbow', 'Avoid close contact with anyone with', 'cold or flu like symptoms', 'Thoroughly cook meat and eggs', 'No unprotected contact with live wild', 'or farm animals', 'World Health', 'organization'</p>	<p>' เส้นทางลัด', 'เพชรบุรี'</p>	<p>'du 1"', 'Mairie', 'Palais du', 'LOUVRE', 'LES ARTS DÉCORATIFS', 'Musée du LOUVRE', 'Théâtre', 'du PALAIS-ROYAL'</p>	<p>On-Premise or Cloud / API</p>	<p>Improve accuracy up to > 99% with your data</p>	<p>Fully Customizable with 80+ languages supported</p>
				<p>Layout / Table / Form Recognition</p>	<p>Signature / QRcode / Barcode Extraction</p>	<p>Semi-Automated Labeling Tool</p>

[INFO]

More info: <https://www.jaide.ai/easyocr> and <https://github.com/JaideAI/EasyOCR>



N e x T

Frameworks & libraries

Backend

AI "back-end"



django

Not Python



Flask



- Lightweight WSGI web application framework.
- Designed to make getting started quick and easy
- Micro web framework written in Python
- Does not require particular other tools or libraries
- No database abstraction layer, form validation, or any other components
- Supports extensions
- Used by Pinterest and LinkedIn

```
# save this as app.py
from flask import Flask, escape, request

app = Flask(__name__)

@app.route('/')
def hello():
    name = request.args.get("name", "World")
    return f'Hello, {escape(name)}!'
```

```
$ flask run
 * Running on http://127.0.0.1:5000/ (Press CTRL+C to quit)
```

[INFO]

More info: <https://palletsprojects.com/p/flask> and <https://flask.palletsprojects.com/en/2.1.x>

FastAPI



- **Fast:** One of the fastest Python frameworks available. (On par with Node.js)
- **Easy:** Designed to be easy to use and learn. (Less time reading docs.)
- **Short:** Minimize code duplication.
- **Robust:** Get production-ready code.

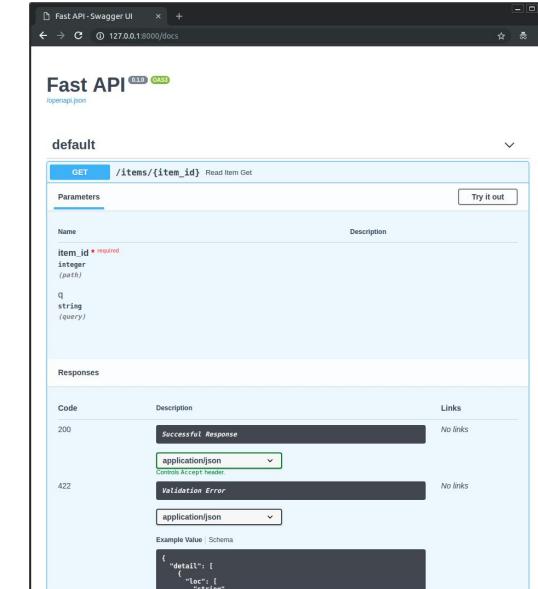
```
from typing import Union

from fastapi import FastAPI

app = FastAPI()

@app.get("/")
def read_root():
    return {"Hello": "World"}

@app.get("/items/{item_id}")
def read_item(item_id: int, q: Union[str, None] = None):
    return {"item_id": item_id, "q": q}
```



[INFO]

More info: <https://fastapi.tiangolo.com>

Django

django

- High-level Python web framework
- Rapid development and clean, pragmatic design
- Ease the creation of complex, database-driven websites
- Used by: Instagram, Mozilla, Bitbucket, ...



Ridiculously fast.

Django was designed to help developers take applications from concept to completion as quickly as possible.



Reassuringly secure.

Django takes security seriously and helps developers avoid many common security mistakes.



Exceedingly scalable.

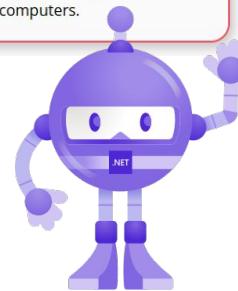
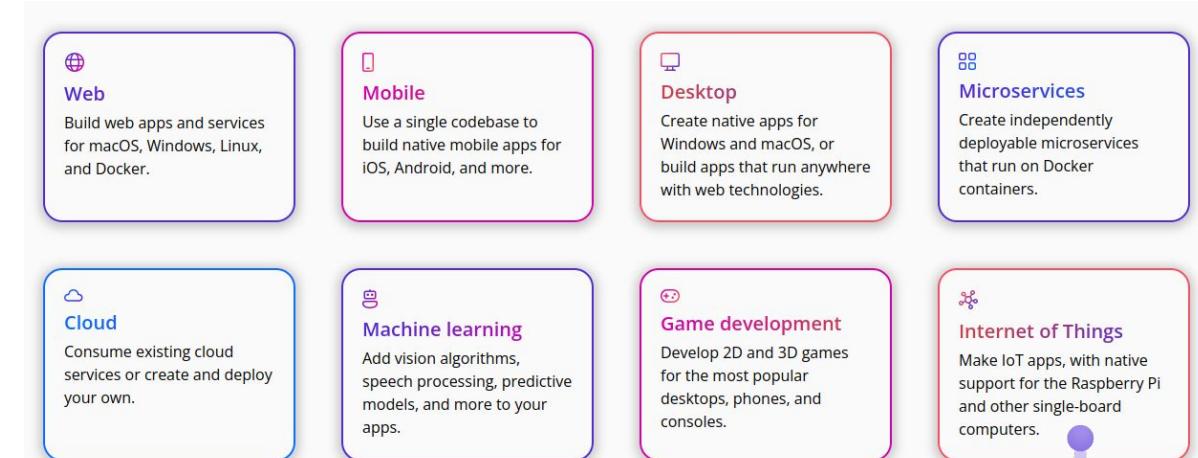
Some of the busiest sites on the web leverage Django's ability to quickly and flexibly scale.

[INFO]

More info: <https://www.djangoproject.com>



- Free and open-source, managed computer software framework
- Written in C++ and C#
- For Windows, Linux, and macOS
- It is a cross-platform successor to .NET Framework
- Currently version 7 (December 13, 2022)
- Supports four cross-platform scenarios:
 - ASP.NET Core web apps
 - Command-line apps
 - Libraries
 - Universal Windows Platform apps



Spring



spring®

- Framework for modern Java-based enterprise applications
- On any kind of deployment platform
- Uses a comprehensive programming and configuration model

Microservices

Quickly deliver production-grade features with independently evolvable microservices.

Reactive

Spring's asynchronous, nonblocking architecture means you can get more from your computing resources.

Cloud

Your code, any cloud—we've got you covered. Connect and scale your services, whatever your platform.

Web apps

Frameworks for fast, secure, and responsive web applications connected to any data store.

Serverless

The ultimate flexibility. Scale up on demand and scale to zero when there's no demand.

Event Driven

Integrate with your enterprise. React to business events. Act on your streaming data in realtime.

Batch

Automated tasks. Offline processing of data at a time to suit you.

[INFO]

More info: <https://spring.io>

Node.js



- Back-end JavaScript runtime environment
- Executes JavaScript code outside a web browser
- Cross-platform (Runs on the V8 engine)
- Corporate users:
GoDaddy, Groupon, IBM, Microsoft, Netflix, PayPal, Amazon Web Services...

```
const http = require('http');

const hostname = '127.0.0.1';
const port = 3000;

const server = http.createServer((req, res) => {
  res.statusCode = 200;
  res.setHeader('Content-Type', 'text/plain');
  res.end('Hello World');
});

server.listen(port, hostname, () => {
  console.log(`Server running at http://${hostname}:${port}/`);
});
```

[INFO]

More info: <https://nodejs.org> and <https://github.com/nodejs/node>

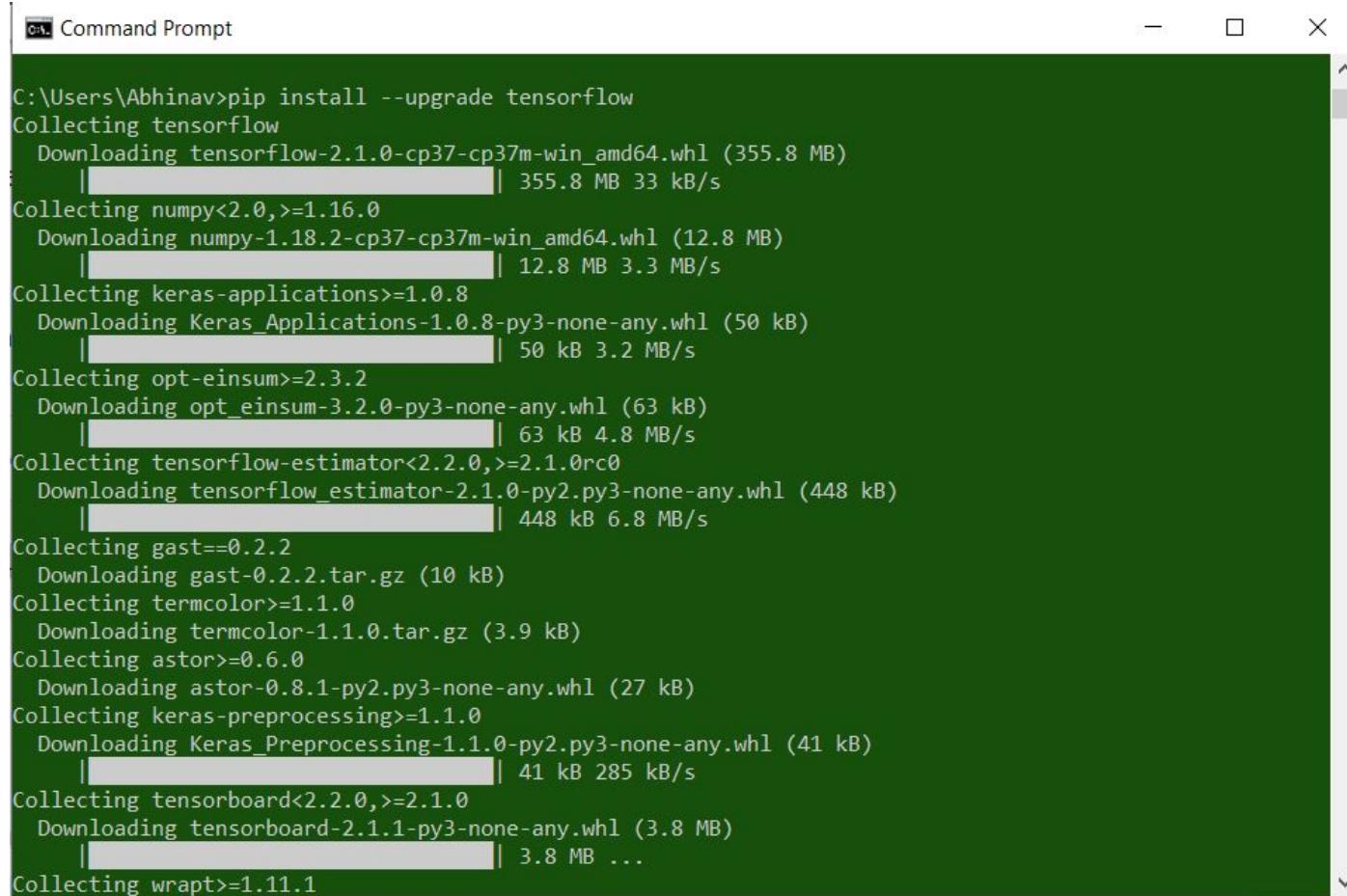


N e x T

Frameworks & libraries

Python dependency handling

How to install Python libraries: pip



A screenshot of a Windows Command Prompt window titled "Command Prompt". The window shows the output of a pip command to upgrade TensorFlow and its dependencies. The output includes the command run, the collection of packages, their download URLs, file sizes, and download speeds. Progress bars are visible for each download.

```
C:\Users\Abhinav>pip install --upgrade tensorflow
Collecting tensorflow
  Downloading tensorflow-2.1.0-cp37-cp37m-win_amd64.whl (355.8 MB)
    |██████████| 355.8 MB 33 kB/s
Collecting numpy<2.0,>=1.16.0
  Downloading numpy-1.18.2-cp37-cp37m-win_amd64.whl (12.8 MB)
    |██████████| 12.8 MB 3.3 MB/s
Collecting keras-applications>=1.0.8
  Downloading Keras_Applications-1.0.8-py3-none-any.whl (50 kB)
    |██████████| 50 kB 3.2 MB/s
Collecting opt-einsum>=2.3.2
  Downloading opt_einsum-3.2.0-py3-none-any.whl (63 kB)
    |██████████| 63 kB 4.8 MB/s
Collecting tensorflow-estimator<2.2.0,>=2.1.0rc0
  Downloading tensorflow_estimator-2.1.0-py2.py3-none-any.whl (448 kB)
    |██████████| 448 kB 6.8 MB/s
Collecting gast==0.2.2
  Downloading gast-0.2.2.tar.gz (10 kB)
Collecting termcolor>=1.1.0
  Downloading termcolor-1.1.0.tar.gz (3.9 kB)
Collecting astor>=0.6.0
  Downloading astor-0.8.1-py2.py3-none-any.whl (27 kB)
Collecting keras-preprocessing>=1.1.0
  Downloading Keras_Preprocessing-1.1.0-py2.py3-none-any.whl (41 kB)
    |██████████| 41 kB 285 kB/s
Collecting tensorboard<2.2.0,>=2.1.0
  Downloading tensorboard-2.1.1-py3-none-any.whl (3.8 MB)
    |██████████| 3.8 MB ...
Collecting wrapt>=1.11.1
```

Package-management system written in Python used to install and manage software packages. It connects to an online repository of public packages, called the Python Package Index (PyPI)

[INFO]

More info: <https://pypi.org> and <https://pypi.org/project/pip>

Python dependencies: requirements.txt

First, output `requirements.txt` to a file.

```
$ pip freeze > requirements.txt
```

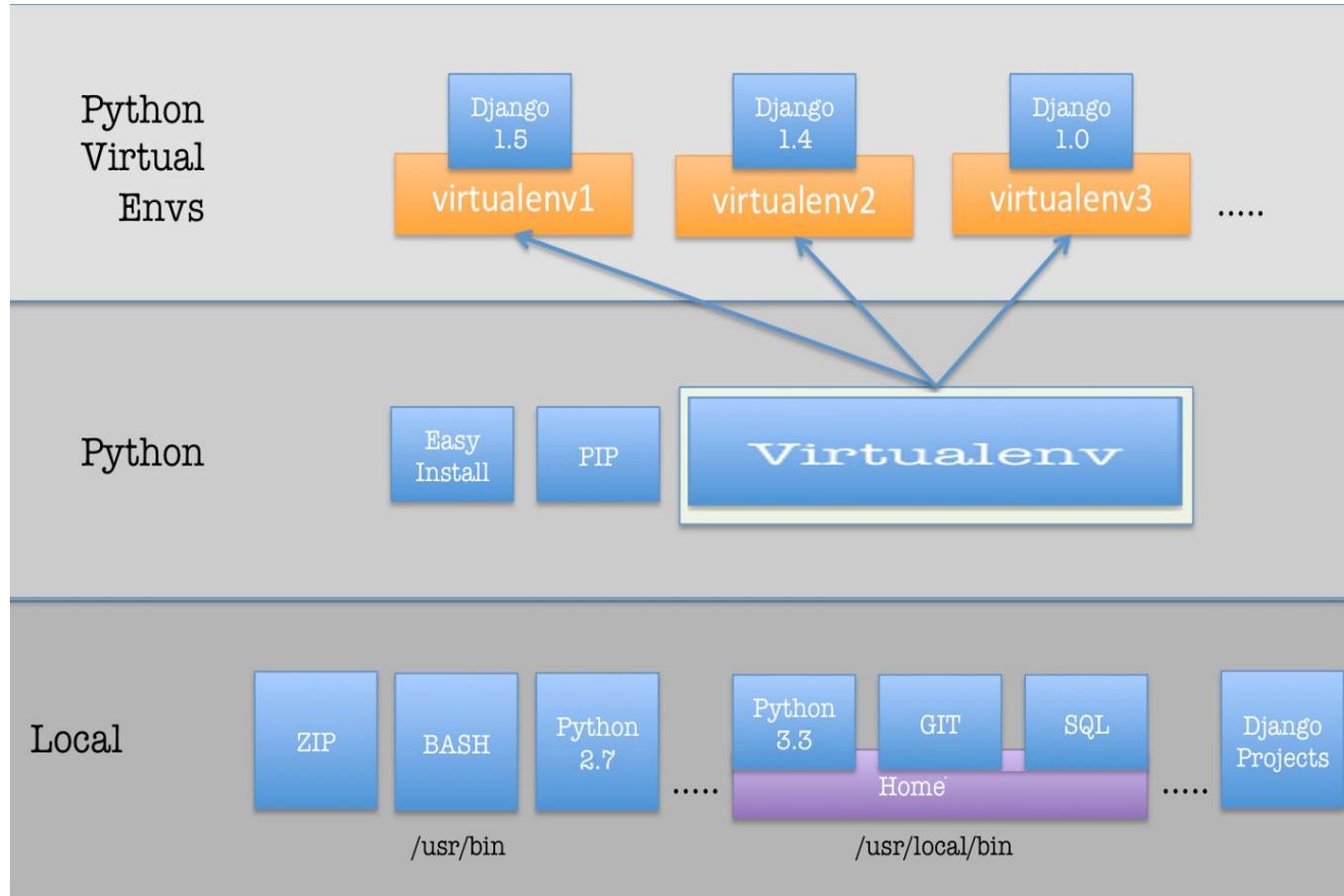
Copy or move this `requirements.txt` to another environment and install with it.

```
$ pip install -r requirements.txt
```

Example requirements.txt:

```
flask==2.0.2
keras==2.7.0
tensorflow==2.7.0
scipy==1.7.3
```

Python environments: venv



Problem: pip installs system wide...

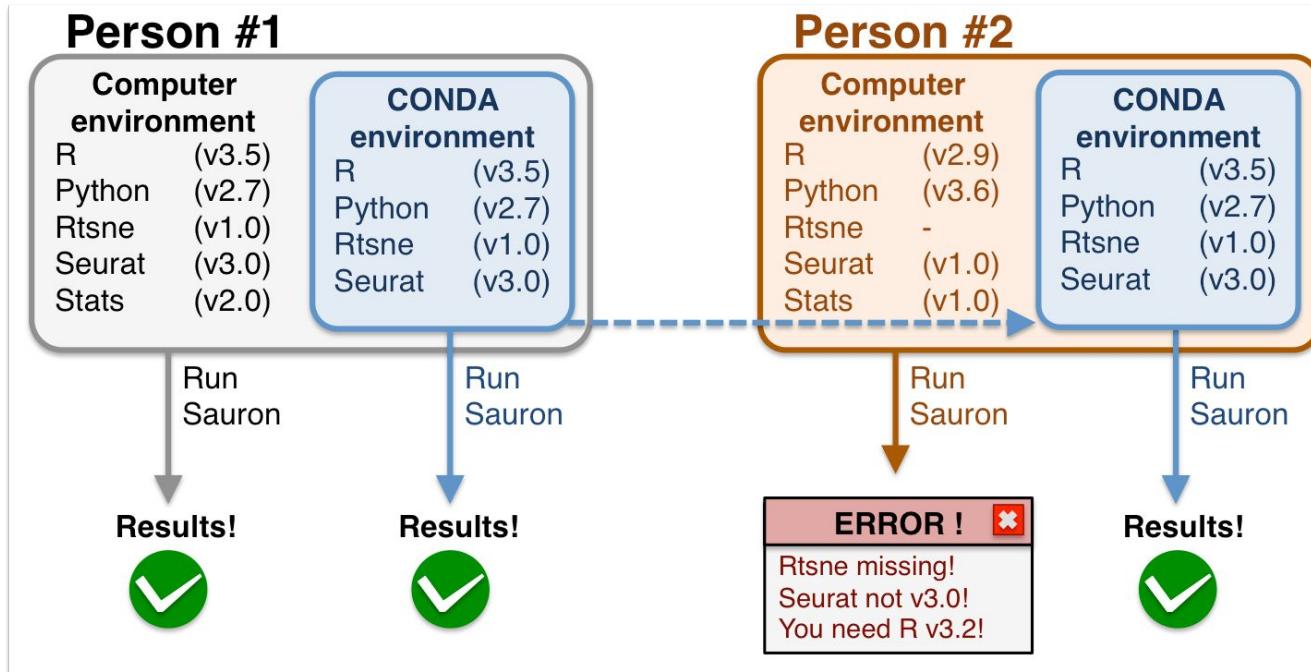
Solution?: venv

- Lightweight “virtual environments”
- own package directories, optionally isolated from system package directories
- own Python binary
- own independent set of installed Python packages

[INFO]

More info: <https://www.cloudfoundry.org>

Python environments: conda

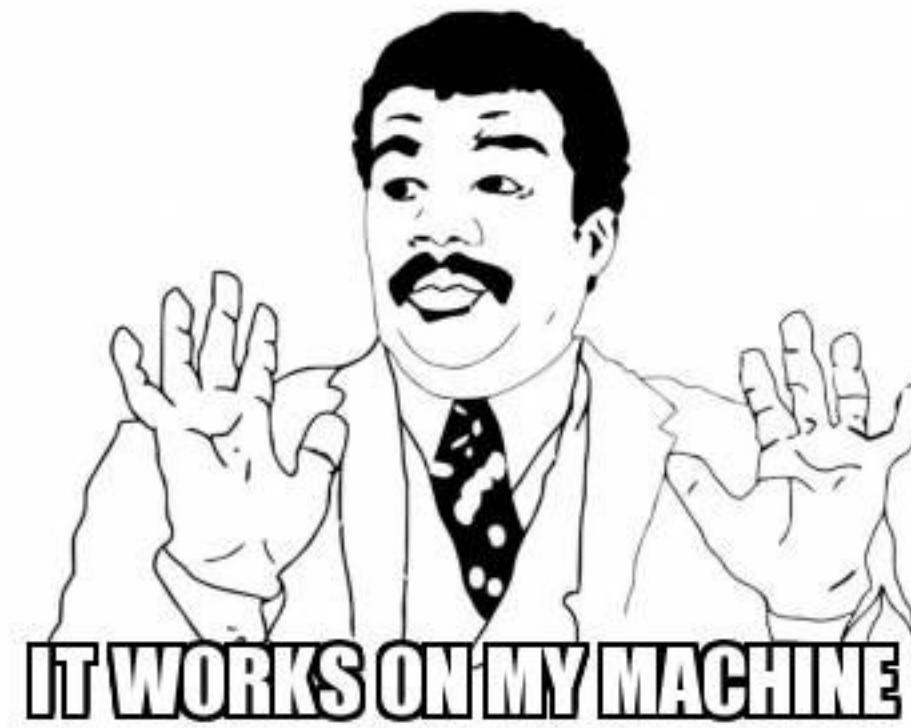


Problem: AI is not only Python!
(OpenCV, ROS, CUDA, ...)

Solution?: conda
Cross-platform language-agnostic
package manager and environment
management system.

**Problem: Still a hassle to set-up...
But it works on my machine!**

Containers



Benefits of containers

- Their images are self-contained
- They can be run ad-hoc, no booting
- Minimal overhead
- Standardized environment
- Host isolation (to some extent)
- Cloud-ready
- **Repeatable and reproducible (AI & robotics/research) solutions**



Containers: Docker

... is the **most used way to create reproducibility, isolated containers.** It's a handy and lightweight way to encapsulated an application to make it **easy to (re)deploy it time after time with the same outcome.**

The Docker website has some more information on containers:

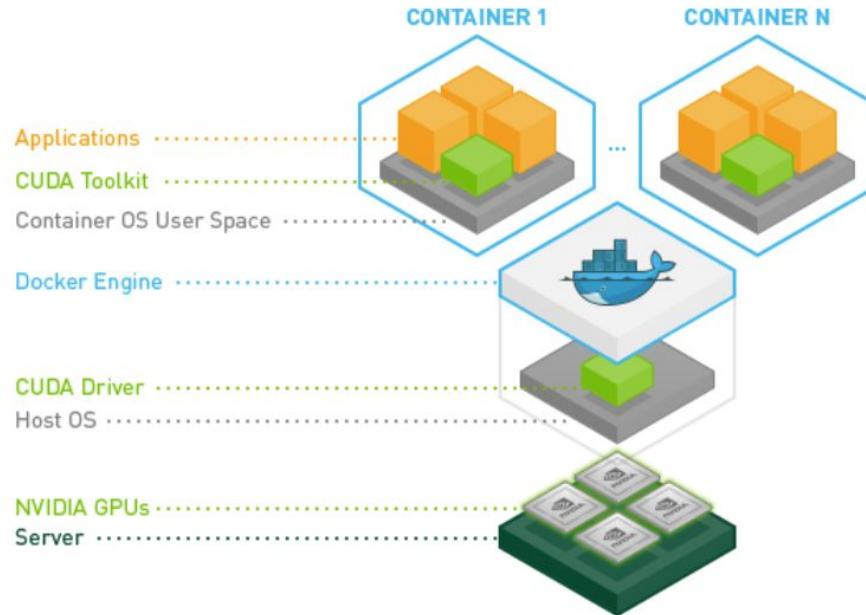
<https://www.docker.com/resources/what-container>



Docker & NVIDIA CUDA

NVIDIA Container Toolkit

license Apache-2.0 documentation wiki packages repository



Introduction

The NVIDIA Container Toolkit allows users to build and run GPU accelerated Docker containers. The toolkit includes a container runtime [library](#) and utilities to automatically configure containers to leverage NVIDIA GPUs.

[INFO]

More info: <https://github.com/NVIDIA/nvidia-docker>



AI in the Cloud

Cloud development → See next lessons...



Options

- Google Colab
- Kaggle
- Paperspace
- Gradient
- Lambda labs
- ...

Advantages:

- You don't have to worry about infrastructure
- Every session is a closed off containerized environment

Disadvantages

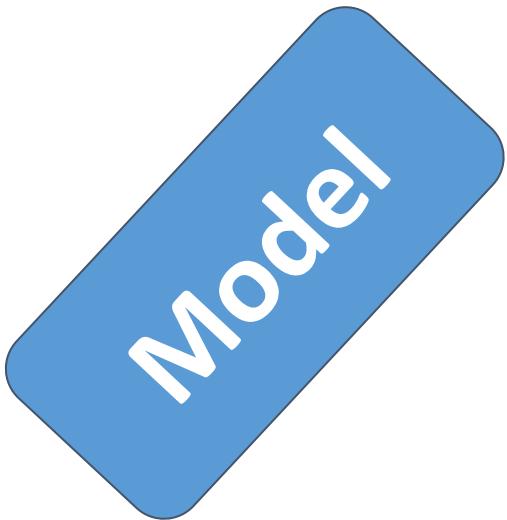
- Dependence on cloud providers
- Limited free options
- Session disconnections
- Shared/limited resources



N e x T

Model deployment

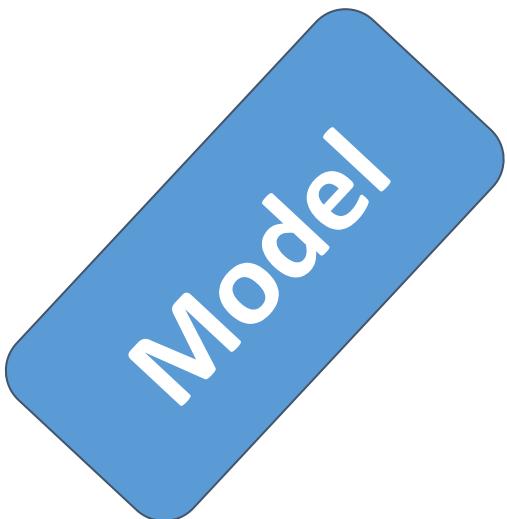
What is a model?



An AI model is a program or algorithm that relies on a set of data to recognize patterns and make predictions or decisions.

In other words it's just some (important) bits. :-)

What is a model? Let's ask ChatGPT!



TI What is inside of an AI model?

2 / 2 An AI model is a mathematical representation of a real-world process or system. It is a set of algorithms that are designed to recognize patterns in data, and make predictions or decisions based on that data.

There are many different types of AI models, and the specific algorithms and techniques used can vary widely depending on the task the model is designed to perform. Some common types of AI models include:

- Supervised learning models: These models are trained on labeled data, which includes both input data and the corresponding correct output. The model is then able to make predictions on new, unseen data by using the patterns it learned from the training data. Examples of supervised learning tasks include classification (predicting which category an input belongs to) and regression (predicting a continuous numerical value).
- Unsupervised learning models: These models are trained on unlabeled data, and are used to discover patterns in the data without being explicitly told what those patterns are. Unsupervised learning tasks include clustering (grouping similar items together) and dimensionality reduction (simplifying the data by identifying underlying patterns).
- Reinforcement learning models: These models learn by interacting with an environment and receiving rewards or punishments based on their actions. They learn to maximize the reward they receive over time by trial and error.

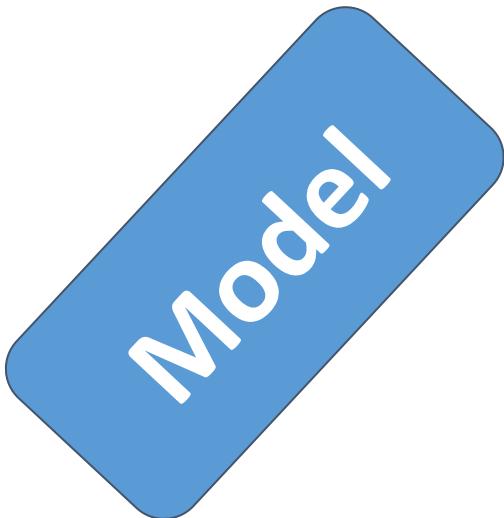
There are many other types of AI models as well, such as generative models, which are used to generate new data that is similar to a given dataset, and deep learning models, which are a type of neural network that is particularly good at learning complex patterns in data.

Regardless of the specific type of model, all AI models consist of a set of mathematical equations that are designed to capture patterns in the data, and make predictions or decisions based on those patterns.

[INFO]

More info: <https://chat.openai.com/chat>

What is a model? Let's ask ChatGPT!



 An AI model can be saved as a .pt file. What is inside such file?

 A .pt file is a file format commonly used to save PyTorch models, which are a type of AI model used for a wide range of tasks, such as image classification, natural language processing, and speech recognition. ↳ ↻

The contents of a .pt file depend on the specific PyTorch model it represents, but in general, a .pt file will contain the model's architecture and its learned parameters.

The model's architecture refers to the structure of the model, including the number and size of the layers, and the connections between them. The learned parameters are the values of the weights and biases that the model has learned during training. These values are what allow the model to make predictions or decisions based on input data.

In addition to the model's architecture and learned parameters, a .pt file may also contain other information, such as the optimizer used to train the model and any other metadata associated with the model.

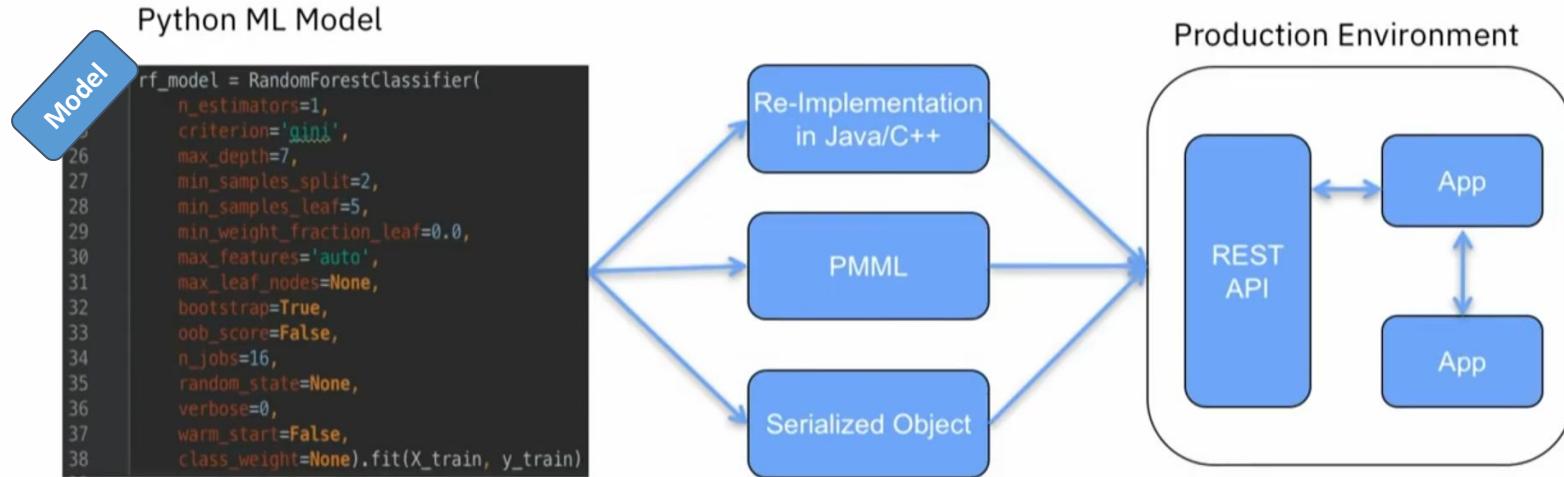
You can load a .pt file in PyTorch using the `torch.load()` function, which will return a PyTorch model that you can then use for tasks such as prediction or evaluation.

[INFO]

More info: <https://chat.openai.com/chat>

Hosting / Serializing

Python ML Model



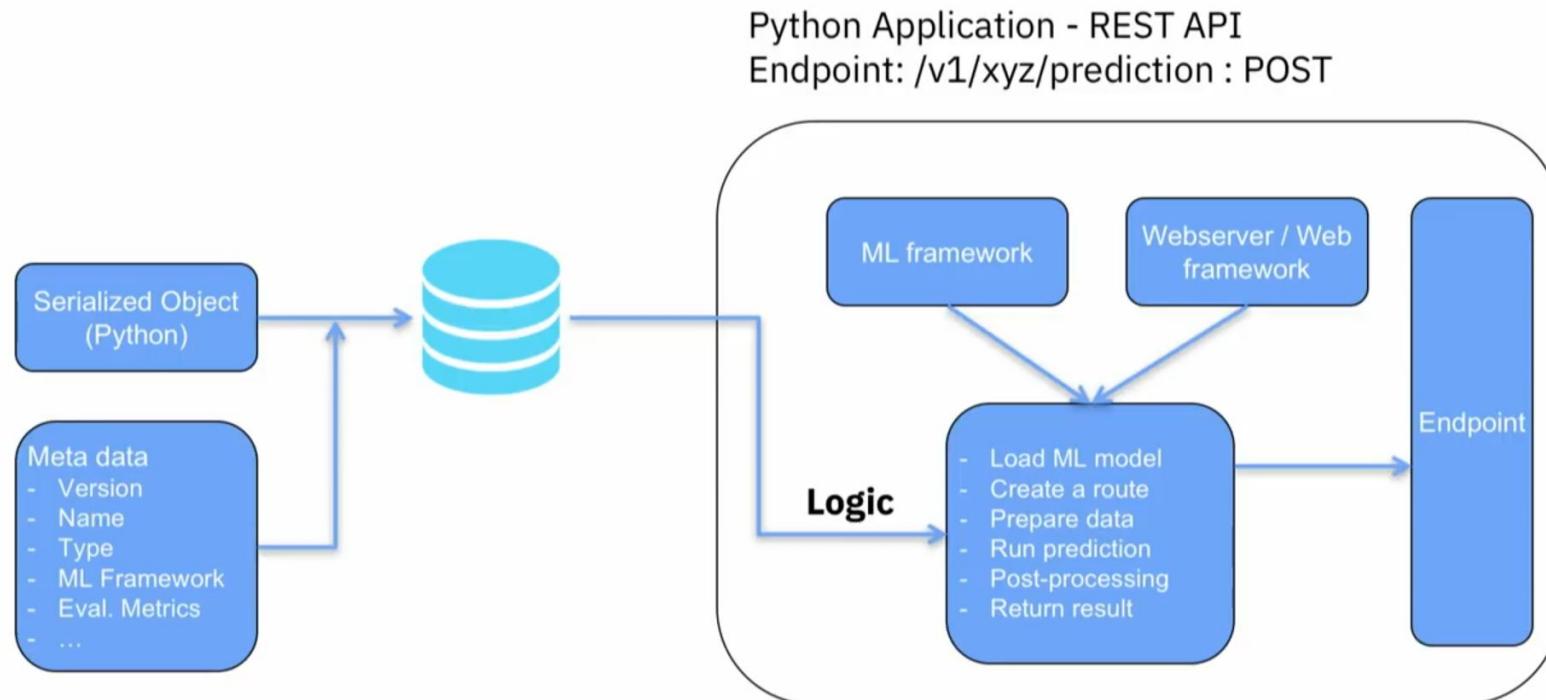
[INFO]

More info: <https://www.youtube.com/watch?v=-UYyyeYJAo>

PMML: Predictive Model Markup Language is an XML-based predictive model interchange format

Hosting / Serializing

Production Environments



Hosting / Serializing

Production Environments – eCommerce



IBM Cloud / Watson and Cloud Platform / © 2018 IBM Corporation

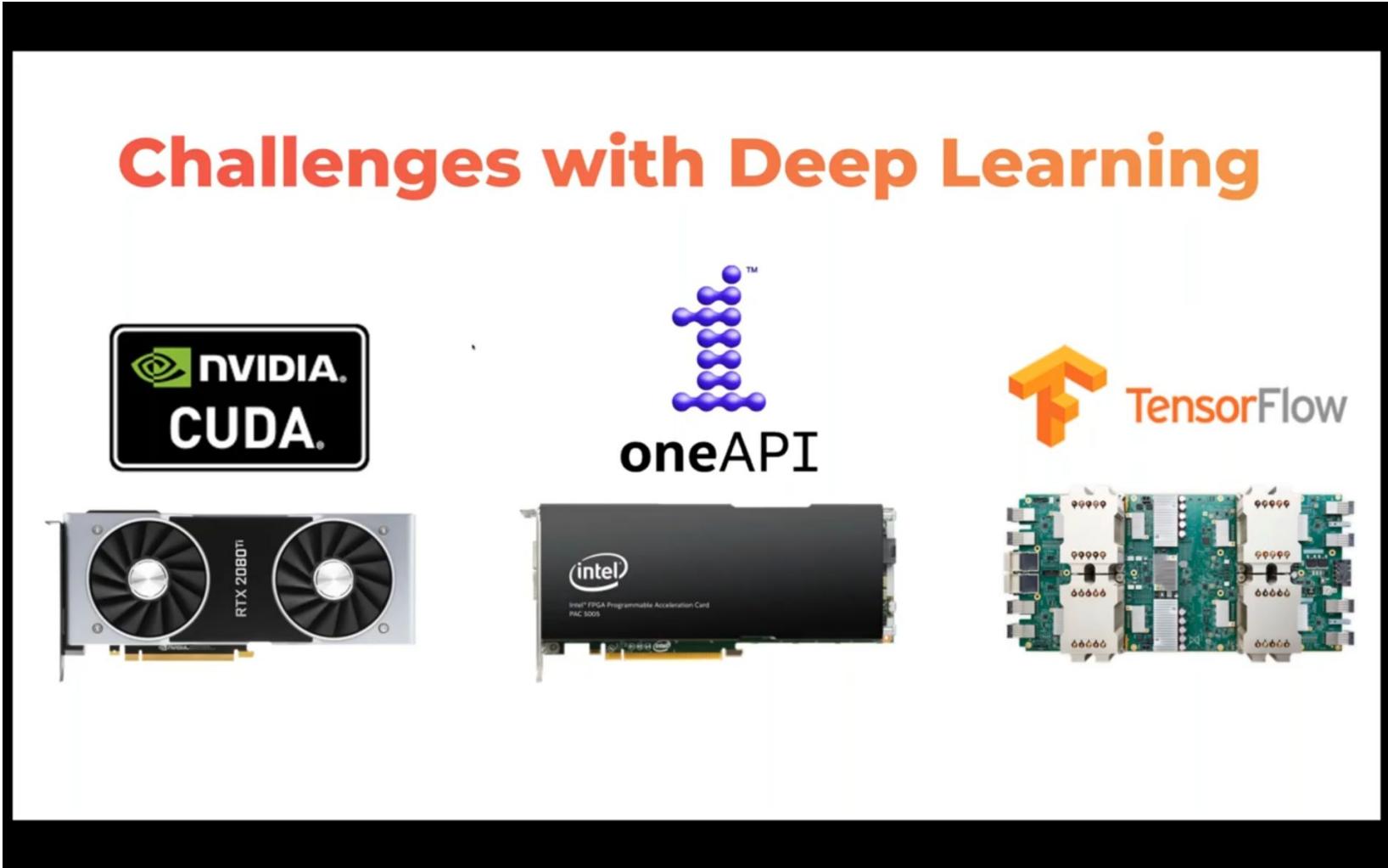
<https://blog.paymill.com/en/keep-e-commerce-system-functioning-optimum-capacity/>,
<http://www.atlascorps.org/blog/customers-and-clients-the-devils-of-business-growth/>

16

[INFO]

More info: <https://www.youtube.com/watch?v=-UYyyeYJAo>

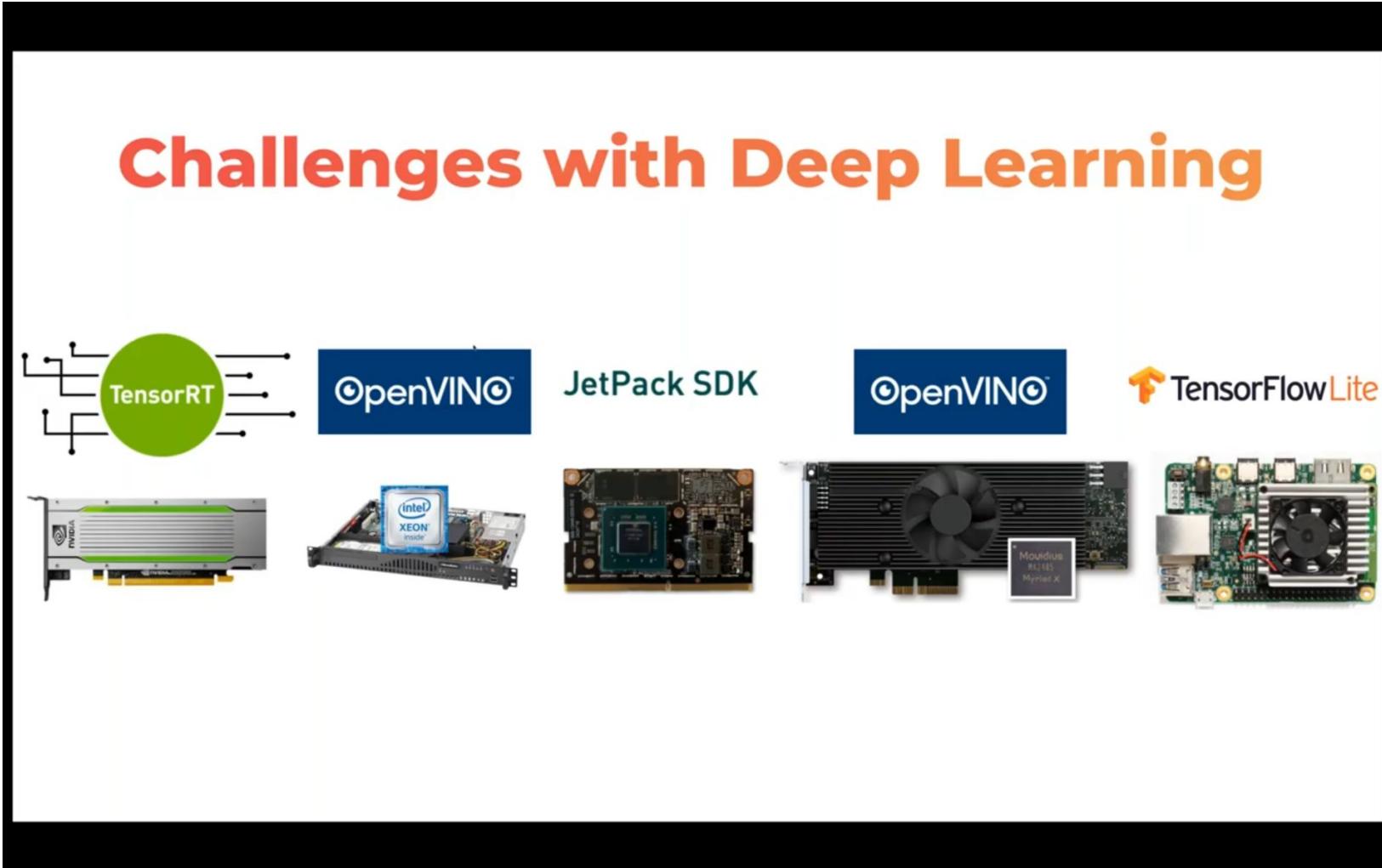
Infrastructure challenges



[INFO]

More info: <https://www.youtube.com/watch?v=cK5AyawZSUI>

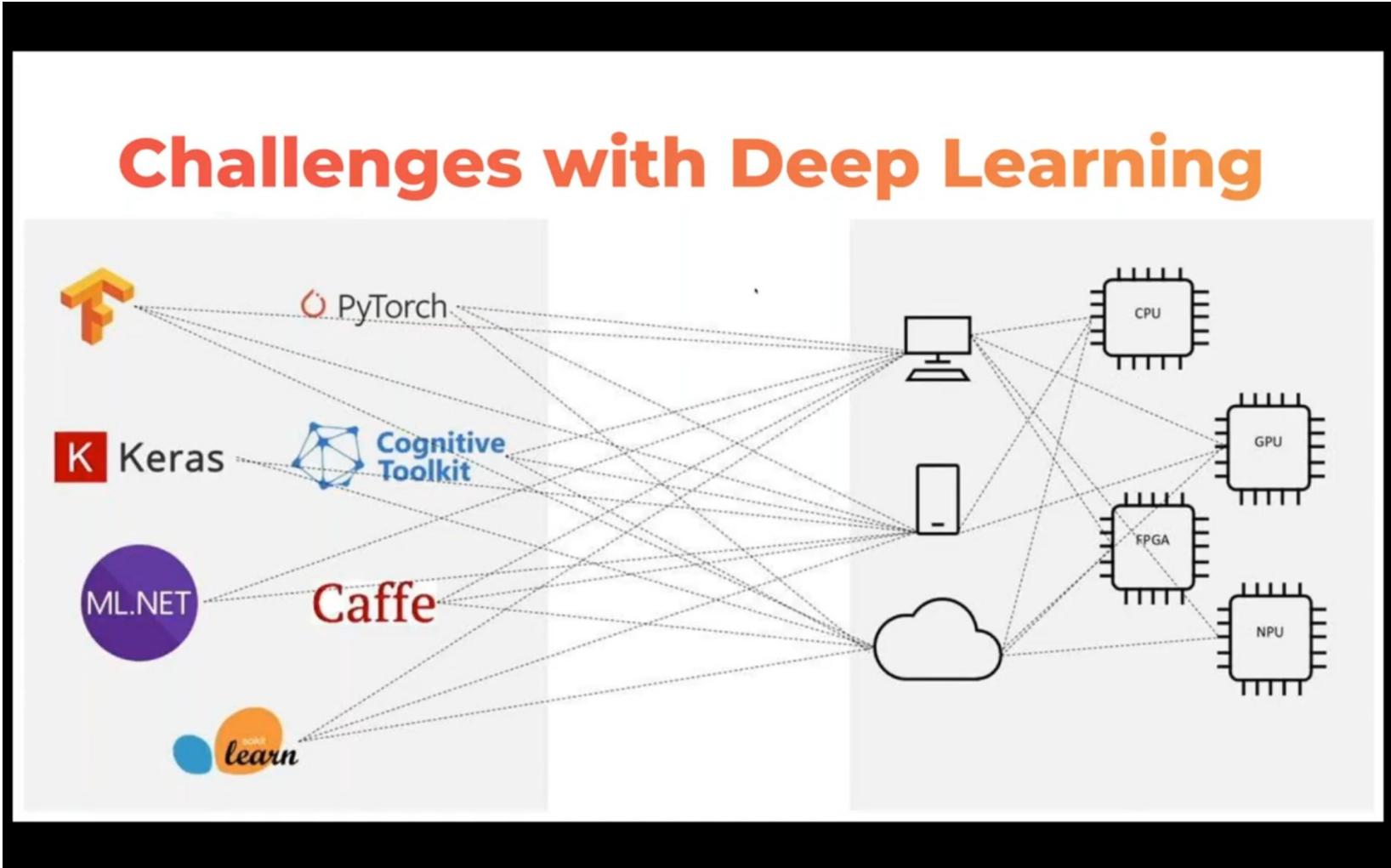
Infrastructure challenges



[INFO]

More info: <https://www.youtube.com/watch?v=cK5AyawZSUI>

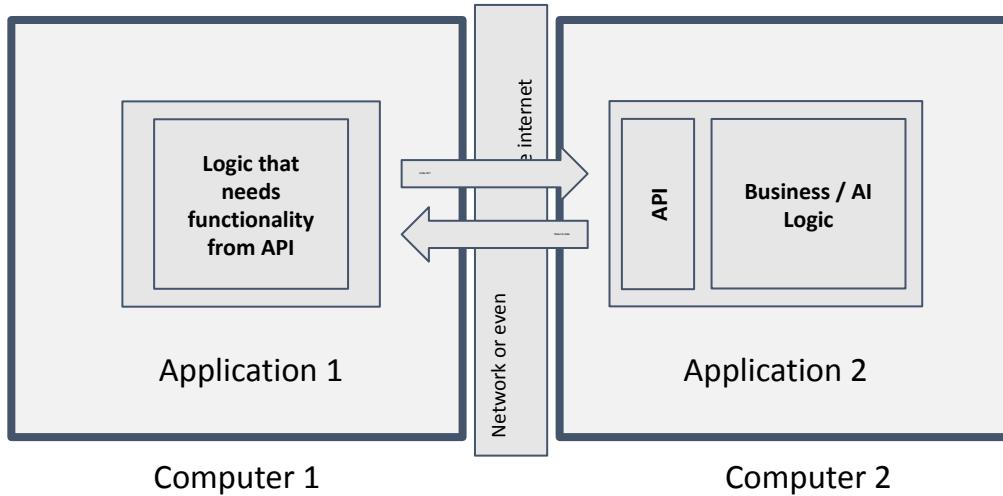
Infrastructure challenges



[INFO]

More info: <https://www.youtube.com/watch?v=cK5AyawZSUI>

Solution: API or Messaging framework



Solution: ONNX?



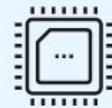
- Open Neural Network eXchange
- The open standard for machine learning interoperability
- File format + runtime



Interoperability

Develop in your preferred framework without worrying about downstream inferencing implications. ONNX enables you to use your preferred framework with your chosen inference engine.

[SUPPORTED FRAMEWORKS >](#)



Hardware Access

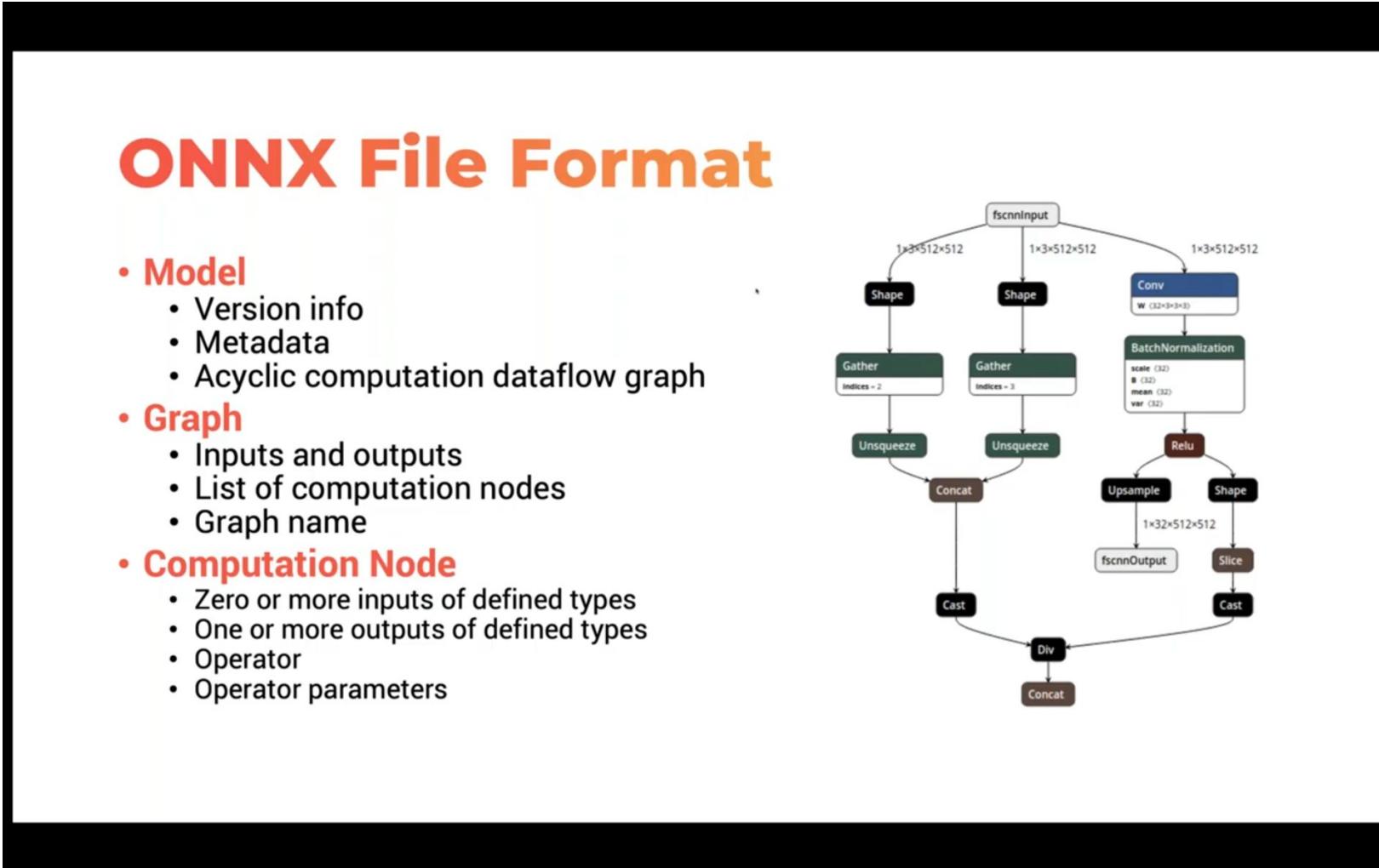
ONNX makes it easier to access hardware optimizations. Use ONNX-compatible runtimes and libraries designed to maximize performance across hardware.

[SUPPORTED ACCELERATORS >](#)

[INFO]

More info: <https://onnx.ai>

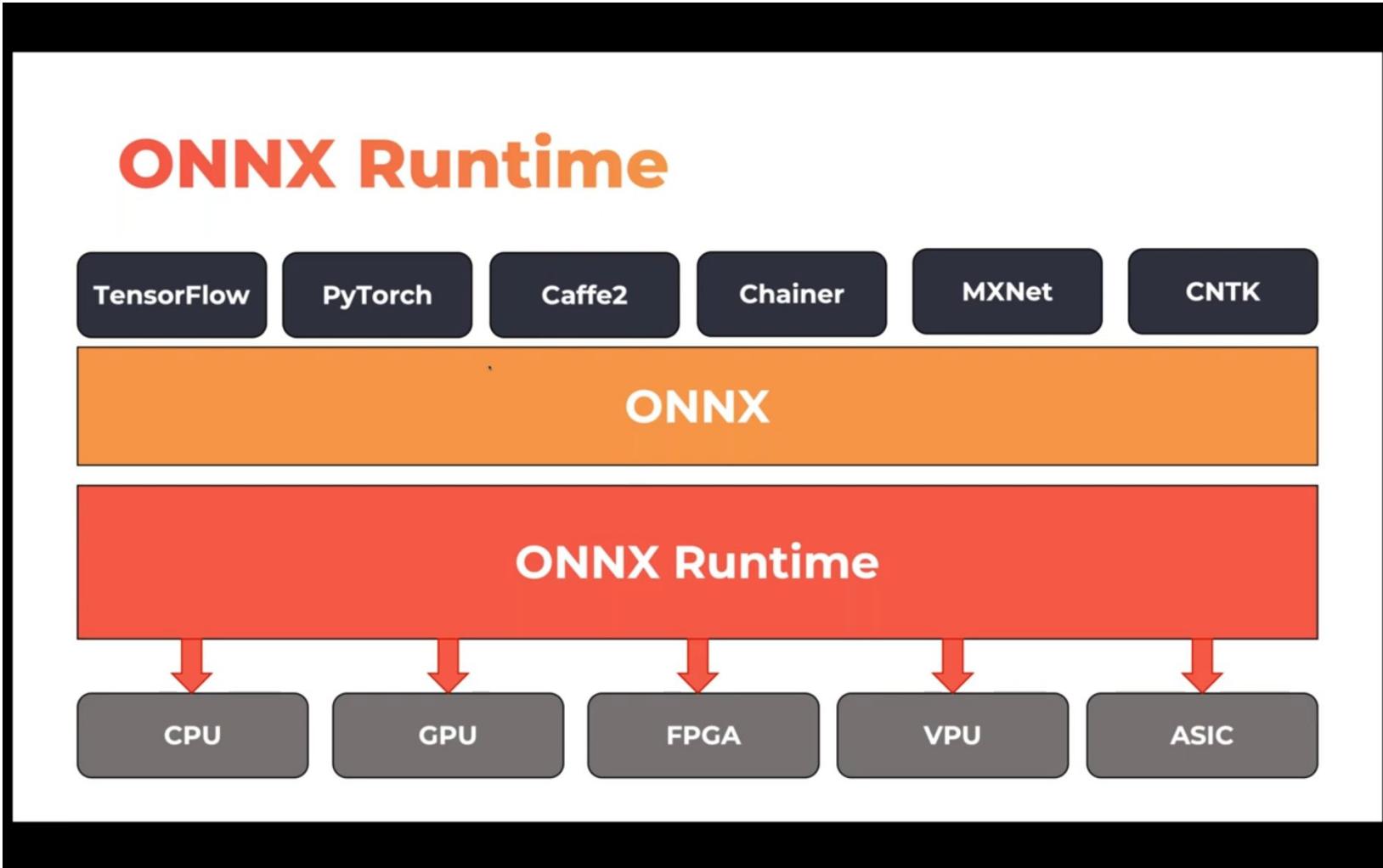
Solution: ONNX?



[INFO]

More info: <https://www.youtube.com/watch?v=cK5AyawZSUI> and <https://onnx.ai>

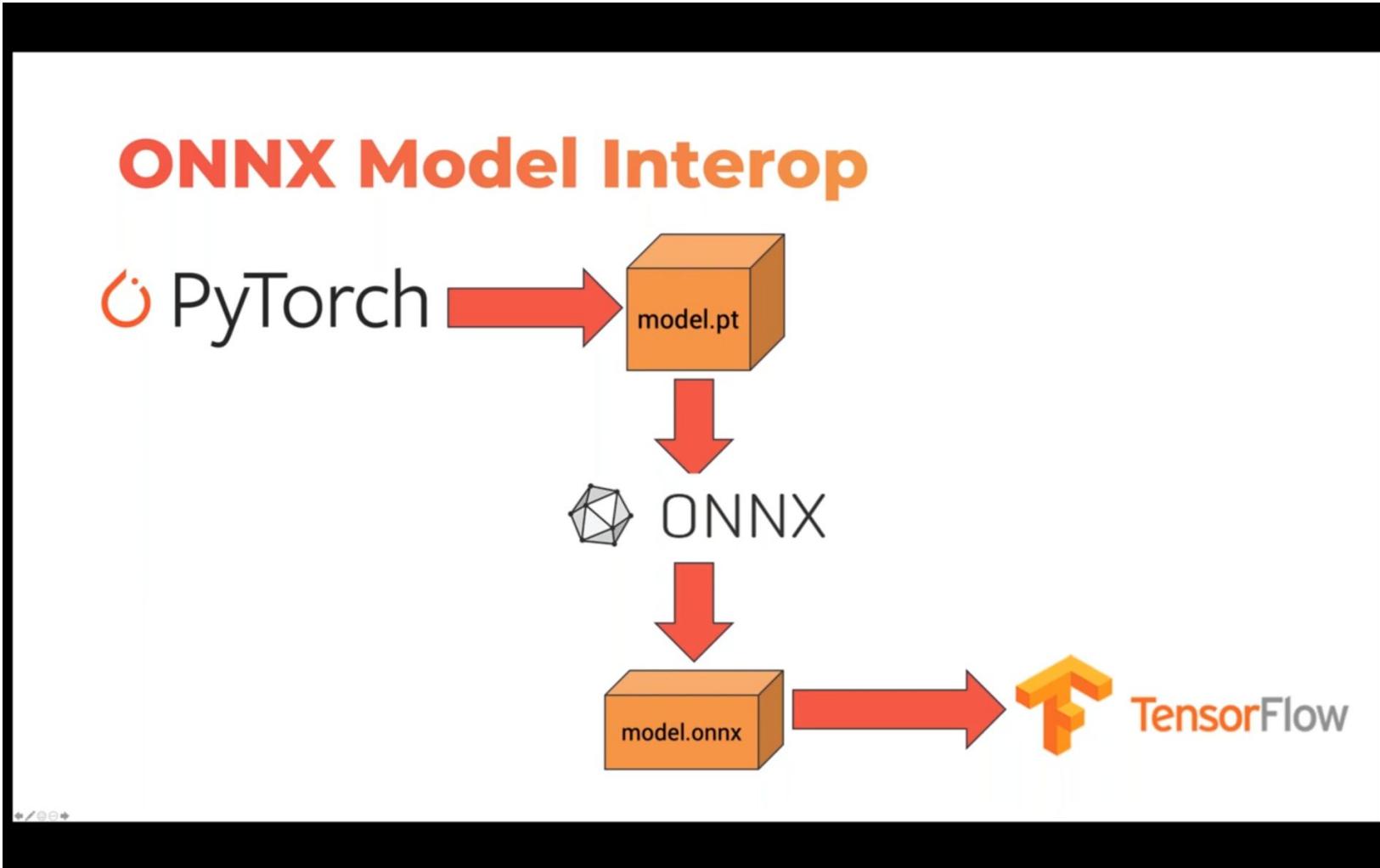
Solution: ONNX?



[INFO]

More info: <https://www.youtube.com/watch?v=cK5AyawZSUI> and <https://onnx.ai>

Solution: ONNX?



[INFO]

More info: <https://www.youtube.com/watch?v=cK5AyawZSUI>

ONNX

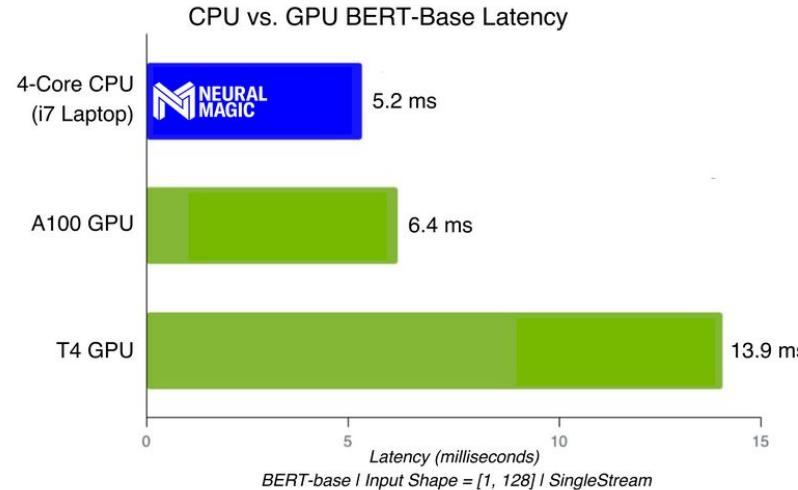


- Fast (with the correct runtime)
- Standardised format
- Multiple runtimes supported
- Portable



- More complicated models: conversion is not straightforward
(Code changes might be needed to support ONNX)
- Performance can be considerably slower (Test this!)
- ONNX Runtime (by Microsoft) has worse support than major libraries
(Run the native library if possible!)
- Not optimized for all hardware, sometimes lacks stable newest CUDA release

Solution for \$\$\$ GPUs: ⚡ DeepSparse?



4-Core CPU (Lenovo Yoga 9 14ITL5) | DeepSparse 1.1.0 | 99% Accuracy | [Replicate Now](#)
A100/T4 NVIDIA GPU | TensorFlow 20.06-py3 NGC | 100% Accuracy | [NVIDIA Numbers](#)

Editions

The DeepSparse Engine is available in two editions:

DeepSparse Community Edition

The Community Edition is open-source and free for evaluation, research, and non-production use.

DeepSparse Enterprise Edition

The Enterprise Edition requires a Trial License or can be fully licensed for production, commercial applications.

- Sparsity-aware neural network inference engine
- GPU-class performance on CPUs
- Loading and deploying **sparse models** with ONNX
- Board support
(Tensorflow, scikit-learn, Apple Core ML, Spark ML, libsvm, XGBoost, PyTorch, ...)
- Zoo website with a lot of models!
(as ONNX or .pt, incl. docs.)



N e x T

Practical example with TF/Keras, Flask & ONNX

File Edit View Insert Runtime Tools Help Last saved at 9:46 AM

Comment Share

+ Code + Text Connect GPU

0. Import statements

```
[ ] import keras
from keras import models
from keras import layers
from keras.datasets import mnist
#from keras.utils import to_categorical
import matplotlib.pyplot as plt
import numpy as np
import tensorflow as tf
import pandas as pd
import warnings

warnings.filterwarnings('ignore')
```

Enable autocomplete (using tab).

```
[ ] %config Completer.use_jedi = False
```

1. Load data

In order to understand what happens in a computer vision model, we first have to understand how computers handle images. We'll use one of the most famous datasets in computer vision, MNIST, for our experiments. MNIST contains images of handwritten digits, collected by the National Institute of Standards and Technology and collated into a machine learning dataset by Yann Lecun and his colleagues. Lecun used MNIST in 1998 in Lenet-5, the first computer system to demonstrate practically useful recognition of handwritten digit sequences. This was one of the most important breakthroughs in the history of AI.

The MNIST database of handwritten digit has a training set of 60,000 examples, and a test set of 10,000 examples. It is a subset of a larger set available from NIST. The digits have been size-normalized and centered in a fixed-size image. It is a good database for people who want to try learning techniques and pattern recognition methods on real-world data while spending minimal efforts on preprocessing and formatting.

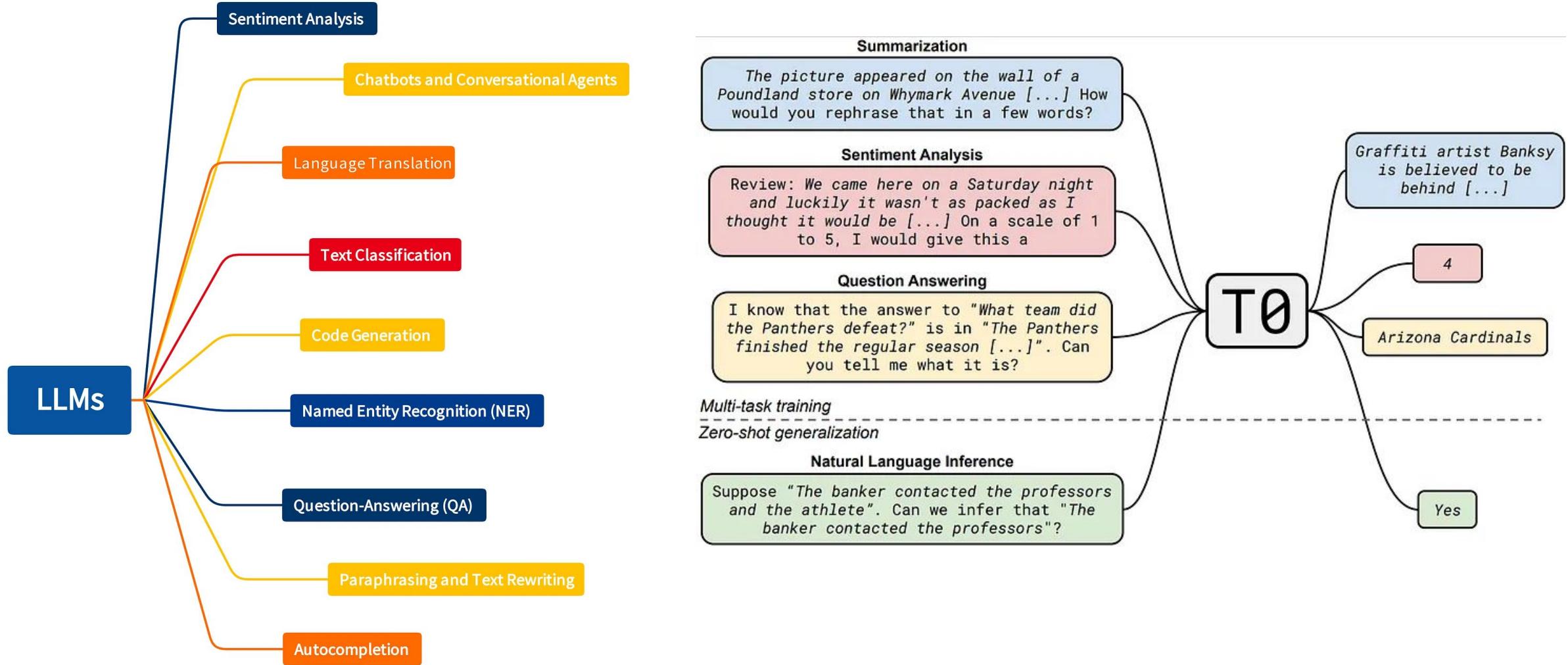
```
[ ] (train_images, train_labels), (test_images, test_labels) = mnist.load_data()

Downloading data from https://storage.googleapis.com/tensorflow/tf-keras-datasets/mnist.npz
11490434/11490434 [=====] - 0s 0us/step
```

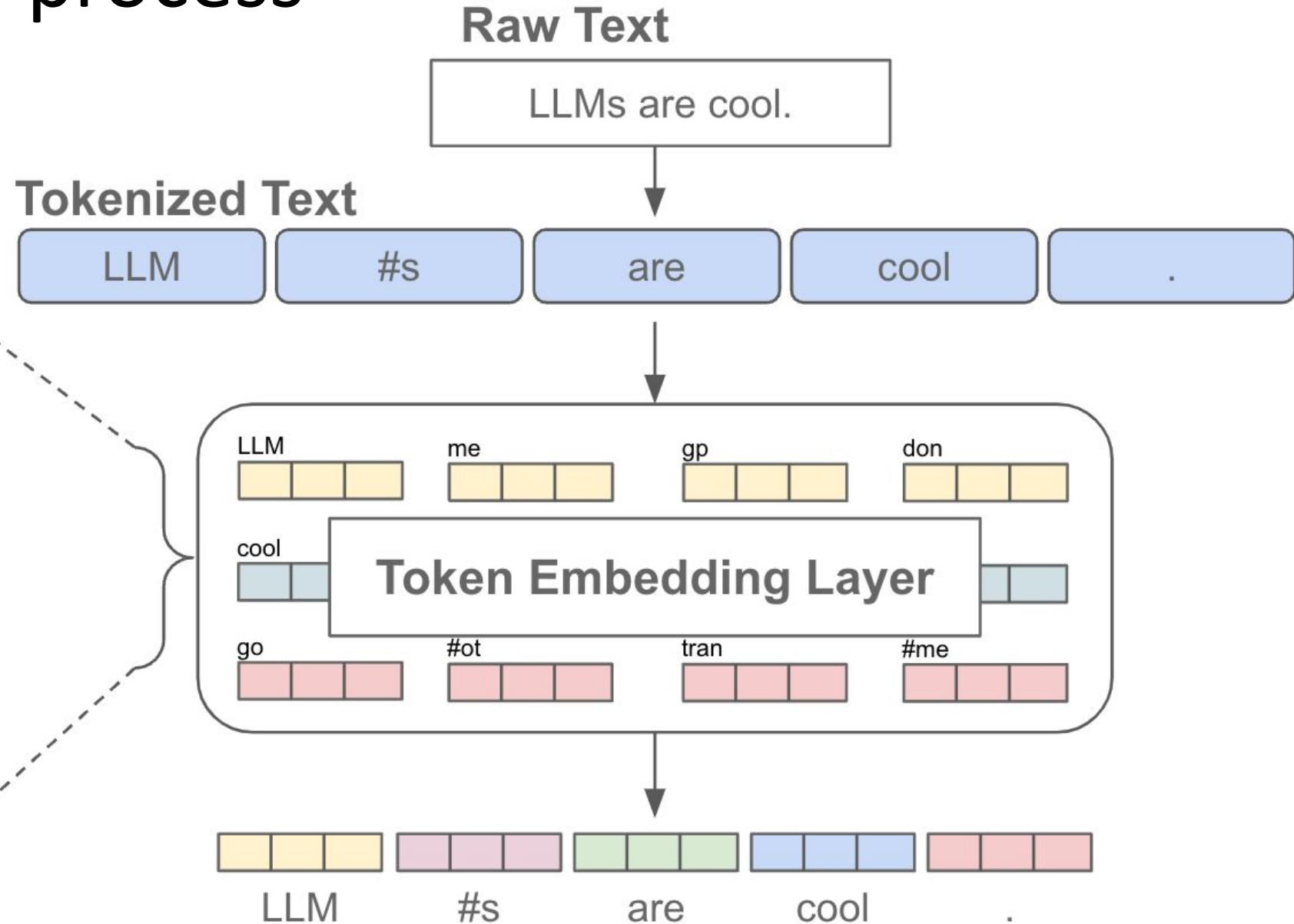
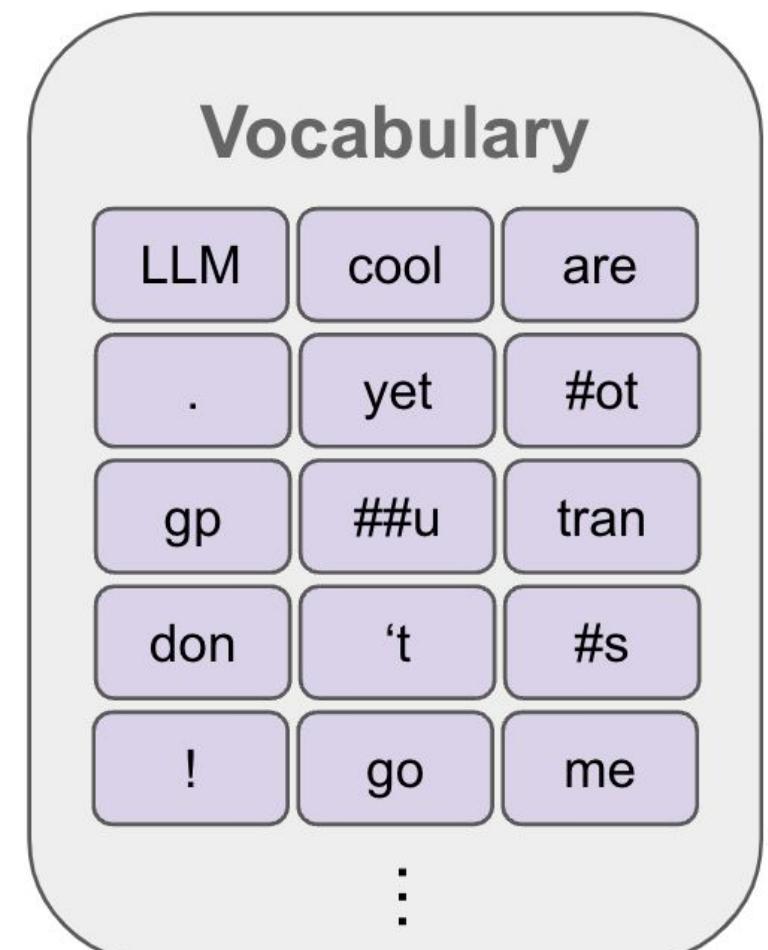


NLP → Large Language Models

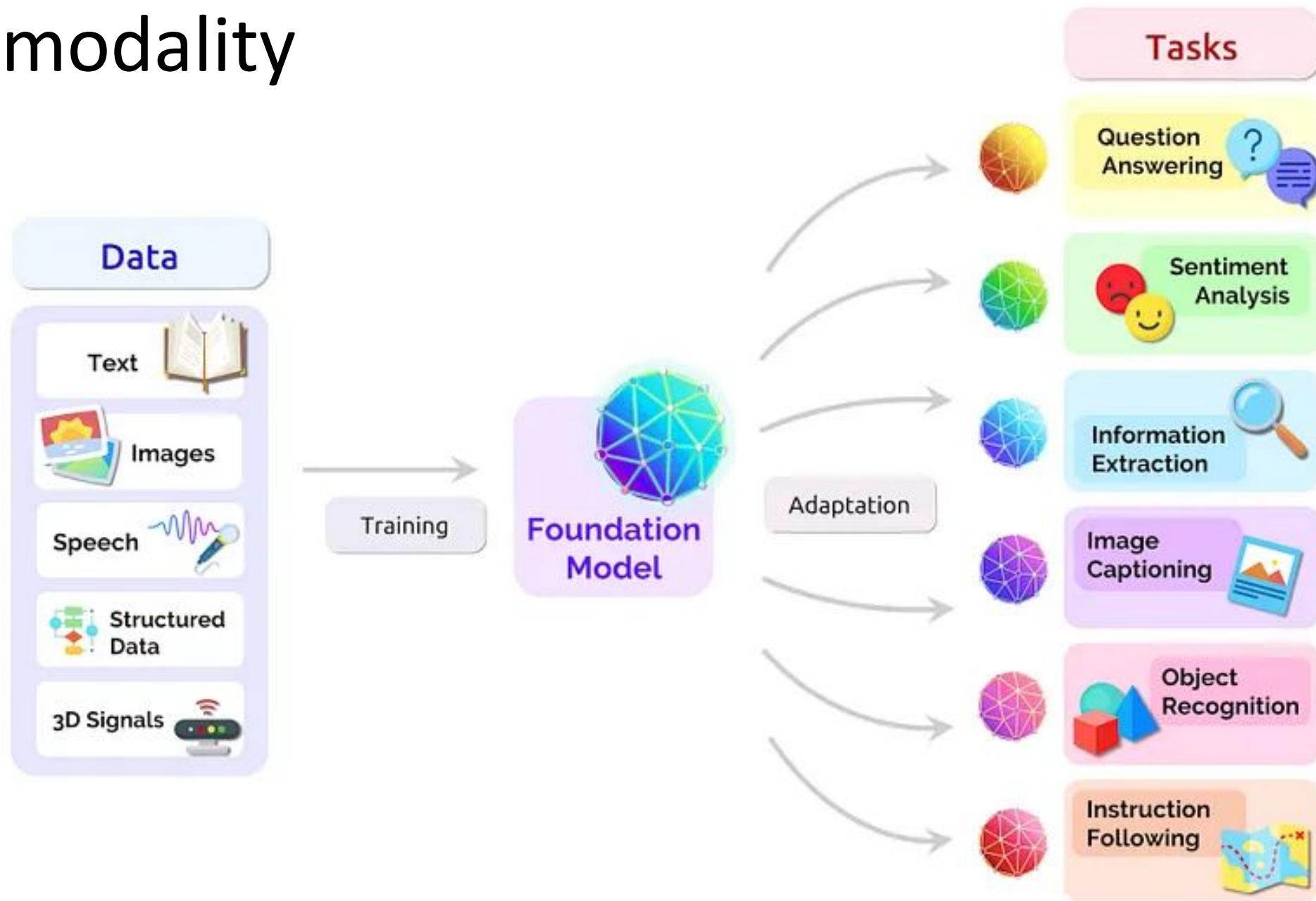
LLMs



Language model process



Multimodality



Prompt engineering

```
[31]: mn = "stabilityai/StableBeluga-7B"
      model = AutoModelForCausalLM.from_pretrained(mn, device_map=0, torch_dtype=torch.bfloat16)
      Loading checkpoint shards:  0%|          | 0/2 [00:00<?, ?it/s]

[32]: sb_sys = """### System:\nYou are Stable Beluga, an AI that follows instructions extremely well. Help as much as you can.\n\n"""

[33]: def mk_prompt(user, syst=sb_sys): return f"{syst}### User: {user}\n\n### Assistant:\n"

[34]: ques = "Who is Jeremy Howard?"

[35]: gen(mk_prompt(ques), 150)

[35]: ['<s> ### System:\nYou are Stable Beluga, an AI that follows instructions extremely well. Help as much as you can.\n\n### User: Who is Jeremy Howard?\n\n### Assistant:\n Jeremy Howard is an Australian entrepreneur, computer scientist, and co-founder of the Machine Learning and Deep Learning startup company, Fast.ai. He is also known for his work in open source software and has co-led the development of several widely used libraries for deep learning and machine learning.</s>']
```

Creating Prompt Template

After loading the instruction dataset, we will define the `create_prompt_formats` function to create a prompt template against each prompt in our dataset and save it in a new dictionary key `text` for further data preprocessing and fine-tuning.

```
def create_prompt_formats(sample):
    """
    Creates a formatted prompt template for a prompt in the instruction dataset

    :param sample: Prompt or sample from the instruction dataset
    """

    # Initialize static strings for the prompt template
    INTRO_BLURB = "Below is an instruction that describes a task. Write a response that appropriately completes the request."
    INSTRUCTION_KEY = "### Instruction:"
    INPUT_KEY = "Input:"
    RESPONSE_KEY = "### Response:"
    END_KEY = "### End"

    # Combine a prompt with the static strings
    blurb = f"{INTRO_BLURB}"
    instruction = f"{INSTRUCTION_KEY}\n{sample['instruction']}"
    input_context = f"{INPUT_KEY}\n{sample['input']}" if sample["input"] else None
    response = f"{RESPONSE_KEY}\n{sample['output']}"
    end = f"{END_KEY}"

    # Create a list of prompt template elements
    parts = [part for part in [blurb, instruction, input_context, response, end] if part]

    # Join prompt template elements into a single string to create the prompt template
    formatted_prompt = "\n\n".join(parts)

    # Store the formatted prompt template in a new key "text"
    sample["text"] = formatted_prompt

    return sample
```

```
create_prompt_formats(dataset[randrange(len(dataset))])
```

```
{'instruction': 'Categorize the news article into one of the 18 categories:\n\nWORLD NEWS\nCOMEDY\nPOLITICS\nTECH\nSPORTS\nBUSINESS\nOTHERS\nENTERTAINMENT\nCULTURE & ARTS\nFOOD & DRINK\nMEDIA\nRELIGION\nMONEY\nHEALTHY LIVING\nSCIENCE\nEDUCATION\nCRIME\nENVIRONMENT\n\n',
 'input': '--- \n\n[Nation] A court has declined to stop the replacement of Ms Grace Kaindi as deputy inspector-general of police. \n(c) AllAfrica News: Kenya – Read entire story here .',
 'output': 'CRIME',
 'text': 'Below is an instruction that describes a task. Write a response that appropriately completes the request.\n\n### Instruction:\n\nCategorize the news article into one of the 18 categories:\n\nWORLD NEWS\nCOMEDY\nPOLITICS\nTECH\nSPORTS\nBUSINESS\nOTHERS\nENTERTAINMENT\nCULTURE & ARTS\nFOOD & DRINK\nMEDIA\nRELIGION\nMONEY\nHEALTHY LIVING\nSCIENCE\nEDUCATION\nCRIME\nENVIRONMENT\n\n\nInput:\n--- \n\n[Nation] A court has declined to stop the replacement of Ms Grace Kaindi as deputy inspector-general of police. \n(c) AllAfrica News: Kenya – Read entire story here .'}
```

Limitations

SA

You

how do I run llama.cpp?



ChatGPT

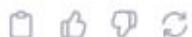
To run a C++ file named `llama.cpp`, you would typically follow these steps:

1. **Compile the C++ File:** Use a C++ compiler like `g++` or `clang++` to compile the file. The command might look something like `g++ llama.cpp -o llama` which will compile

`llama.cpp` and create an executable named `llama`

2. **Run the Executable:** After compiling, you run the executable on a Unix-like system (like Linux or macOS) or `llama`

This assumes you have a C++ compiler installed on your system and you have successfully compiled the C++ file without errors. If there are dependencies or libraries required by the code, you might need to include them in the compilation process.



LLMs don't know what they don't know!

Description

The main goal of `llama.cpp` is to run the LLaMA model using 4-bit integer quantization on a MacBook Pro M1 chip.

- Plain C/C++ implementation without dependencies
- Apple silicon first-class citizen - optimized via ARM NEON, Accelerate and Metal frameworks
- AVX, AVX2 and AVX512 support for x86 architectures
- Mixed F16 / F32 precision
- 2-bit, 3-bit, 4-bit, 5-bit, 6-bit and 8-bit integer quantization support
- CUDA, Metal and OpenCL GPU backend support

Limitations

Lost in the Middle: How Language Models Use Long Contexts

Nelson F. Liu^{1*} Kevin Lin² John Hewitt¹ Ashwin Paranjape³

Michele Bevilacqua³ Fabio Petroni³ Percy Liang¹

¹Stanford University

²University of California, Berkeley

³Samaya AI

nfliu@cs.stanford.edu

Abstract

While recent language models have the ability to take long contexts as input, relatively little is known about how well they *use* longer context. We analyze language model performance on two tasks that require identifying relevant information within their input contexts: multi-document question answering and key-value retrieval. We find that performance is often highest when relevant information occurs at the beginning or end of the input context, and significantly degrades when models must access relevant information in the middle of long contexts. Furthermore, performance substantially decreases as the input context grows longer, even for explicitly long-context models. Our analysis provides a better understanding of how language models use their input context and provides new evaluation protocols for future long-context models.

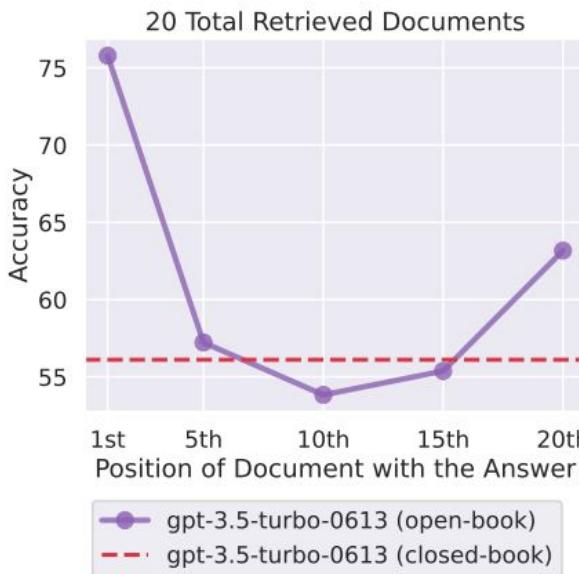
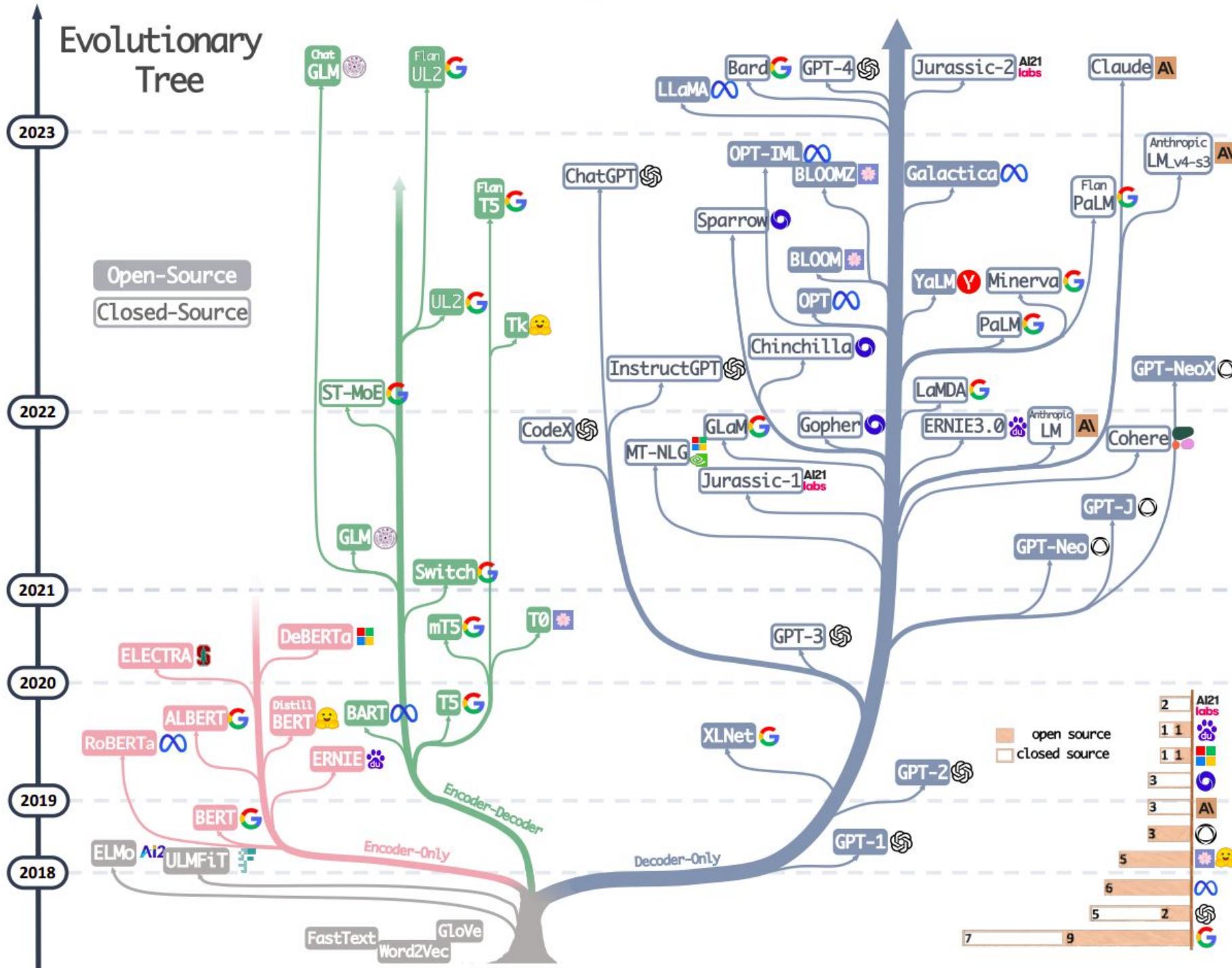


Figure 1: Changing the location of relevant information (in this case, the position of the passage that answers an input question) within the language model’s input context results in a U-shaped performance curve—models are better at using relevant information that occurs at the very beginning or end of its input context, and performance degrades significantly when models must access and use information located in the middle of its input context. For example, GPT-3.5-Turbo’s open-book performance on the multi-document question task when relevant information is placed in the middle of its input context is lower than its performance when predicting *without any documents* (i.e., the closed-book setting; 56.1%). See Figure 5 for full results.

Evolutionary Tree



Benchmarks

- FastEval
- HuggingFace leaderboard
- (Super)GLUE
- SQuAD (Stanford Question Answering Dataset)
- MultiNLI (Multi-Genre Natural Language Inference)
- SWAG (Situations With Adversarial Generations)

Open LLM Leaderboard

The Open LLM Leaderboard aims to track, rank and evaluate open LLMs and chatbots.
Submit a model for automated evaluation on the GPU cluster on the "Submit" page! The leaderboard's backend runs the great [Eleuther AI Language Model Evaluation Harness](#) - read more details in the "About" page!

The screenshot shows the Open LLM Leaderboard interface. It includes a search bar, filter options for model types (pretrained, fine-tuned, instruction-tuned, RL-tuned), precision levels (torch.float16, torch.bfloat16, torch.float32, 8bit, 4bit, GPTQ), and model sizes (in billions of parameters). The main table lists various models with their names, average scores, and specific metrics like ARC, HellaSwag, MMLU, and TruthfulQA.

Model	Average	ARC	HellaSwag	MMLU	TruthfulQA
ValiantLabs/ShiningValiant	74.17	72.95	87.88	70.97	64.88
TCBU-NPU/FashionGPT-70B-V1.2	74.11	73.04	88.15	70.11	65.15
sequelbox/StellarBright	74.1	72.95	87.82	71.17	64.46
Riiid/sheep-duck-llama-2-70b-v1.1	74.07	73.04	87.81	70.84	64.58
AIDC-ai-business/Marcoroni-70B-v1	74.06	73.55	87.62	70.67	64.41
TCBU-NPU/FashionGPT-70B-V1.1	74.05	71.76	88.2	70.99	65.26
adonlee/LLaMA_2_70B_LoRA	73.9	72.7	87.55	70.84	64.52
uni-tianyan/Uni-TianYan	73.81	72.1	87.4	69.91	65.81
Riiid/sheep-duck-llama-2	73.69	72.35	87.78	70.82	63.8
Riiid/sheep-duck-llama-2	73.67	72.27	87.78	70.81	63.8

Click on the links for more details!

Rank	Size	Model	Total	MT-Bench	CoT	Python Code	LM-Eval
1		GPT-4-0613	77.78	8.89	0.71	0.66	
2		GPT-3.5-Turbo-0613	65.17	8.22	0.52	0.53	
3		GPT-3.5-Turbo-0301	63.67	8.11	0.48	0.55	
4	70B	WizardLM 70B V1.0	56.38	7.68	0.47	0.27	72.23
5	70B	Stable Beluga 2	55.60	7.42	0.47	0.29	
6	70B	OpenAssistant Llama-2 70B V10	55.01	7.13	0.47	0.31	74.89
7	70B	WizardMath 70B V1.0	51.78	6.86	0.49	0.16	
8	70B	LLAMA-2 70B Chat	48.18	7.10	0.36	0.20	71.54
9	34B	WizardCoder Python 34B V1.0	47.15	6.30	0.29	0.49	
10	65B	Guanaco-65B	44.32	6.63	0.33	0.15	73.92
11	13B	Open-Assistant Code-Llama 13B V10	43.98	6.00	0.29	0.37	62.03
12	33B	Vicuna 33B V1.3	43.04	6.97	0.28	0.11	70.35
13	7B	Mistral-7B-Instruct-v0.1	42.66	6.69	0.28	0.15	
14	13B	Nous-Hermes Llama-2 13B	42.51	6.54	0.29	0.16	70.03
15	13B	OpenOrca x OpenChat - Preview2 - 13B	42.45	6.38	0.31	0.15	72.42
16	13B	OpenOrca-Platypus2-13B	41.41	6.49	0.30	0.09	72.22
17	30B	MPT-30B-Chat	41.24	6.45	0.26	0.16	69.45
18	13B	StableBeluga-13B	41.05	6.19	0.30	0.14	71.31
19	13B	LLAMA-2 13B Chat	40.94	6.66	0.26	0.11	68.48
20	40B	Open-Assistant Falcon-40B SFT-Mix	40.30	6.16	0.28	0.16	74.41
21	13B	Dolphin Llama 13B	39.76	6.04	0.31	0.07	67.12
22	13B	Nous-Hermes-13B	38.49	6.23	0.24	0.12	68.84
23	40B	Open-Assistant Falcon-40B SFT-Top1	38.25	6.30	0.22	0.12	73.95
24	13B	WizardMath 13B V1.0	38.03	5.52	0.34	0.05	
25	13B	WizardCoder Python 13B V1.0	37.09	5.01	0.21	0.40	
26	7B	LLAMA-2 7B Chat	35.36	6.07	0.21	0.05	65.23
27	7B	Vicuna 7B V1.3	34.50	6.11	0.18	0.06	65.83
28	40B	Falcon-40B-Instruct	31.75	5.21	0.19	0.09	72.67
29	7B	MPT-7B-Chat	30.97	5.34	0.15	0.10	64.76
30	12B	Open-Assistant Pythia-12B SFT-8	28.41	5.12	0.13	0.07	60.30
31	15B	WizardCoder-15B	27.48	3.56	0.14	0.38	48.44
32	12B	Open-Assistant Pythia-12B SFT-1	23.30	4.28	0.11	0.03	58.83
33	7B	Falcon-7B-Instruct	21.78	3.88	0.12	0.01	62.50



NLP → Large Language Models

Tools



AutoGPT

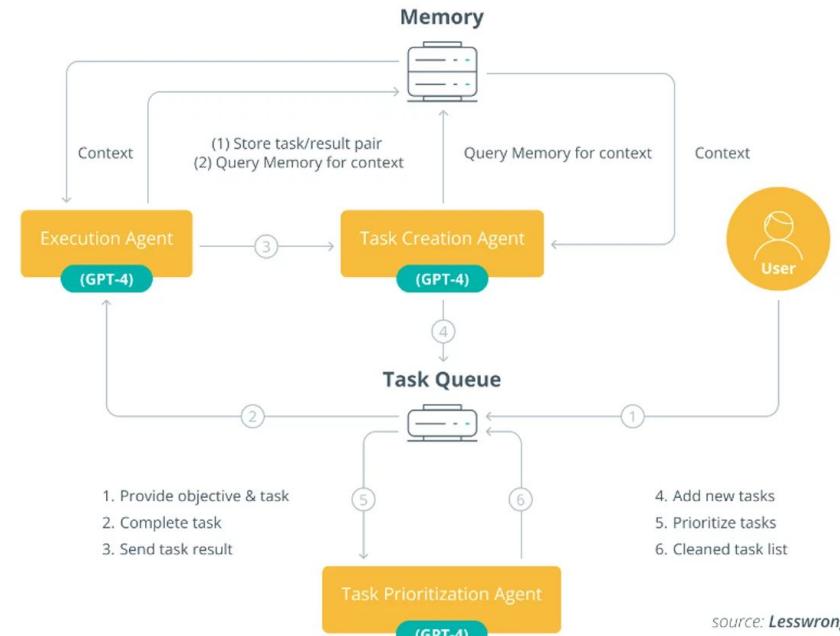
- Experimental, open-source "AI agent" (Python application)
Given a goal in natural language will attempt to achieve it
 - by breaking it into sub-tasks
 - and using the internet and other tools
 - in an automatic loop (with interaction)
- On top of OpenAI models

```
Welcome to Auto-GPT! run with '--help' for more information.
Create an AI-Assistant: Enter the name of your AI and its role below. Entering nothing will load defaults.
Name your AI: For example, 'Entrepreneur-GPT'
AI Name: MikesNutritionist
MikesNutritionist here! I am at your service.
Describe your AI's role: For example, 'an AI designed to autonomously develop and run businesses with the sole goal of increasing your net worth.'
MikesNutritionist is: create a meal plan for this week
Enter up to 5 goals for your AI: For example: Increase net worth, Grow Twitter Account, Develop and manage multiple businesses autonomously'
Enter nothing to load defaults, enter nothing when finished.
Goal 1: Create a 7 day meal plan for me
Goal 2: Write the recipes for each dinner in diet.txt
Goal 3: When seven meals are added to the file, stop
Goal 4:
Using memory of type: LocalCache
Using Browser: chrome
Thinking...
```

[INFO]

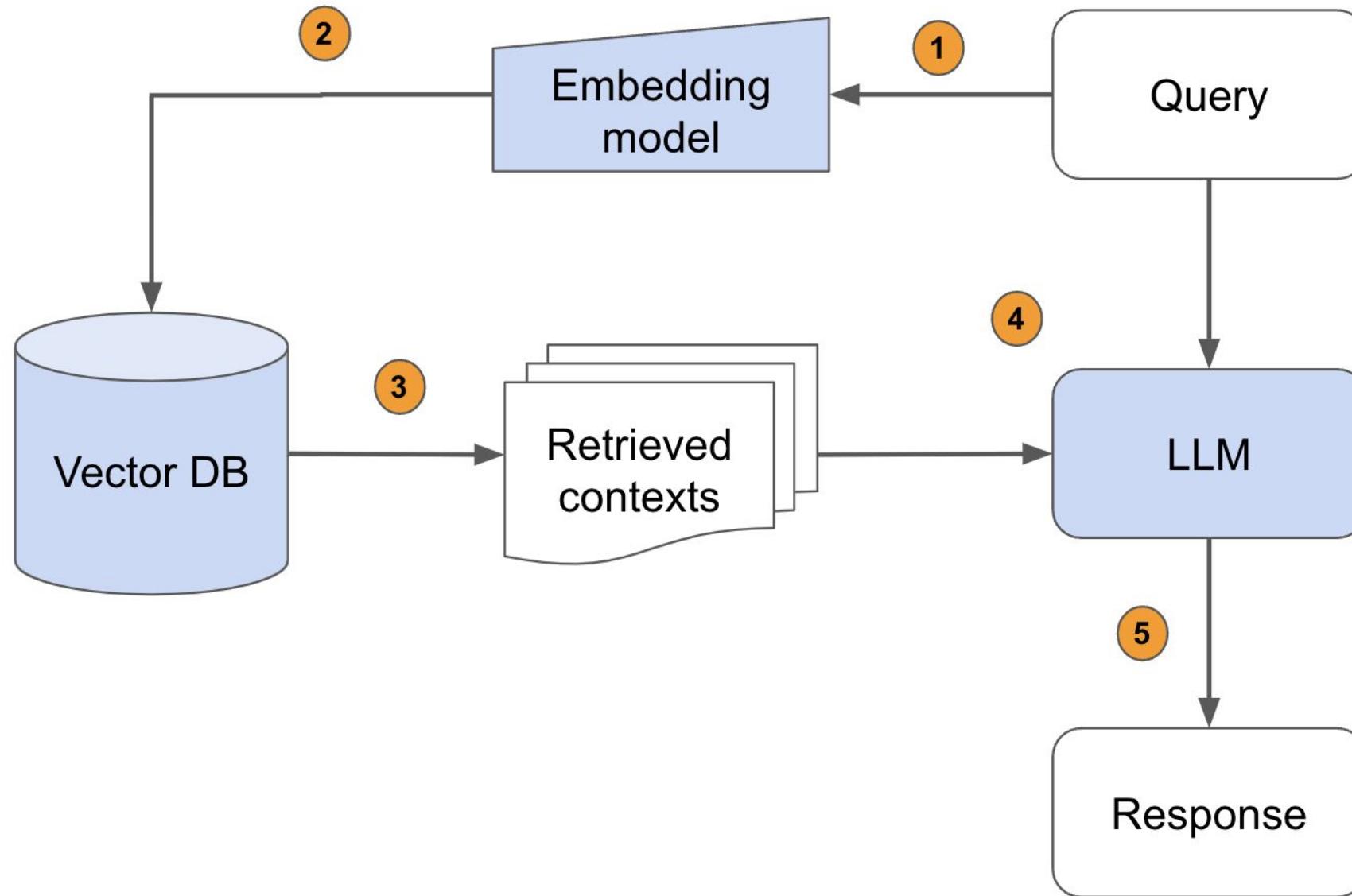
More info: <https://news.agpt.co> and <https://github.com/Significant-Gravitas/AutoGPT>

Working of Auto-GPT

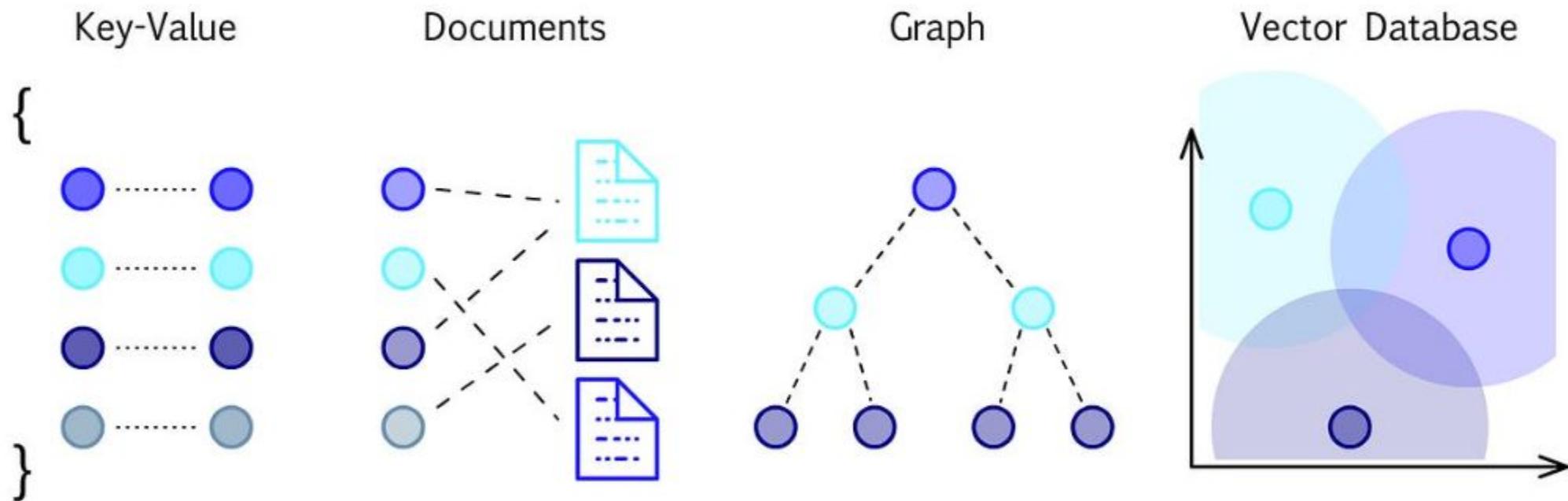


SOURCE: Lesswrong

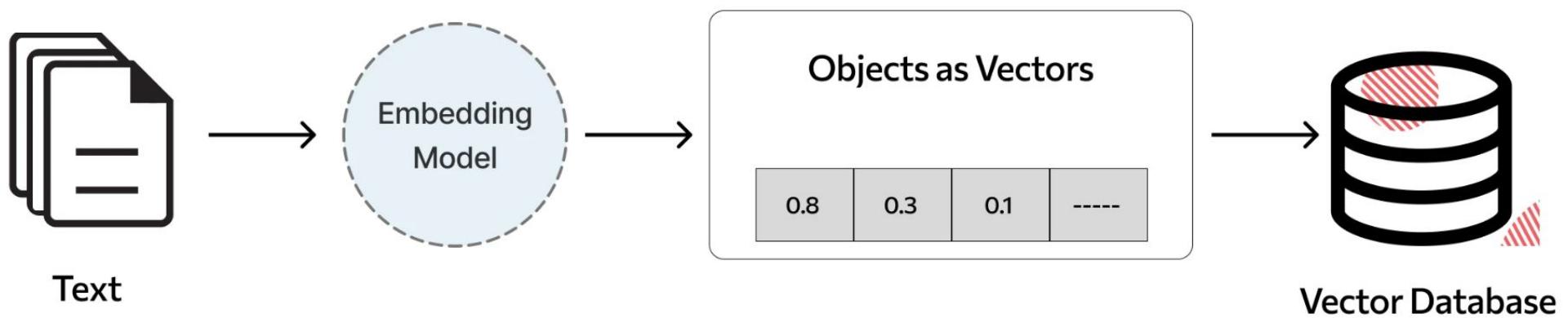
Retrieval Augmented Generation (RAG)



DB Types



Vector DB

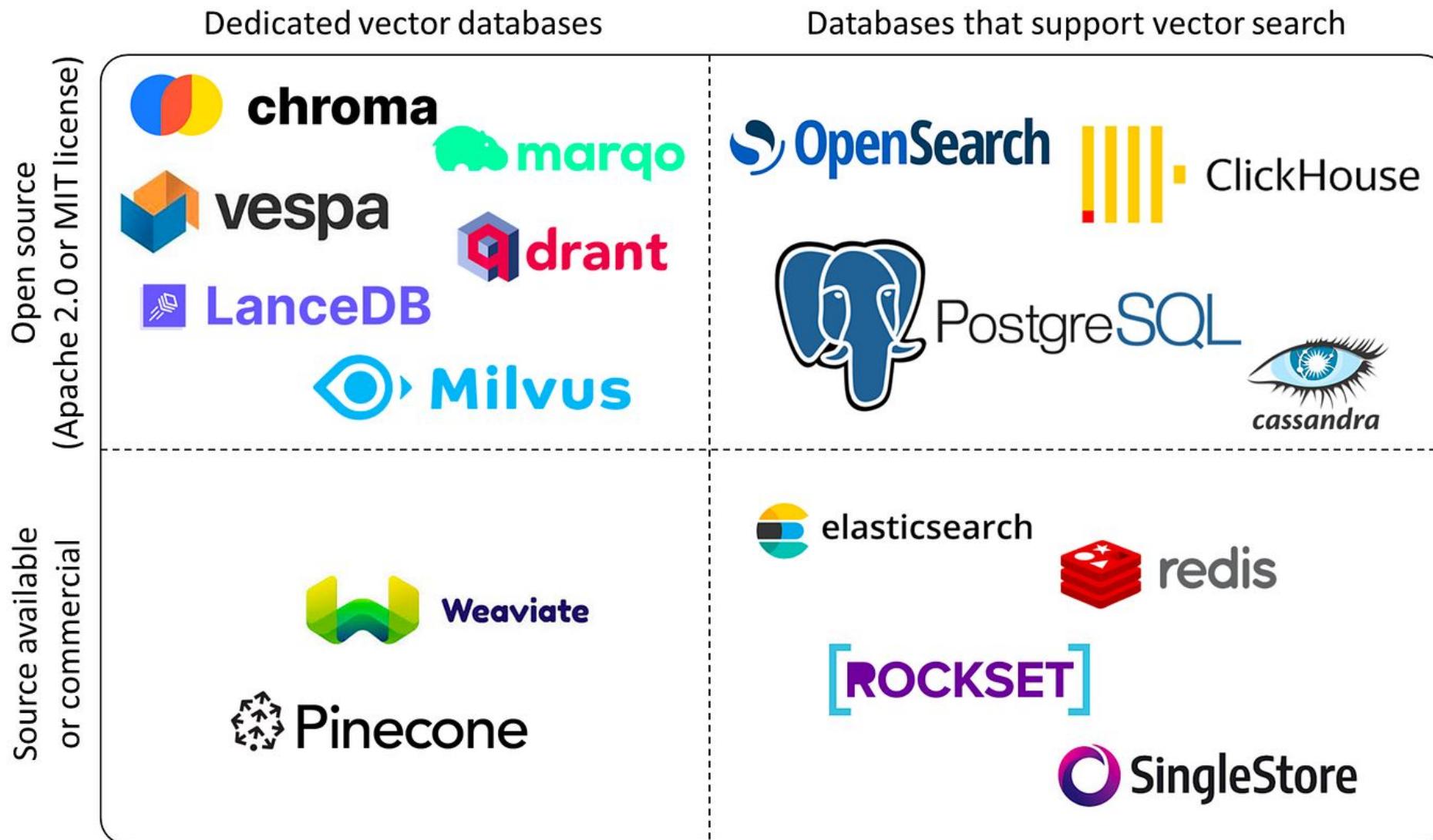


graft

Vector DB



Vector DBs



LangChain

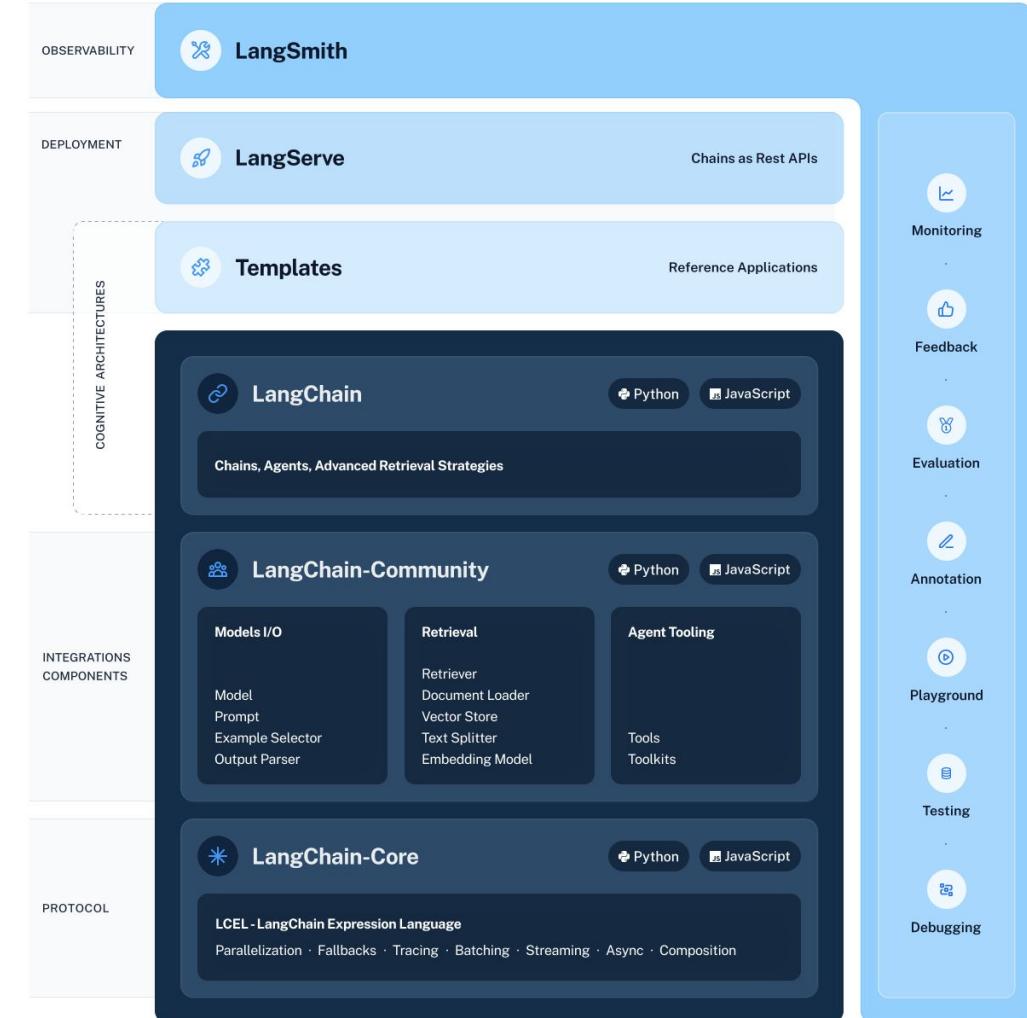


LangChain

- Open Source framework (MIT)
- For applications powered by language models.
- LangChain Libraries (Python & JS)

The Python and JavaScript libraries.
(interfaces and integrations, a basic run time
for combining these components,
and implementations of chains and agents.)

- LangChain Templates
Easily deployable reference architectures
- LangServe
Library for deploying LangChain chains as a REST API
- LangSmith
A developer platform to debug, test, evaluate,
and monitor chains built on any LLM framework and
seamlessly integrates with LangChain.



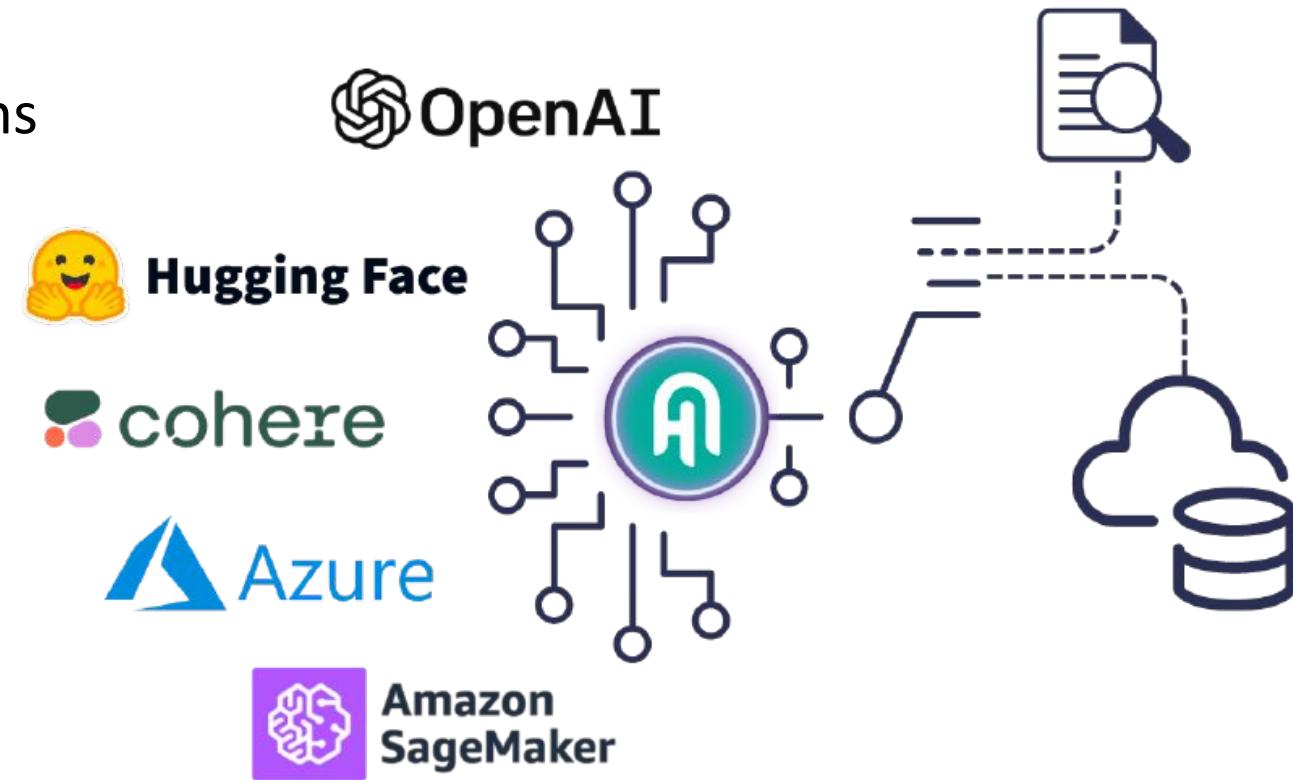
[INFO]

More info: <https://www.langchain.com>

HayStack

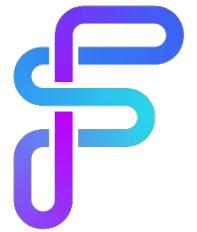


- Fully open source NLP Python framework
 - Semantic Search (Question Answering, Document Search)
 - Text generation
 - ...
- To build production-ready applications
- Integration with
 - Hugging Face
 - OpenAI
 - ...
- End-to-end
 - Preprocessing, pipelines
 - Agents & tools, prompts,
 - Evaluation and fine tuning
- Intro: [FOSDEM 2023 talk](#)



[INFO]

More info: <https://haystack.deepset.ai> and <https://github.com/deepset-ai>



AutoChain

- Lightweight and testable alternative to LangChain
- Inspired by LangChain & AutoGPT
- Aims to solve both problems
- Extensible framework to build own agents using LLMs with custom tools and automatically evaluating different user scenarios with simulated conversations.
- Easy to navigate for LangChain users
- The goal:
Enable rapid iteration on generative agents,
by simplifying agent customization and evaluation.

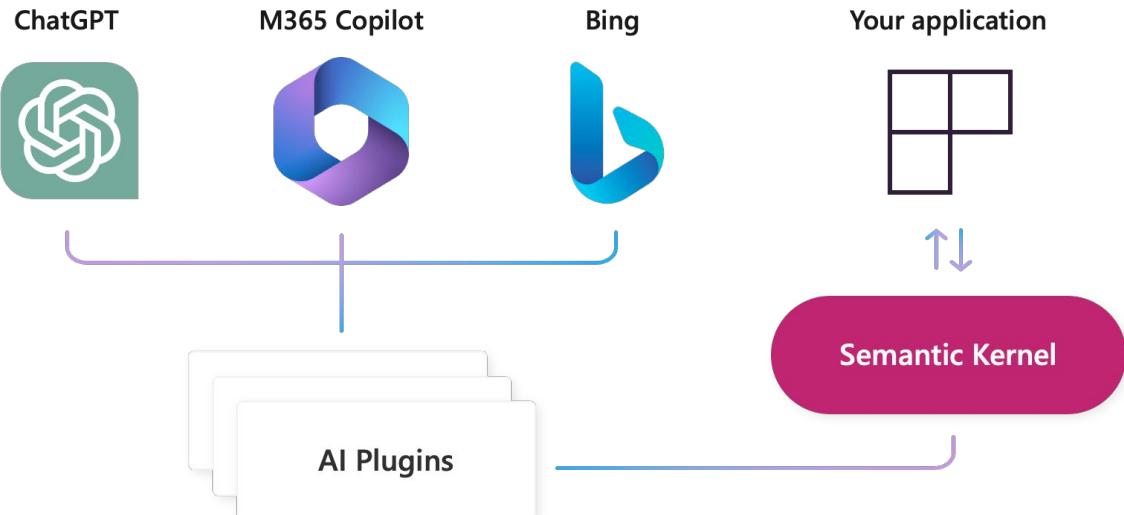
[INFO]

More info: <https://autochain.forethought.ai> and <https://github.com/Forethought-Technologies/AutoChain>

Semantic Kernel



- SDK to integrates LLMs
(OpenAI, Azure OpenAI, and Hugging Face, ...)
with conventional programming languages
(C#, Python, and Java)
- Build agents that can call existing code



Services	C#	Python	Java	Notes
TextGeneration	✓	✓	✓	Example: Text-Davinci-003
TextEmbeddings	✓	✓	✓	Example: Text-Embeddings-Ada-002
ChatCompletion	✓	✓	✓	Example: GPT4, Chat-GPT
Image Generation	✓	✗	✗	Example: Dall-E

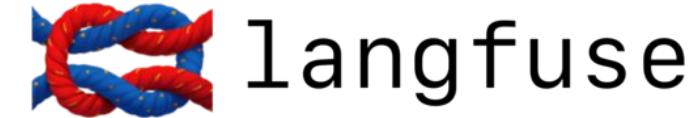
Endpoints	C#	Python	Java	Notes
OpenAI	✓	✓	✓	
AzureOpenAI	✓	✓	✓	
Hugging Face Inference API	✗	✗	✗	Coming soon to Python, not all scenarios are covered for .NET
Hugging Face Local	✗	✓	✗	
Custom	✓	✗	✗	Requires the user to define the service schema in their application

Tokenizers	C#	Python	Java	Notes
GPT2	✓	✓	✓	
GPT3	✓	✗	✗	
tiktoken	✗	✗	✗	Coming soon to Python and C#. Can be manually added to Python via <code>pip install tiktoken</code>

[INFO]

More info: <https://github.com/microsoft/semantic-kernel> and <https://learn.microsoft.com/en-us/semantic-kernel/overview> and [SupportedSKLanguages](#)

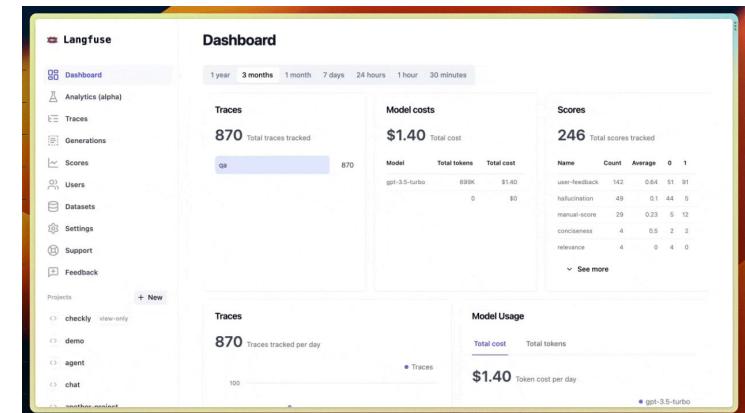
LangFuse



- Open source observability & analytics for LLM-based applications
- Observability
 - Explore and debug complex logs & traces in a visual UI
 - Admin UI to explore the ingested data.
 - Nested view of LLM app executions
 - Detailed information along the traces on: latency, cost, scores
 - Segment execution traces
- Analytics
 - Measure & improve costs, latency and response quality

Reporting on

- Token usage by model
- Volume of traces
- Scores/evals



[INFO]

More info: <https://langfuse.com> and <https://github.com/langfuse/langfuse>

Evaluation framework for RAG pipelines

- Component-Wise Evaluation
(Faithfulness, answer relevancy, context recall/precision/relevancy)
- End-to-end evaluation (Answer semantic similarity & answer correctness)

	question	ground_truths	answer	contexts	context_relevancy	faithfulness	answer_relevancy
0	How to deposit a cheque issued to an associate...	[Have the check reissued to the proper payee...]	\nThe best way to deposit a cheque issued to a...	[Just have the associate sign the back and the...	0.867	1.0	0.922
1	Can I send a money order from USPS as a business?	[Sure you can. You can fill in whatever you w...]	\nYes, you can send a money order from USPS as...	[Sure you can. You can fill in whatever you w...	0.855	1.0	0.923
2	1 EIN doing business under multiple business n...	[You're confusing a lot of things here. Compan...	\nYes, it is possible to have one EIN doing bu...	[You're confusing a lot of things here. Compan...	0.768	1.0	0.824
3	Applying for and receiving business credit	["I'm afraid the great myth of limited liabili...]	\nApplying for and receiving business credit c...	[Set up a meeting with the bank that handles y...	0.781	1.0	0.830
4	401k Transfer After Business Closure	[You should probably consult an attorney. Howe...	\nIf your employer has closed and you need to ...	[The time horizon for your 401K/IRA is essenti...	0.737	1.0	0.753

ragas score

generation

faithfulness

how factually accurate is the generated answer

answer relevancy

how relevant is the generated answer to the question

retrieval

context precision

the signal to noise ratio of retrieved context

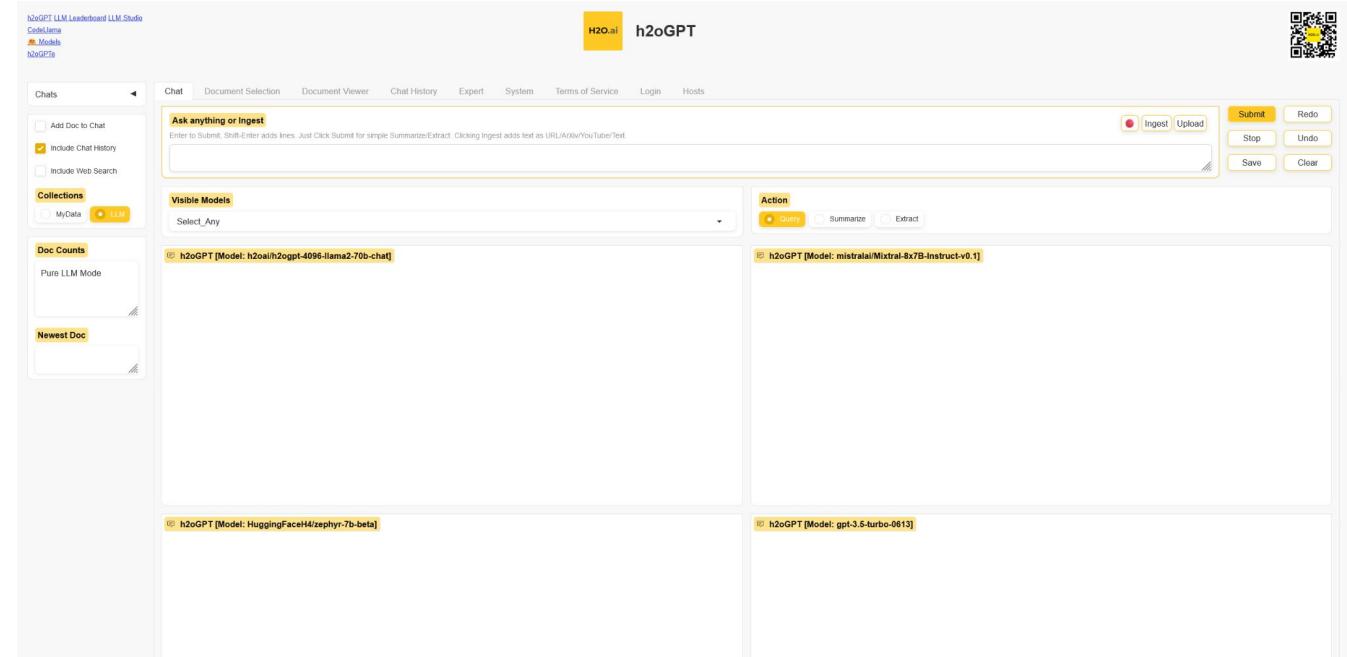
context recall

can it retrieve all the relevant information required to answer the question

h2oGPT

H2O.ai

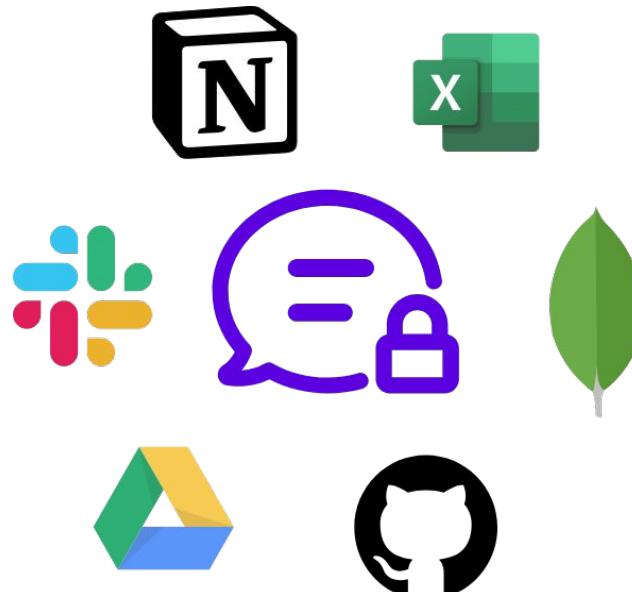
- 100% private
- Open Source (Apache 2.0) LLM tool
- Private Q&A
(Private offline database of any documents.)
- Summarization of documents & images
- Chat with local GPT
- Supports
 - Mixtral
 - llama.cpp, and
 - ...
- Demo
 - <https://gpt.h2o.ai>
 - <https://codellama.h2o.ai>



PrivateGPT

PrivateGPT

- Open source (Apache 2.0) production-ready AI project
- Interact with your documents using the power of GPT
- 100% privately
- No data leaks



The screenshot shows the PrivateGPT web application interface. At the top, there's a purple header bar with the 'PRIVATE GPT' logo. Below it, the main interface has two main sections. On the left, under 'Mode', 'Query Docs' is selected. It includes fields for 'Upload a File' (with 'bitcoin.pdf' and 'Chapter1-DigitalCurrency_DavidLee.pdf' listed), 'File name', and a 'Search in Docs' button. On the right, under 'LLM: local', a query is shown: 'According to David Lee's document, what is the major benefit of Bitcoin, and where in the document is it explained?'. The response below it states: 'The major benefit of Bitcoin, as explained in the Handbook of Digital Currency, is its ability to enable fast transactions at low costs. This is mentioned in section 1.5.1 under the subheading "Freedom of payments."'. At the bottom, there's a message input field ('Type a message...'), a 'Submit' button, and three buttons for 'Retry', 'Undo', and 'Clear'.

[INFO]

More info: <https://www.privategpt.io> and <https://github.com/imartinez/privateGPT>



N e x T

Practical example with LLMs



N e x T

Large Language Models

The experiences...

New tools spring up like mushrooms!



Hallucinations...



Integrations are not trivial



Non deterministic output of LLMs → Use as PA!



LLMs are not always the best tool...



You

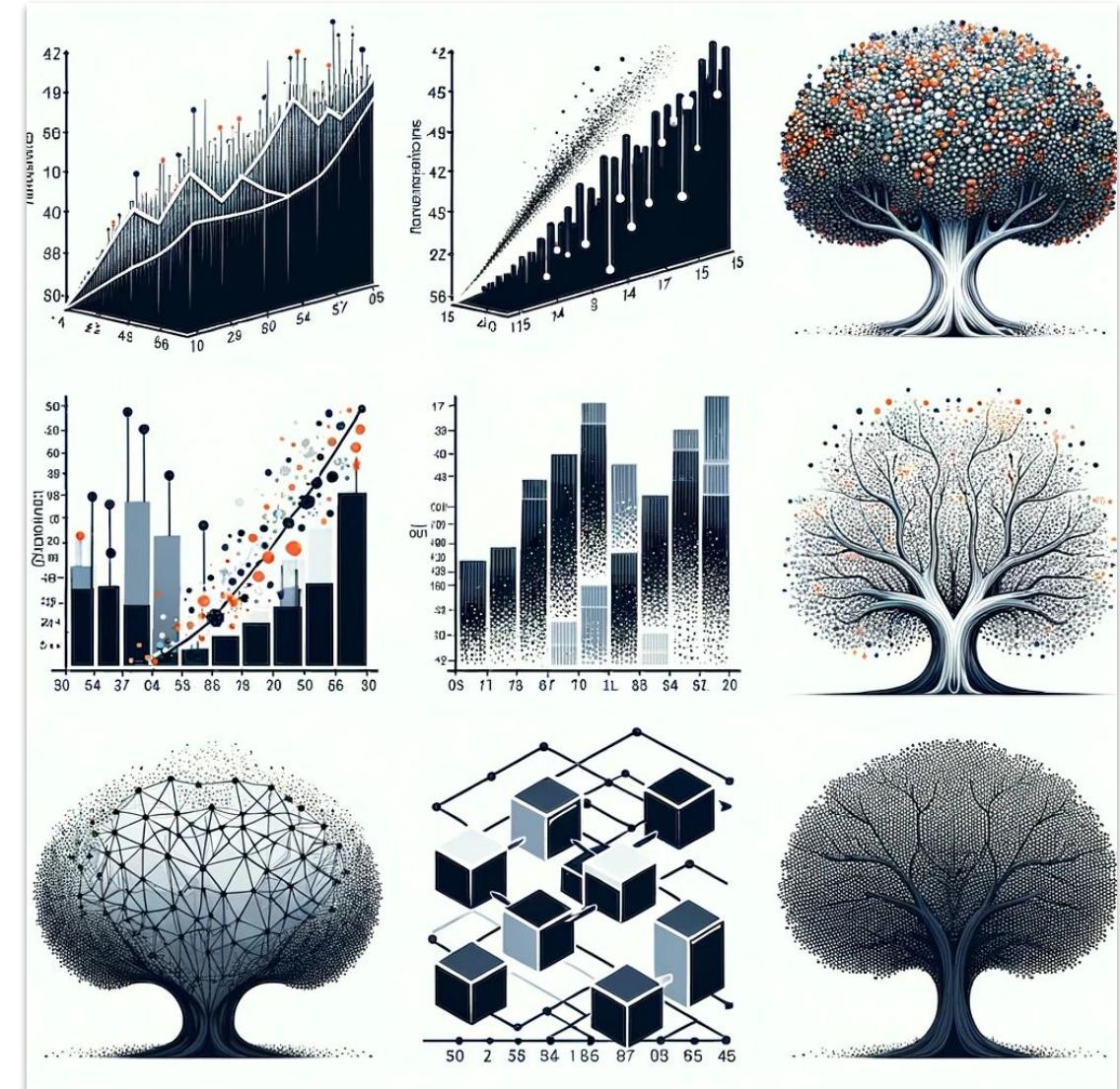
Stop giving me colorful images and text give me formal images depicting old school AI algorithms



DALL-E

Understood. I'll create a more formal and traditional set of images depicting old school AI algorithms. The focus will be on clarity and simplicity, avoiding colorful or overly artistic representations. Let's illustrate the following algorithms:

- 1. Linear Regression:** A classic Cartesian graph with a data scatter plot and a straight line depicting the regression line.
- 2. Decision Tree Algorithm:** A flowchart-like tree structure, showing a hierarchical decision-making process with simple branching.
- 3. K-Nearest Neighbors (KNN):** A 2D plot showing clusters of data points with different markers, highlighting the nearest neighbors concept.
- 4. Random Forests:** A series of decision trees side by side to represent the ensemble approach, with each tree showing a simple decision path.
- 5. Support Vector Machines (SVM):** A 2D plot with two types of data points on either side of a dividing line, illustrating the margin and support vectors.





N e x T

Hands-on exercises

<https://github.com/PXLAIRobotics/PG.AI-TA-2324-HandsOn>