ECON441B : Intro Machine Learning Lab

Week 7, Lecture 7 | Large Language Models

Sam Borghese

Wednesday, February 14th, 2024

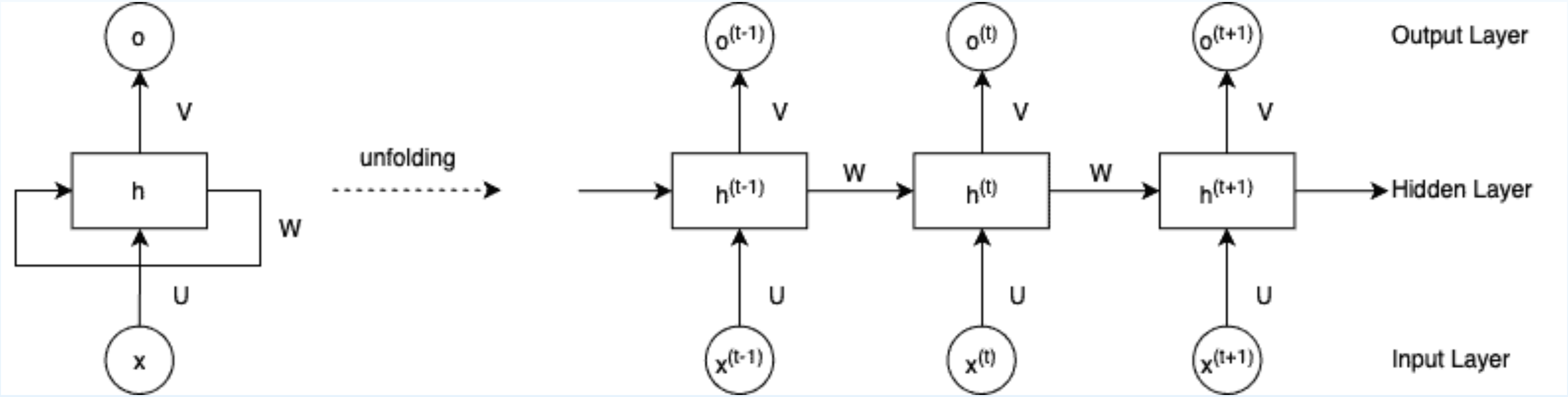
1. How do Transformers work?
2. History of LLMs
3. Custom LLMs
4. OpenAI API
5. In-Class Example

How do Transformers work?

RNN(1986)->LSTM->Transformers

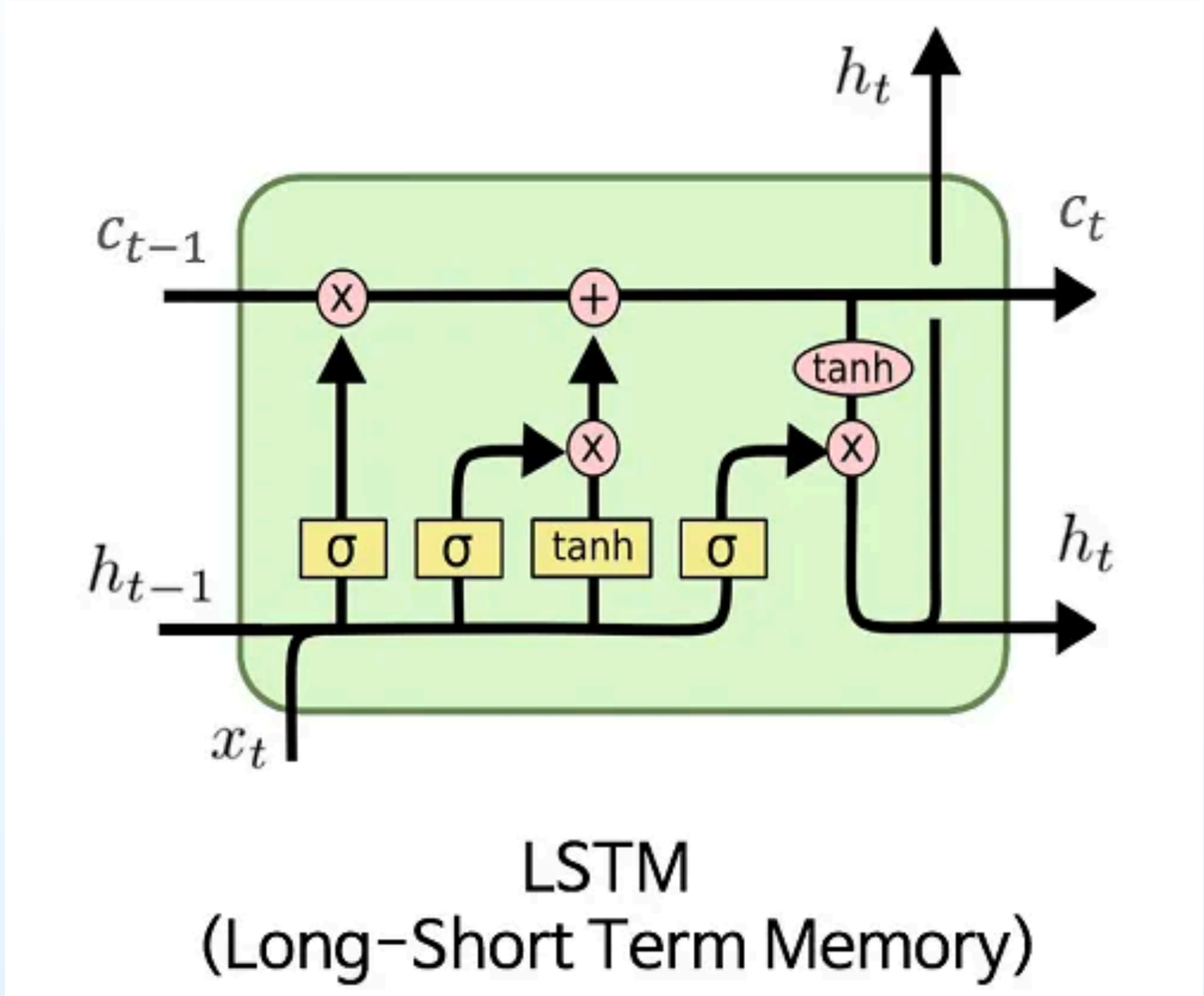
RNN architecture is used as a building block for Transformers

These cells do not necessarily have to be LSTM cells



LSTM Cells (1997)

These now allow for information in sequential data to be stored and passed on to the next cells

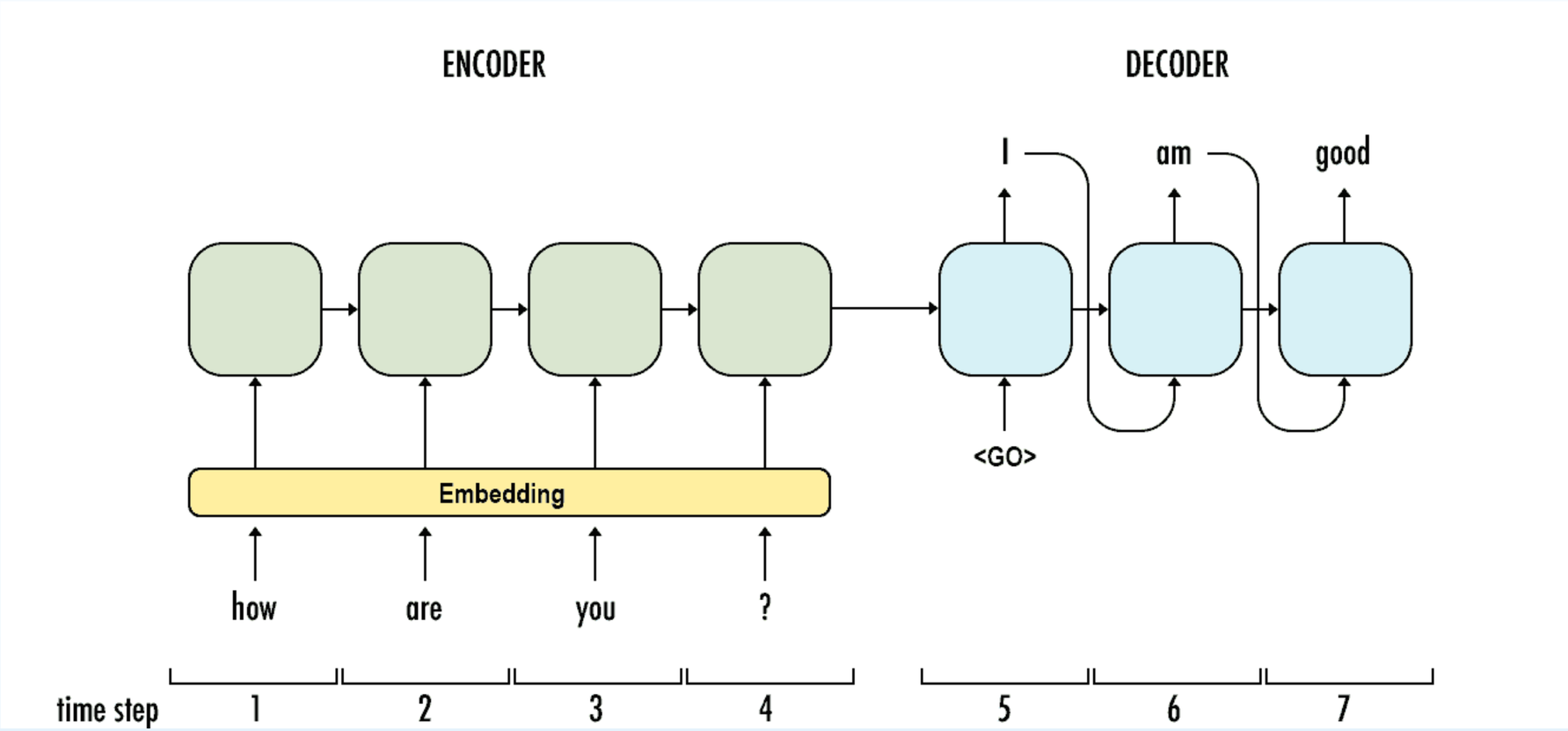


Encoder-Decoder Structure

The small difference with this structure is every output in the sequence knows what the last output was

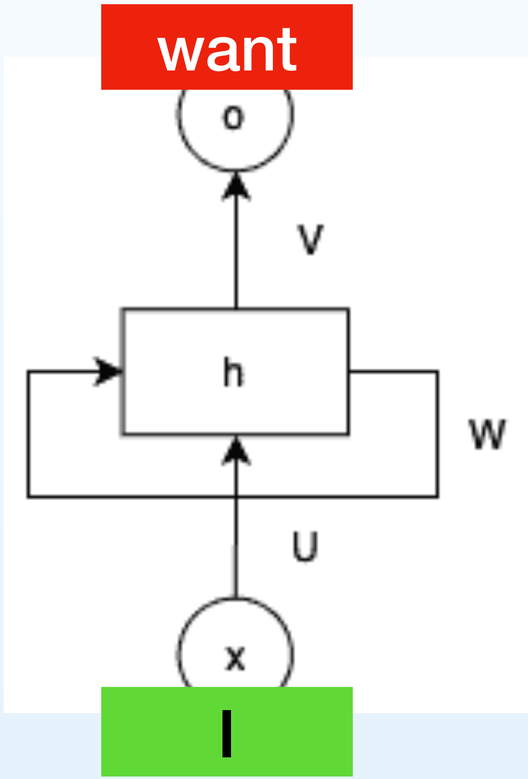
Gated recurrent units (2014)

Attention Mechanism (2014)



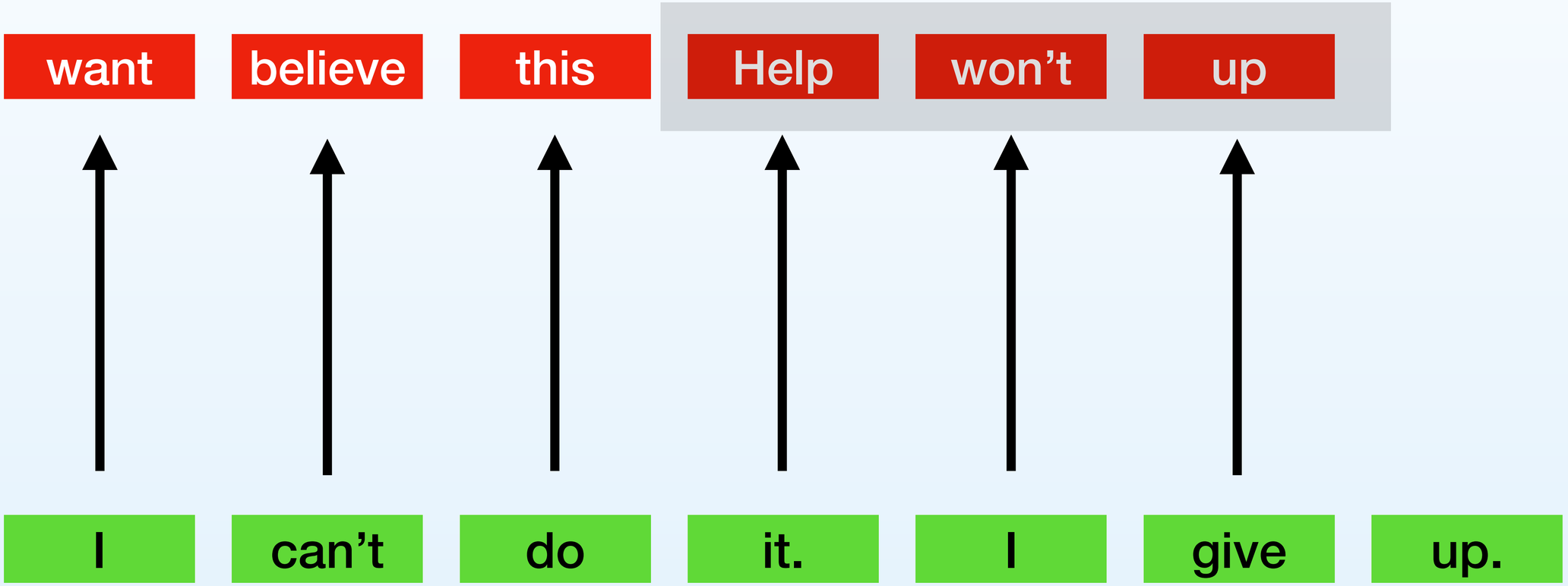
Example

Let's say that we build an RNN to try and predict the next word that will be said in the hopes that it can make full sentences.



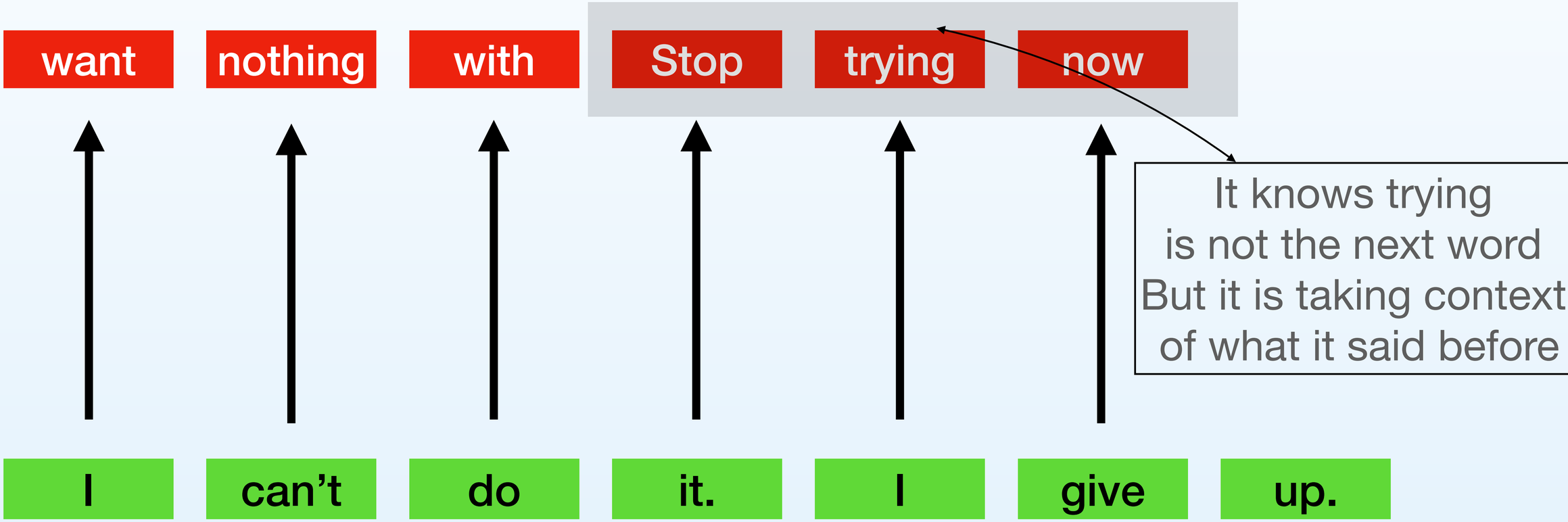
Example - RNN

Let’s say that we build an RNN to try and predict the meaning of the sentence by the next word at a time, that will be said in the hopes that it can make full sentences.



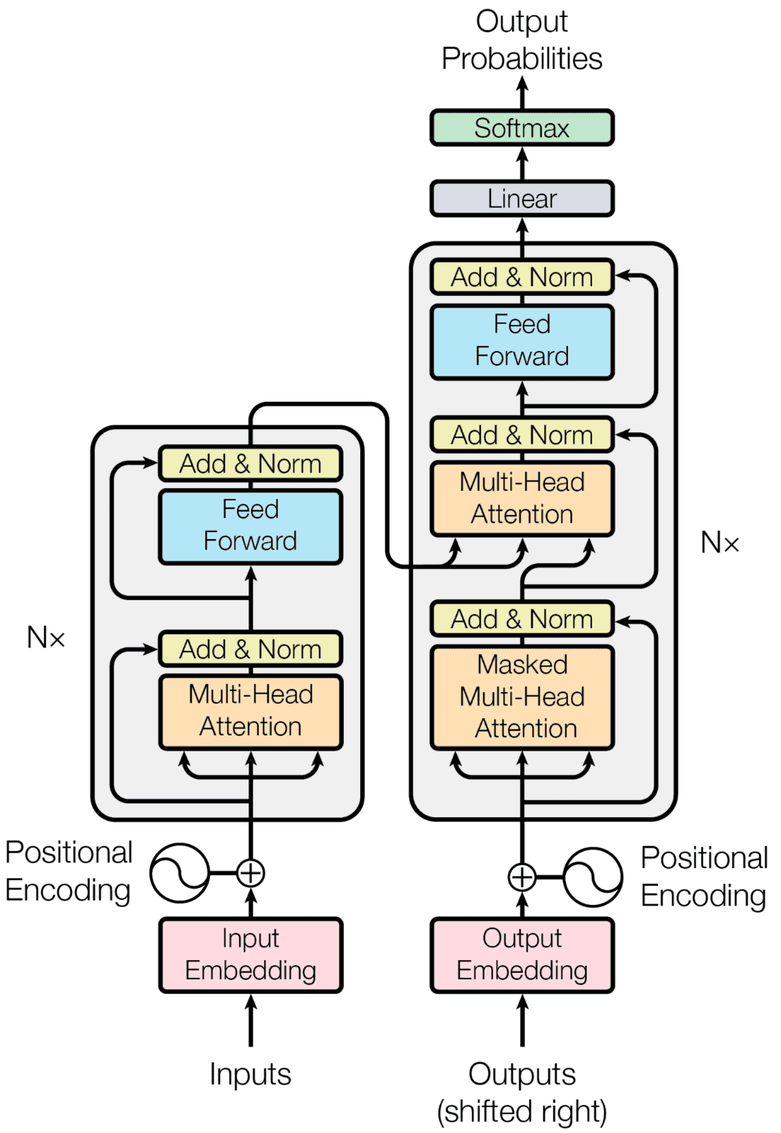
Example - Transformer

The transformer has the context of what it said before



Inside of a Transformer (2017)

This is called an attention model. It is a deep learning version of RNN



History Of LLMs

History of Large-Language Models

There is a difference between large-language models and Transformers

voicebot.ai
GIVING VOICE TO A REVOLUTION

Synthedia

LARGE LANGUAGE MODELS - PIVOTAL INNOVATIONS

Original Eliza Script

1966

ELIZA
ELIZA developed by Joseph Weizenbaum at MIT to simulate limited conversations with a human.

1972

STNLP
Statistically-Trained Natural Language Processing System (STNLP) developed by Terry Winograd at MIT. STNLP was a language model that could generate text from statistical rules.

1997

LSTM
Long Short-Term Memory (LSTM) network developed by Sepp Hochreiter and Jürgen Schmidhuber as recurrent neural network able to learn from data and generate text.

1999

NVIDIA GPU
Nvidia introduces first Graphics Processing Unit (GPU), the Geforce 256.

2000

IBM Model 1
IBM releases IBM Model 1, the first version of its statistical machine translation system.

2006

FAIR Facebook AI Research
Facebook AI Research (FAIR) is created, focused on advancing the field of AI through cutting-edge research.

2013

IBM Tangora
IBM Tangora, is able to recognize 20,000 spoken words.

2014

Google Brain
Google releases the Google Brain project, a deep learning artificial intelligence research project.

2015

Google Tensor Processing Units
Google begins using Tensor Processing Units (TPUs) internaly.

OpenAI Founded

Processing Units (TPUs) internaly.

OpenAI Founded
December: OpenAI founded to pursue the develoment of general AI.

2016

Stanford SQuAD
Stanford University's NLP Group releases the Stanford Question Answering Dataset (SQuAD), a dataset for NLP research.

2017

Introduction of Transformer Models
Transformer Models are introduced through papers like Googles Transformer: A Novel Neural Network Architecture for Language Understanding and Attention Is All You Need, Vaswani et al., 2017.

Elmo
October: Allen Institute for AI researchers develop Embeddings from Language Models (Elmo) trained on 5.5B words and 1B parameters.

Tensor Flow Framework
November: Google releases its TensorFlow framework, which allows developers to build and train their own AI models.

Universal Sentence Encoder
December: The Universal Sentence Encoder, a pre-trained model for encoding sentence pairs, is introduced by Google.

2018

Google Tensor Processing Units
Google Makes TPUs availabe to third parties.

Open AI Generative Pre-Trained Transformer
June: OpenAI publishes paper on Generative Pre-Trained Transformer (GPT) model for natural language processing with 110 million parameters, setting the foundation for future generative AI models

Google Introduces BERT
October: BERT-Large, a pre-trained transformer model for natural language processing, is introduced by Google.

2019

OpenAI GPT-2
February: OpenAI releases GPT-2, Trained on 1.6 Billion parameters GPT-2 demonstrates the ability to generate coherent and human-like text.

Baidu ERNIE
March: Baidu develops Enhanced Representati through kNowledge IntEgration (ERNIE) as a large-scale pre-trained language model, with a capacity of 4.5 billion parameters.

XLNet
June: XLNet released, beating BERT model with pre-training on a dataset of over 1 billion words with 570 million parameters.

Microsoft
Microsoft invests $1 billion into OpenAI.

Microsoft LXMERT
August: Microsoft Research announces its new language model, LXMERT, which uses pre-training on large-scale tasks and fine-tuning on target tasks.

Facebook RoBERTa
August: RoBERTa, a robustly optimized BERT model, is introduced by Facebook AI.

Amazon DeepComposer
December: Amazon Web Services introduces the DeepComposer, a language model that generates music based on text input.

2020

OpenAI GPT-3
May: OpenAI releases GPT the largest language mode to date with 175 billion parameters.

AI21 Wordtune
AI21 Wordtune generative text AI tool released.

Nvidia Megatron
October: Nvidia/Microsoft Megatron language model launches.

2021

2021

Google T5-xxl
January: Google trains T5-xxl with over 1.6 trillion parameters.

OpenAI DALL-E
January: OpenAI releases DALL-E a 12-billion parameter version of GPT-3 trained to generate images from text descriptions.

Google Transformer-X
February: Google releases its next-generation language model Transformer-X 7.5x faster than base T5.

EleutherAI GPT-Neo
March: EleutherAI releases its first open-source GPT-Neo LLMs.

Facebook XLM-R
April: The XLM-R language model, introduced by Facebook AI, is released.

Google LaMDA
May: Google introduces Language Model for Dialogue Applications (LaMDA) neural language models.

Github Copilot
June: GitHub announces Copilot, an AI pair-programmer for coding.

OpenAI Codex
July: OpenAI launches Codex, which translates natural language into code.

AI21 Labs Releases Jurassic-1
August: AI21 Labs releases Jurassic-1 large language model in two sizes as well as the AI21 Studio developer platform for creating generative AI tools and apps.

OpenAI Releases WebGPT
December: OpenAI releases WebGPT, a fine-tuned GPT-3 model for web browsers that can answer open-ended questions with citations and links to sources.

2022

OpenAI Introduces InstructGPT
January: InstructGPT OpenAI introduces InstructGPT model as new default model, one better at following instructions and less likely to hallucinate wrong answers.

Google PaLM
April: Google introduce Pathways Language Model (PaLM).

Hugging Face BLOOM
July: Machine learning developer Hugging Face launches BigScience Large Open-science Open-access Multilingual (BLOOM) Language Model trained on open-source databases.

Google DeepMind
October: Google DeepMind Sparrow dialogue agent introduced

Google Wordcraft
November: Google Wordcraft text generating tool launches.

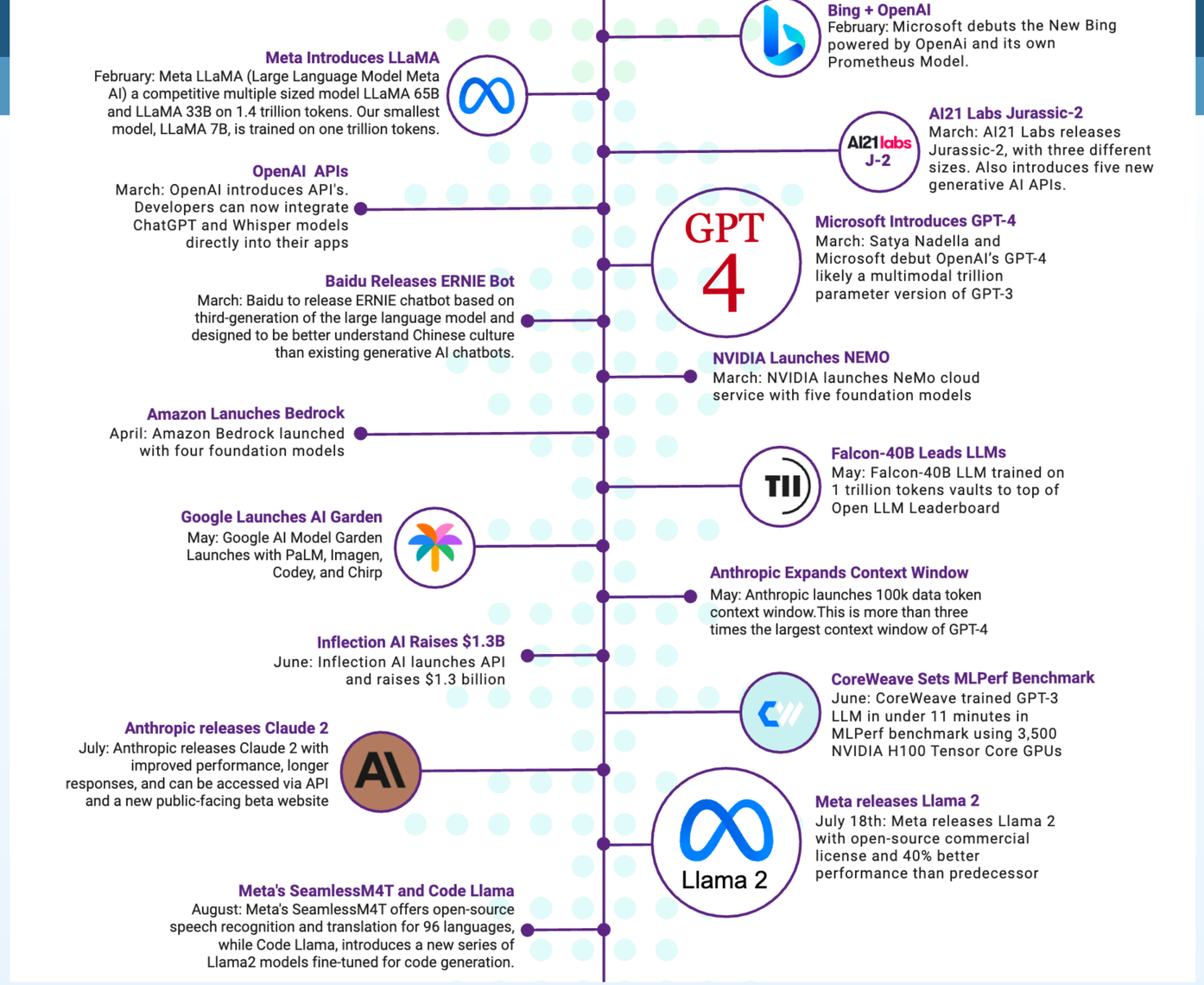
OpenAI Chat GPT
November 30: OpenAI debuts generative AI chatbot ChatGPT to explosive interest.

2023

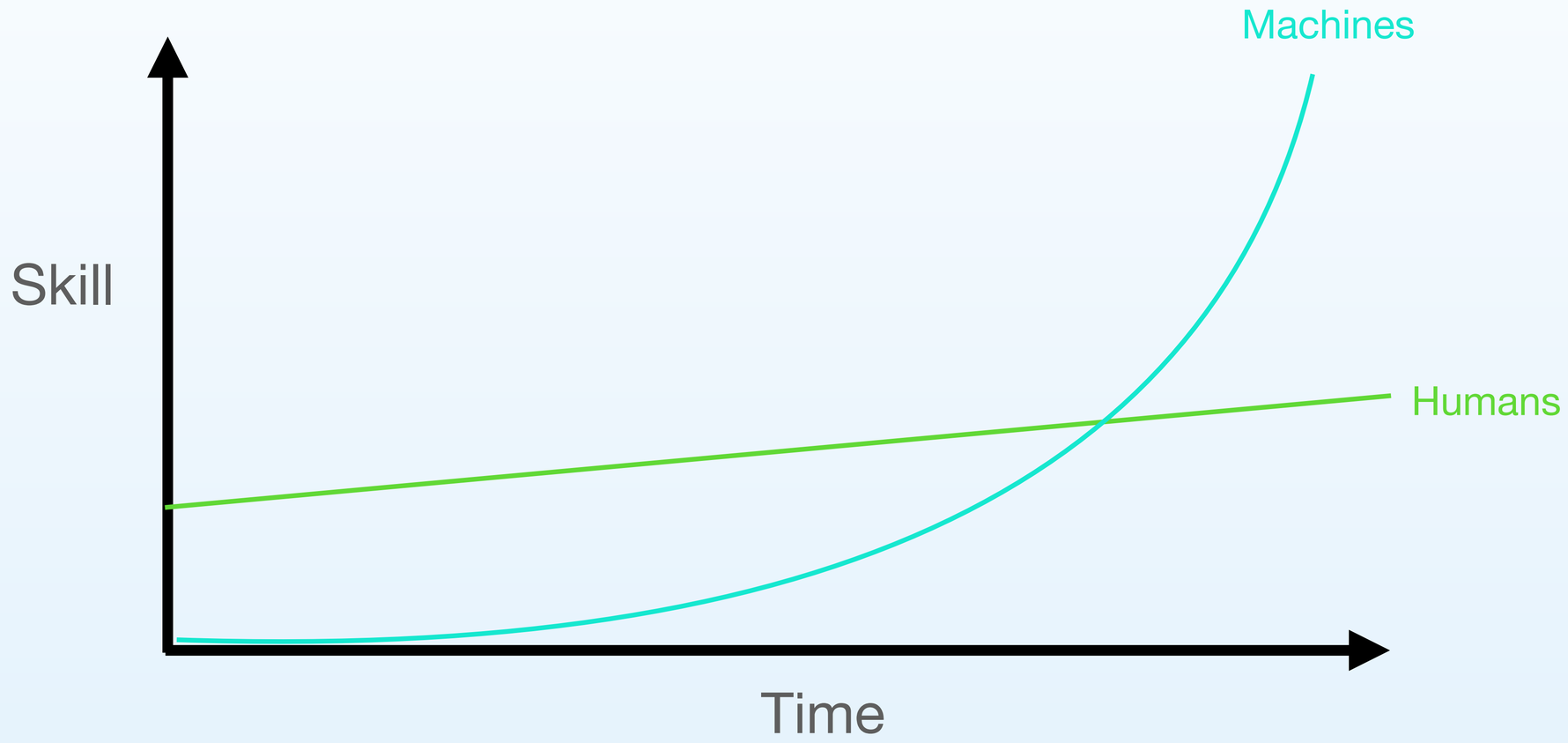
Microsoft Invests in OpenAI
January: MIcrosoft invests a rumored $10 billion more into OpenAI.

Baidu ERNIE Bot
February: Baidu previews Ernie Bot.

Google BARD
February: Google announces BARD, a ChatGPT rival.


Bing + OpenAI
February: Microsoft debuts the New Bing powered by OpenAi and its own Prometheus Model.
Meta Introduces LLaMA
February: Meta LLaMA (Large Language Model Meta AI) a competitive multiple sized model LLaMA 65B and LLaMA 33B on 1.4 trillion tokens. Our smallest model, LLaMA 7B, is trained on one trillion tokens.
AI21 Labs Jurassic-2
March: AI21 Labs releases Jurassic-2, with three different sizes. Also introduces five new generative AI APIs.
AI21labs J-2
OpenAI APIs
March: OpenAI introduces API's. Developers can now integrate ChatGPT and Whisper models directly into their apps
GPT 4
Microsoft Introduces GPT-4
March: Satya Nadella and Microsoft debut OpenAI's GPT-4 likely a multimodal trillion parameter version of GPT-3
Baidu Releases ERNIE Bot
March: Baidu to release ERNIE chatbot based on third-generation of the large language model and designed to be better understand Chinese culture than existing generative AI chatbots.
NVIDIA Launches NEMO
March: NVIDIA launches NeMo cloud service with five foundation models
Amazon Lanuches Bedrock
April: Amazon Bedrock launched with four foundation models
Falcon-40B Leads LLMs
May: Falcon-40B LLM trained on 1 trillion tokens vaults to top of Open LLM Leaderboard
TII
Google Launches AI Garden
May: Google AI Model Garden Launches with PaLM, Imagen, Codey, and Chirp
Anthropic Expands Context Window
May: Anthropic launches 100k data token context window.This is more than three times the largest context window of GPT-4
Inflection AI Raises $1.3B
June: Inflection AI launches API and raises $1.3 billion
CoreWeave Sets MLPerf Benchmark
June: CoreWeave trained GPT-3 LLM in under 11 minutes in MLPerf benchmark using 3,500 NVIDIA H100 Tensor Core GPUs
Anthropic releases Claude 2
July: Anthropic releases Claude 2 with improved performance, longer responses, and can be accessed via API and a new public-facing beta website
AI
Meta releases Llama 2
July 18th: Meta releases Llama 2 with open-source commercial license and 40% better performance than predecessor
Llama 2
Meta's SeamlessM4T and Code Llama
August: Meta's SeamlessM4T offers open-source speech recognition and translation for 96 languages, while Code Llama, introduces a new series of Llama2 models fine-tuned for code generation.

A.I. is Accelerating but Humans are not



Existential Conversation

Will A.I. replace jobs?

Case Study #1 : Automated Elevators

Elevator Automation [edit]

The elevator strikes were instrumental to the automation of the elevator. As elevators were a dangerous machine that could only be comfortably operated by elevator operators, manufacturers began adding safety features and allowing the elevator to run on its own.[19] New features included emergency phones, emergency stop buttons, and alarms.[19]

In the 1950s, Otis Elevator created many automatic elevators and created advertisements centered on their automation. Over time, most elevators did not need operators, and were running on their own.[19]

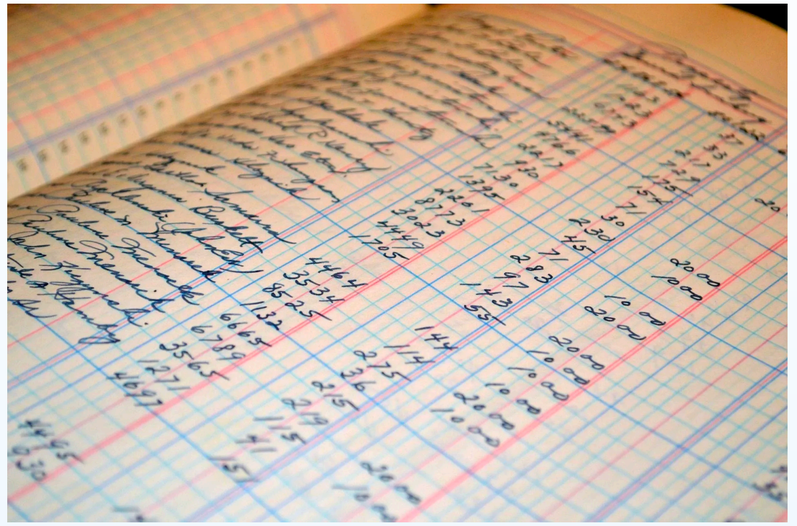
Resistance of Elevators caused automation to come quicker

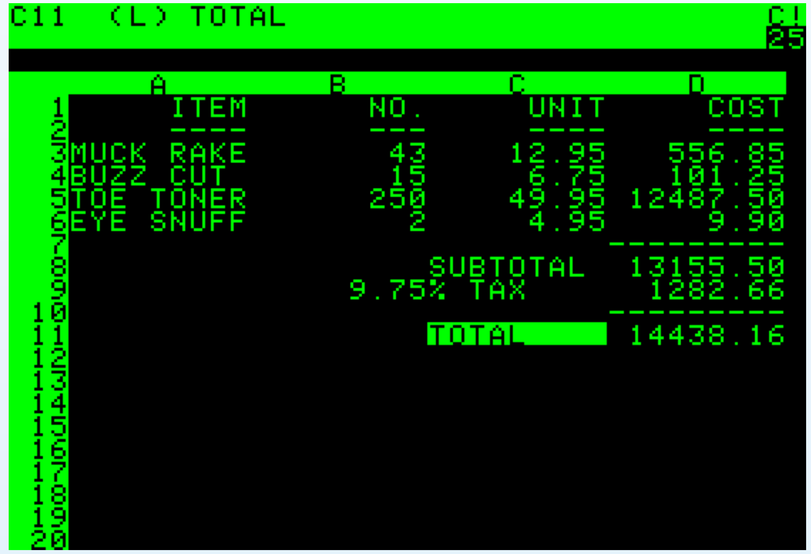
Existential Conversation

Will A.I. replace jobs?

Case Study #2 : Electronic Spreadsheets VisiCalc

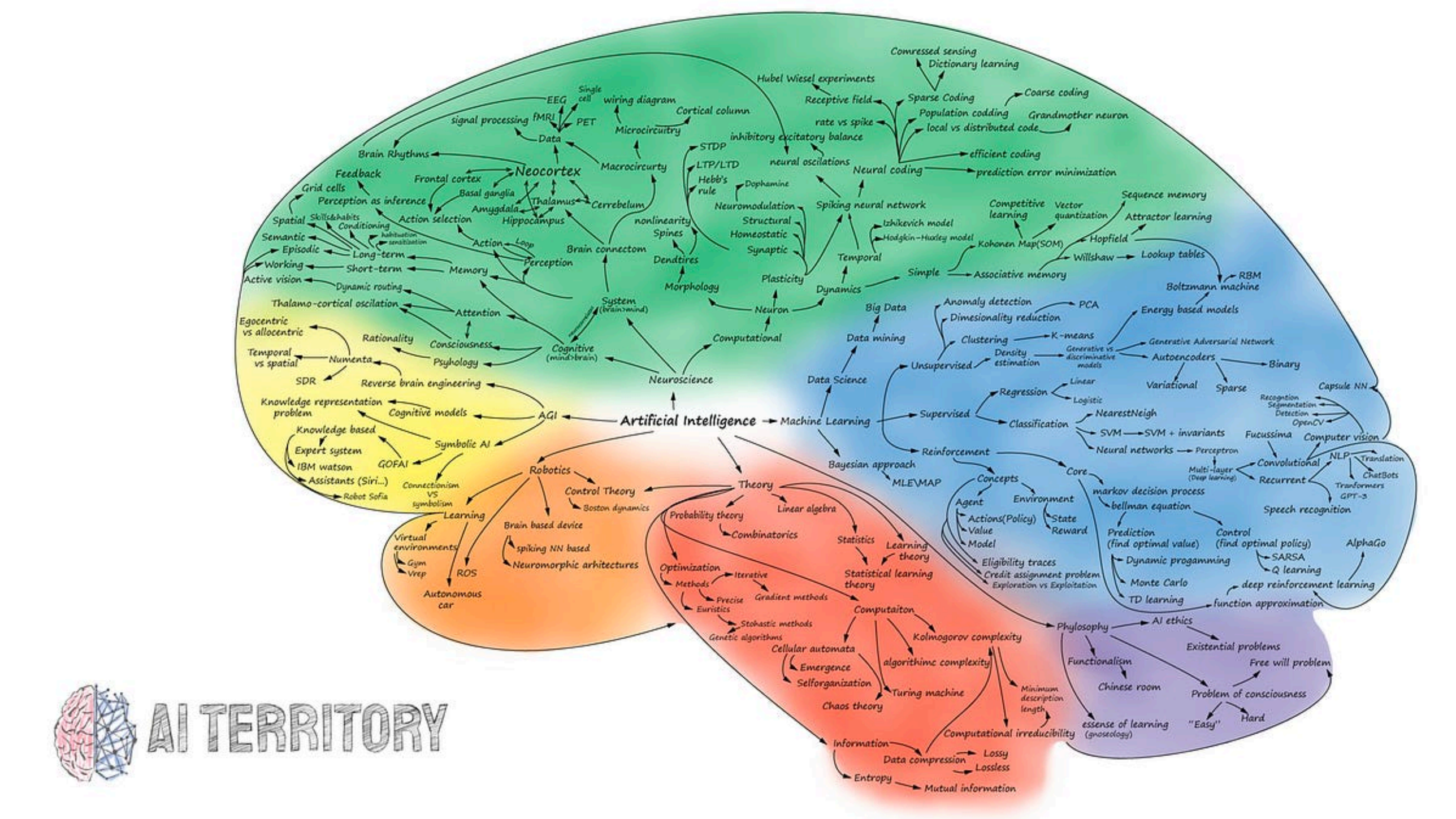
There was ~ 1 million accountants (about the same number of people who do a form of driving for a living). When the advent of electronic spreadsheet arose





Existential Conversation

“A.I. will not replace your job but someone using A.I. will”


Artificial Intelligence
Neuroscience
Machine Learning
Data Science
AGI
Cognitive models
Knowledge representation problem
Symbolic AI
Knowledge based
Expert system
IBM watson
GOFAI
Assistants (Siri...)
Robot Sofia
Connectionism VS symbolism
Reverse brain engineering
Numenta
SDR
Temporal vs spatial
Egocentric vs allocentric
Rationality
Psyhology
Consciousness
Cognitive (mind>brain)
Attention
Thalamo-cortical oscilation
Dynamic routing
Active vision
Working
Episodic
Semantic
Spatial
Skills&habits
Conditioning
habituation
sensitization
Long-term
Short-term
Memory
Action selection
Perception as inference
Grid cells
Feedback
Brain Rhythms
Frontal cortex
Basal ganglia
Amygdala
Hippocampus
Thalamus
Cerrebelum
Neocortex
Macrocircurty
Microcircuitry
Cortical column
wiring diagram
Single cell
EEG
fMRI
PET
signal processing
Data
Action
Loop
Perception
Brain connectom
System (brain>mind)
Computational
Big Data
Data mining
Anomaly detection
Dimesionality reduction
PCA
Clustering
K-means
Unsupervised
Density estimation
Generative vs discriminative models
Energy based models
Generative Adversarial Network
Autoencoders
Variational
Sparse
Binary
Boltzmann machine
RBM
Supervised
Regression
Linear
Logistic
Classification
NearestNeigh
SVM
SVM + invariants
Neural networks
Perceptron
Multi-layer (Deep learning)
Reinforcement
Bayesian approach
MLE\MAP
Concepts
Agent
Environment
Actions(Policy)
Value
Model
State
Reward
Eligibility traces
Credit assignment problem
Exploration vs Exploitation
Core
markov decision process
bellman equation
Prediction (find optimal value)
Control (find optimal policy)
Dynamic progamming
Monte Carlo
TD learning
SARSA
Q learning
deep reinforcement learning
AlphaGo
function approximation
Capsule NN
Recognition
Segmentation
Detection
OpenCV
Fucussima
Computer vision
Convolutional
Recurrent
NLP
Translation
ChatBots
Transformers
GPT-3
Speech recognition
Robotics
Control Theory
Boston dynamics
Learning
Virtual environments
Gym
Vrep
ROS
Autonomous car
Brain based device
spiking NN based
Neuromorphic arhitectures
Theory
Probability theory
Combinatorics
Linear algebra
Statistics
Learning theory
Statistical learning theory
Optimization
Methods
Iterative
Gradient methods
Precise
Euristics
Stohastic methods
Genetic algorithms
Computaiton
Cellular automata
Emergence
Selforganization
Chaos theory
Turing machine
Kolmogorov complexity
algorithimc complexity
Minimum description length
Computational irreducibility
Information
Data compression
Lossy
Lossless
Entropy
Mutual information
Phylosophy
AI ethics
Existential problems
Functionalism
Chinese room
Problem of consciousness
Free will problem
"Easy"
Hard
essense of learning (gnoseology)
Neuron
Dynamics
Plasticity
Simple
Associative memory
Temporal
Spiking neural network
Neural coding
neural oscilations
inhibitory excitatory balance
Hubel Wiesel experiments
Receptive field
rate vs spike
Sparse Coding
Comressed sensing
Dictionary learning
Population codding
Coarse coding
local vs distributed code
Grandmother neuron
efficient coding
prediction error minimization
STDP
LTP/LTD
Hebb's rule
Dophamine
Neuromodulation
Structural
Homeostatic
Synaptic
Spines
nonlinearity
Dendtires
Morphology
Izhikevich model
Hodgkin-Huxley model
Competitive learning
Vector quantization
Kohonen Map(SOM)
Sequence memory
Attractor learning
Hopfield
Willshaw
Lookup tables
AI TERRITORY

Many companies follow the hype but do not know anything about the topic

TECH / ARTIFICIAL INTELLIGENCE

Forty percent of ‘AI startups’ in Europe don’t actually use AI, claims report /
Companies want to take advantage of the AI hype

According to the survey from London venture capital firm MMC, 40 percent of European startups that are classified as AI companies don’t actually use artificial intelligence in a way that is “material” to their businesses. MMC studied some 2,830 AI startups in 13 EU countries to come to its conclusion, reviewing the “activities, focus, and funding” of each firm.

“Companies that people assume and think are AI companies are probably not.”

“In 40 percent of cases we could find no mention of evidence of AI,” MMC head of research David Kelnar, who compiled the report, told *Forbes*. Kelnar says that this means “companies that people assume and think are AI companies are probably not.”

Custom LLMs

There are many types of LLMs

1. General-purpose LLMs

These models are trained on large amounts of general web text and aim to be useful for a wide range of tasks. Examples include:

- GPT-3 — Developed by OpenAI, GPT-3 has 175 billion parameters and can generate text, answer questions and perform other tasks.
- BERT — Developed by Google, BERT uses Transformer architecture and is widely used for downstream NLP tasks.

2. Domain-specific LLMs

These models are trained on text from a specific domain like science, medicine or law. They tend to perform better for domain-specific tasks. Examples are:

- BioBERT — Trained on biomedical literature and used for biomedical NLP tasks.
- SciBERT — Trained on scientific text and effective for scientific information extraction and question answering.

3. Multilingual LLMs

These models support multiple languages and are trained on text data from different languages. Examples are:

- XLM — Developed by Facebook, XLM supports 100 languages and aims to be useful for cross-lingual tasks.
- Multilingual BERT — Supports 104 languages and can perform tasks like sentiment analysis and named entity recognition in any of those languages.

4. Few-shot LLMs

These models are designed to perform well even when fine-tuned with small amounts of labeled data. Examples are:

- GPT-3
- T5 — Developed by Google, T5 can achieve high performance with as little as 10 examples for training.

5. Task-specific LLMs

These models are tailored for a specific NLP task like summarization, question answering, translation, etc. Examples are:

- BART — Facebook's model for text generation tasks like summarization and question generation.
- ALBERT — Google's model for question answering and sentence classification.

Open Source & Closed Source LLMs

Many LLMs are open source and

Open Source LLMs

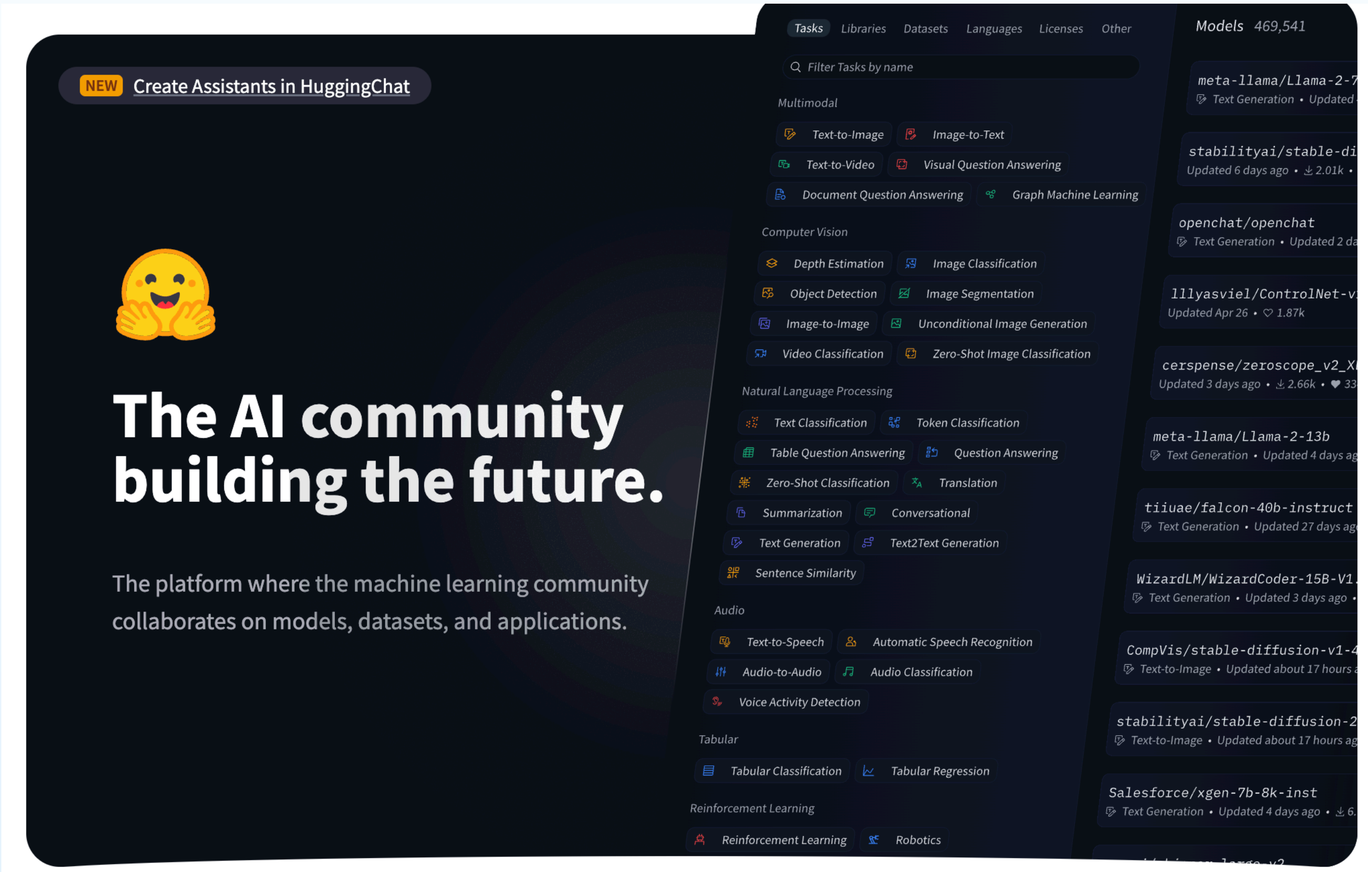
Open Source LLMs are language models whose source code is publicly available and can be freely accessed, used, modified, and distributed by anyone. Open source models encourage collaboration, transparency, and community involvement. Developers, researchers, and enthusiasts can contribute to the development, improvement, and customization of these models. The open-source nature allows for greater innovation, knowledge sharing, and collective development efforts.

Closed Source LLMs

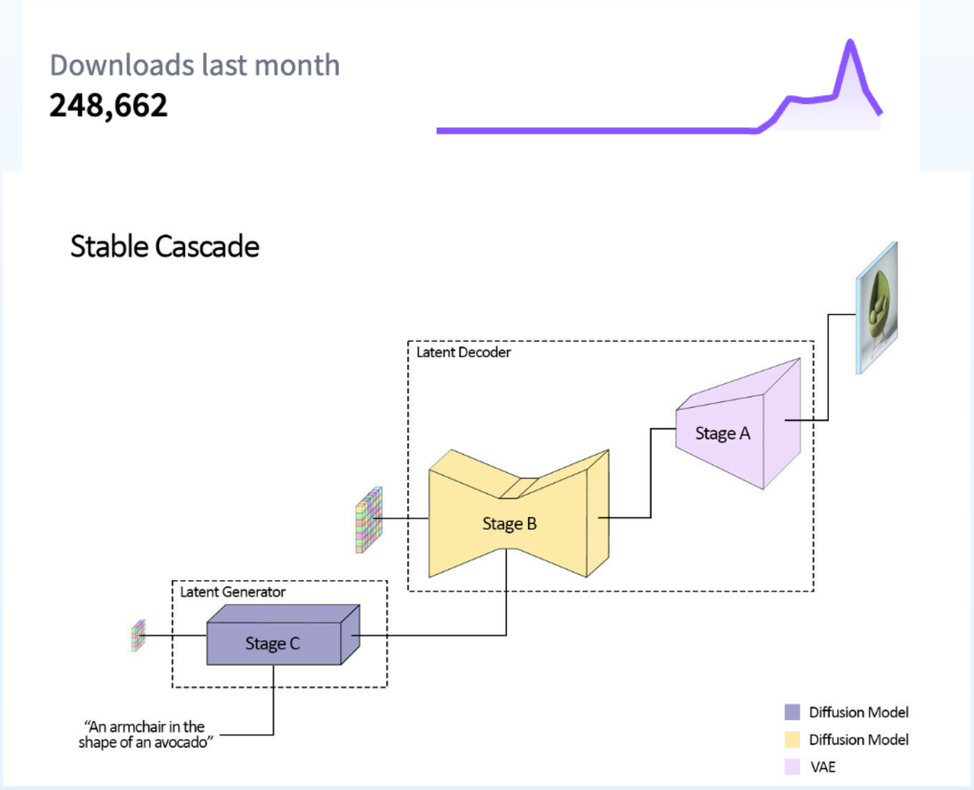
Closed Source LLMs, on the other hand, are language models whose source code is not publicly available. They are developed and maintained by organizations or companies that typically keep the underlying code proprietary and closed to the public. These models are often developed as commercial products and may require licenses or subscriptions for their use. The specific details of their architecture, training data, and algorithms are generally not accessible to the public.

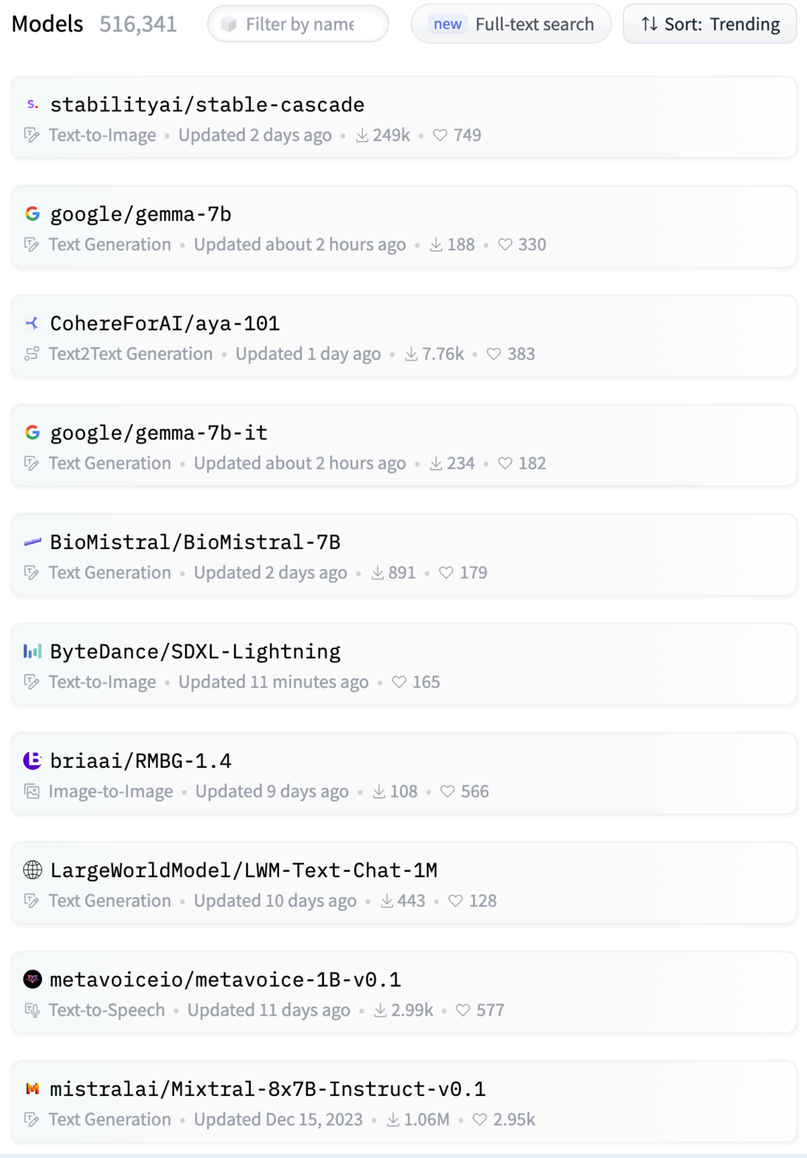
1000's of opens source LLMs are available for download and training

Many LLMs are open source and downloadable locally or accessible over the internet



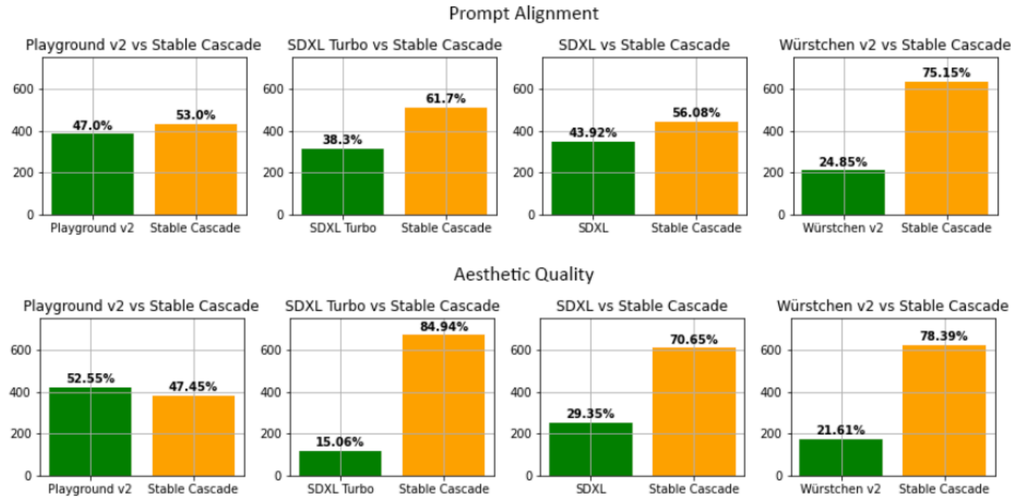
These are downloadable to your local machine and you can run and deploy them





Each of these were built for a given purpose

Evaluation



Code Example

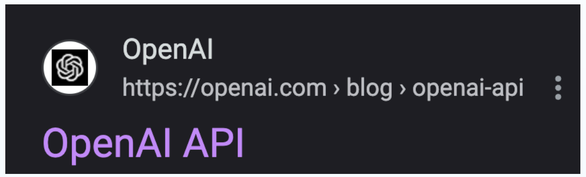
⚠ **Important**: For the code below to work, you have to install `diffusers` from this branch while the PR is WIP.

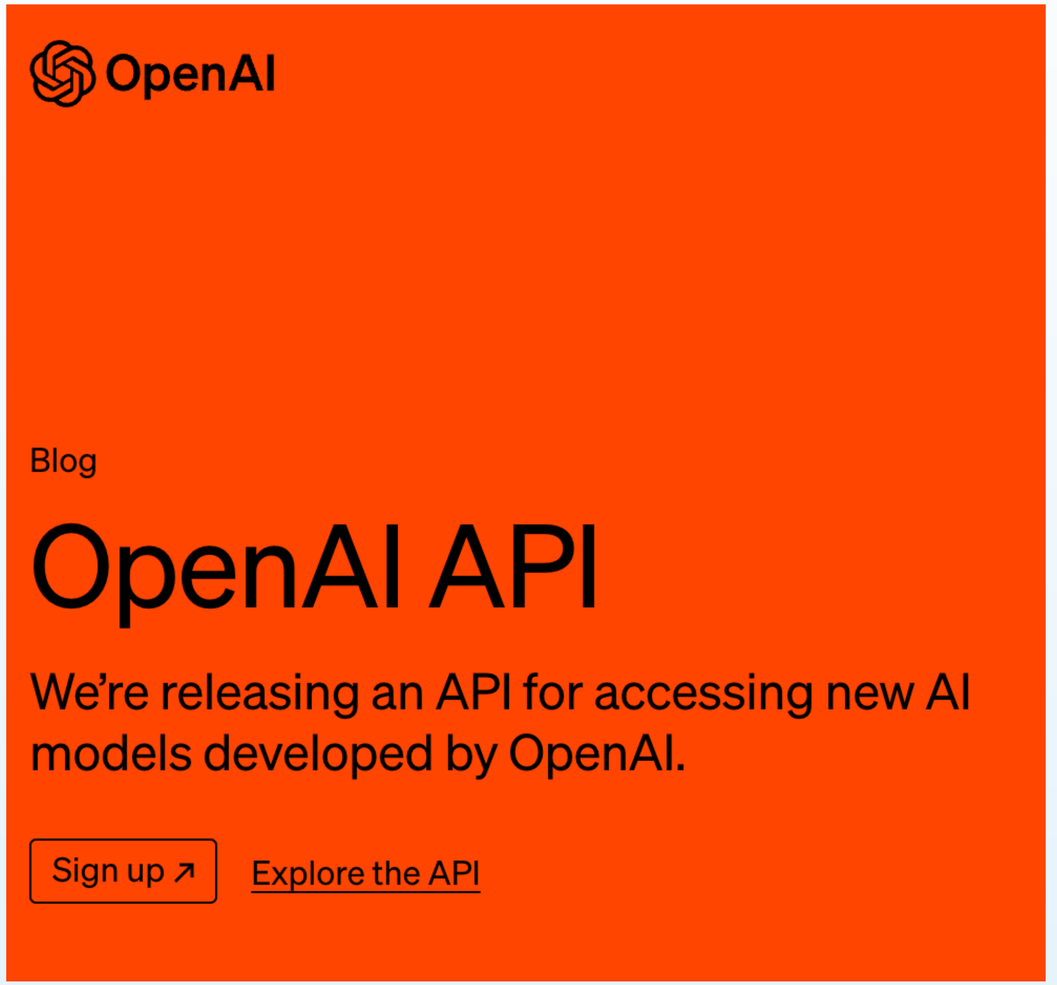
```
pip install git+https://github.com/kashif/diffusers.g
```

```
import torch
from diffusers import StableCascadeDecoderPipeline, S

device = "cuda"
num_images_per_prompt = 2

prior = StableCascadePriorPipeline.from_pretrained("s
decoder = StableCascadeDecoderPipeline.from_pretraine

prompt = "Anthropomorphic cat dressed as a pilot"
negative_prompt = ""
```

OpenAI API

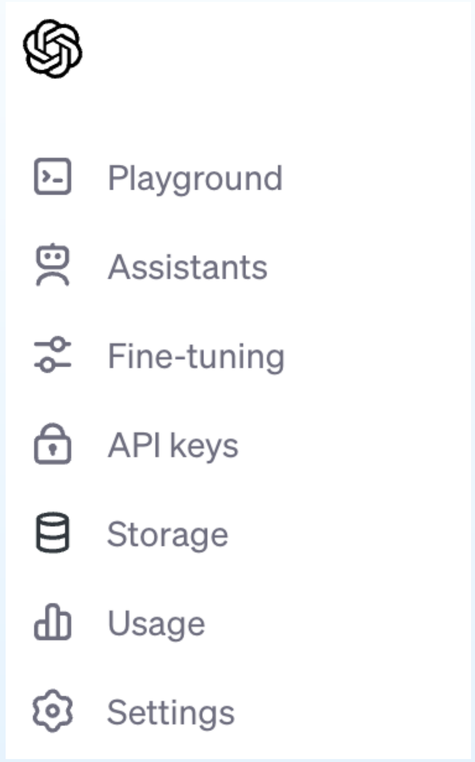
OpenAI API

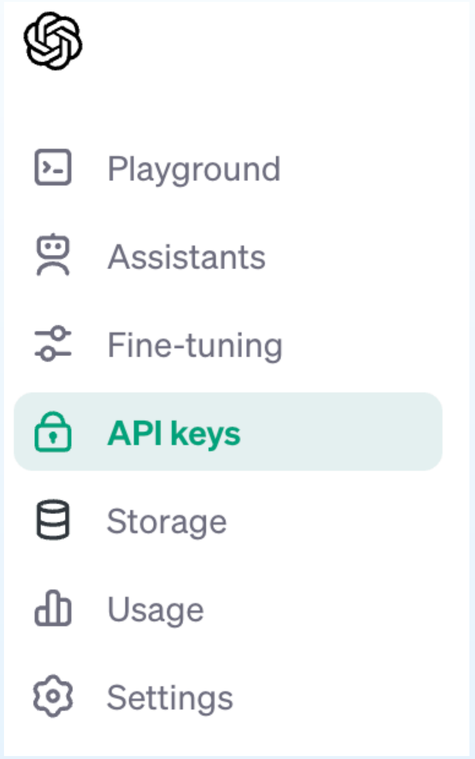

OpenAI
https://openai.com › blog › openai-api
OpenAI API

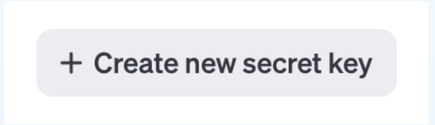

OpenAI
Blog
OpenAI API
We're releasing an API for accessing new AI models developed by OpenAI.
Sign up ↗
Explore the API

OpenAI API

Get API Keys







OpenAI API

Import and install packages

```
pip install openai
```

```
import openai
import os
```

```
openai.api_key = 'sk-m6xgK4
```

OpenAI API

Creat a client and get a response

```
client = openai.OpenAI(
    api_key=openai.api_key,
)
```

```
chat_completion = client.chat.completions.create(
    model="gpt-4",  # Use the appropriate model for your task
    messages=[
        {"role": "system", "content": "I will be passing you a t
        {"role": "user", "content": text}
    ],
)
```

```
chat_completion.choices[0].message.content
```

OpenAI API Docs

https://
platform.openai.com/
docs/introduction

Create image

`POST https://api.openai.com/v1/images/generations`

Creates an image given a prompt.

Request body

`prompt` string Required
A text description of the desired image(s). The maximum length is 1000 characters for `dall-e-2` and 4000 characters for `dall-e-3`.

`model` string Optional Defaults to dall-e-2
The model to use for image generation.

`n` integer or null Optional Defaults to 1
The number of images to generate. Must be between 1 and 10. For `dall-e-3`, only `n=1` is supported.

`quality` string Optional Defaults to standard
The quality of the image that will be generated. `hd` creates images with finer details and greater consistency across the image. This param is only supported for `dall-e-3`.

`response_format` string or null Optional Defaults to url
The format in which the generated images are returned. Must be one of `url` or `b64_json`. URLs are only valid for 60 minutes after the image has been generated.

In-Class Example

Anyone can submit information to wikipedia. Build a wikipedia fact-checker

1.) Create an OpenAI API account
2.) Use the wikipedia api to get a function that pulls in the text of a wikipedia page
3.) Build a chatgpt bot that will analyze the text given and try to locate any false info
4.) Make User input functionality for this bit to return a report of any potentially false info via wikipedia