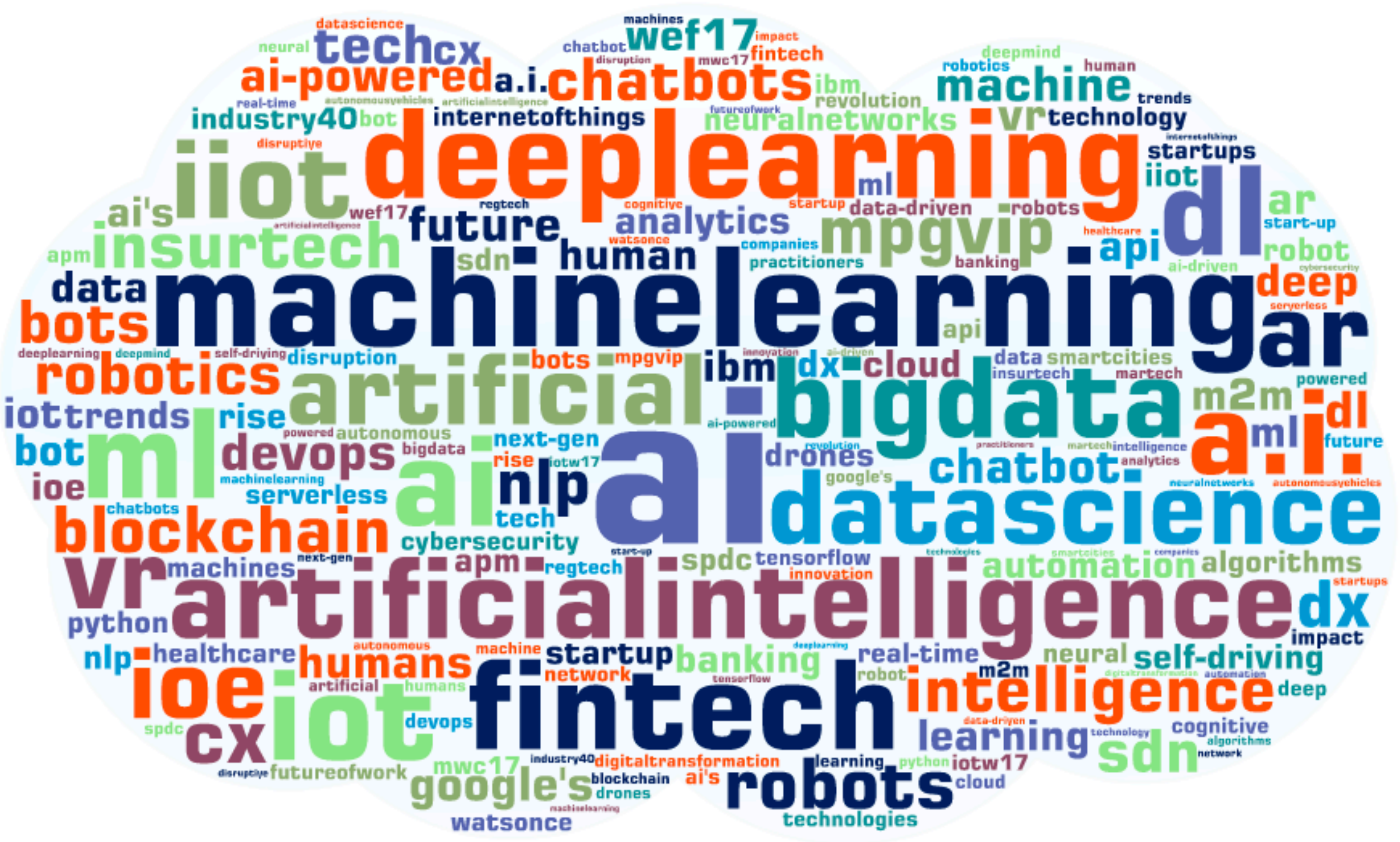
Gen AI - Azure

What is Generative AI?

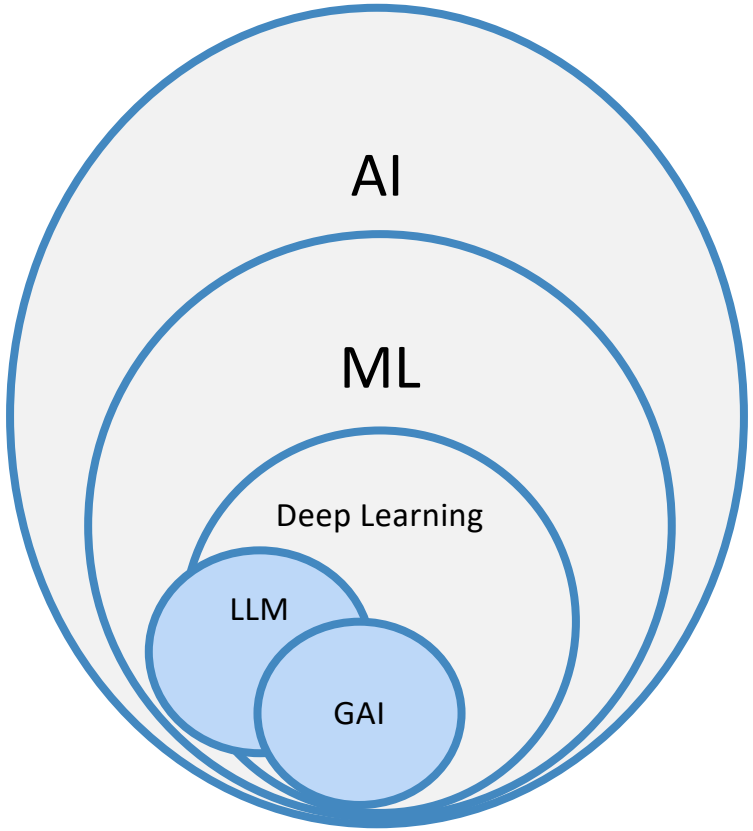
- Gen AI is a type of Artificial Intelligence that creates new content based on what it has learned from existing content.
- Generative AI was introduced in the 1960s in chatbots.
- In 2014 on introduction of GANs a type of machine learning algorithm that generative AI could create convincingly authentic images, videos and audio of real people.
- In 2017 Googles Brain team came up with Transformers as another ML algorithm with Deep Learning capabilities using LLM’s.

Buzz Words


techcx
wef17
chatbots
ai-powered
a.i.
machine
industry40
internetofthings
neuralnetworks
technology
deeplearning
iiot
dl
future
analytics
mpgvip
insurtech
human
api
robot
data
machinelearning
deep
bots
ar
robotics
artificial
ibm
dx
cloud
bigdata
iottrends
rise
m2m
a.i.
ml
devops
ai
nlp
drones
chatbot
datascience
blockchain
serverless
cybersecurity
tensorflow
automation
algorithms
vr
artificialintelligence
python
healthcare
humans
startup
banking
real-time
self-driving
ioe
iot
fintech
intelligence
cx
learning
sdn
google's
robots
watsonce
technologies

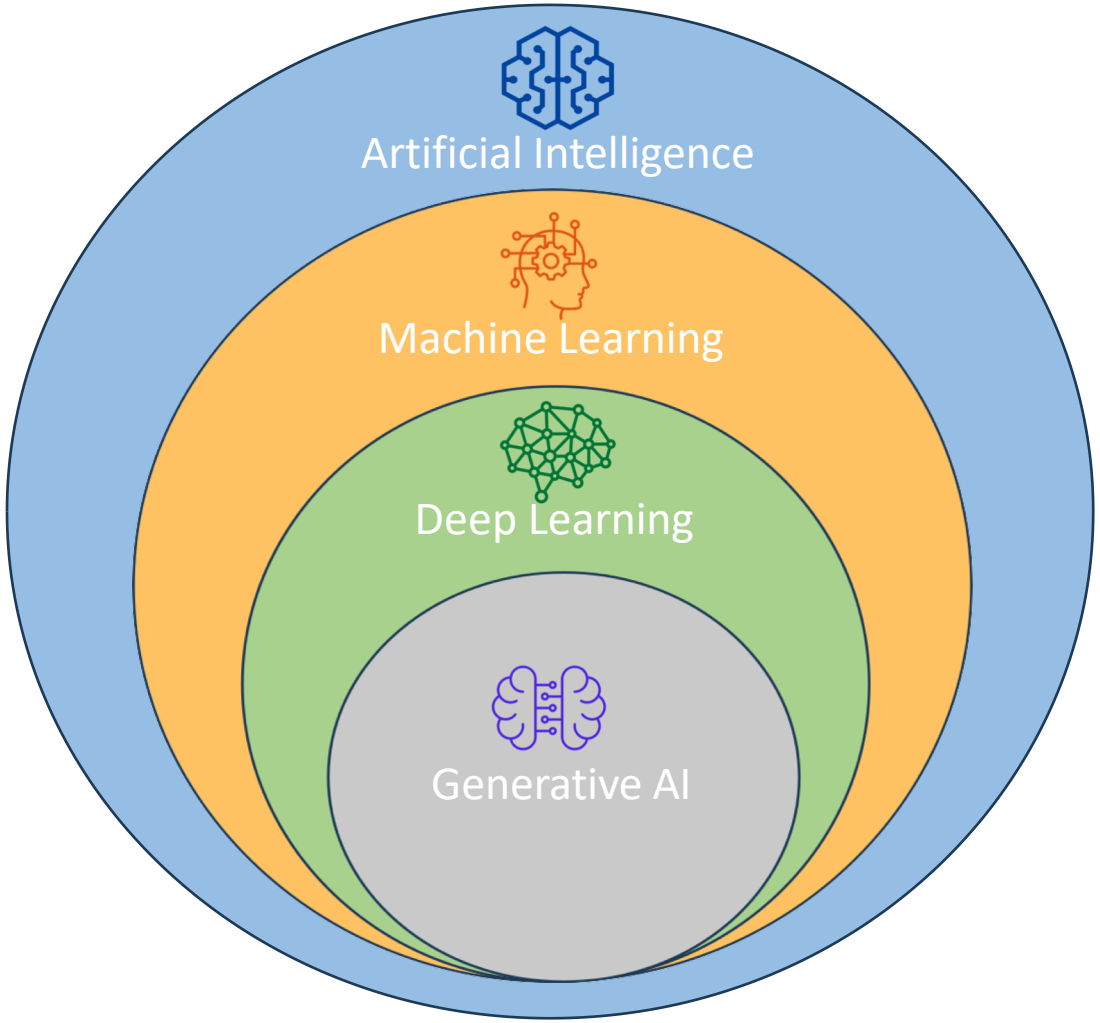
What is Generative AI

Generative AI is a type of artificial intelligence technology that can produce various types of content, including text, imagery, audio and synthetic data.



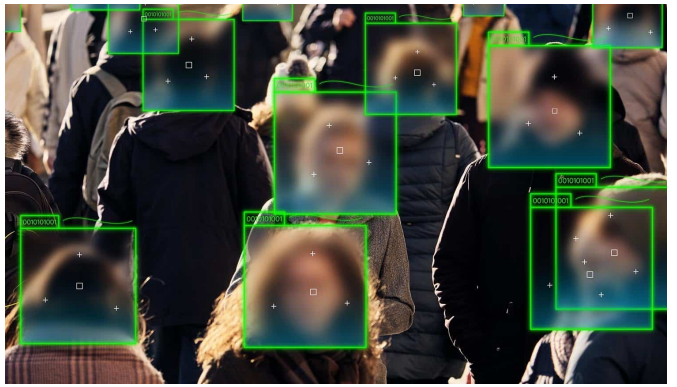
What is Artificial Intelligence (AI)?

- AI is a broad field for the development of intelligent systems capable of performing tasks that typically require human intelligence:
  - Perception
  - Reasoning
  - Learning
  - Problem solving
  - Decision-making
  - Umbrella-term for various techniques

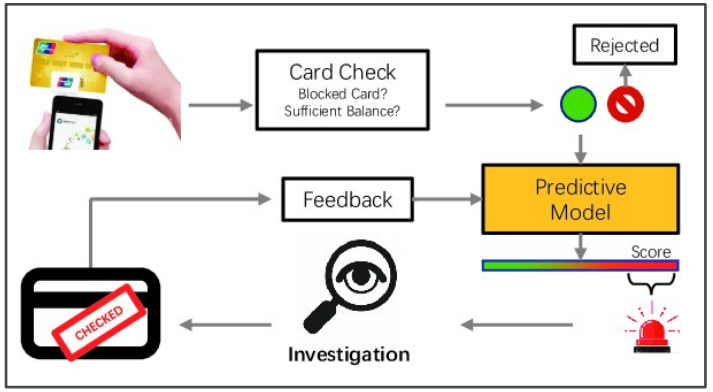


Artificial Intelligence – Use Cases

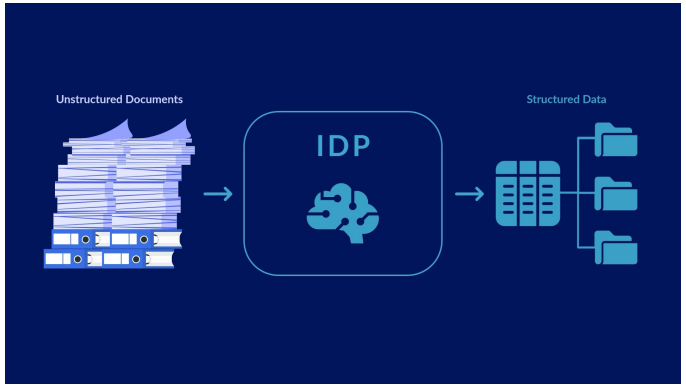
Computer Vision



Facial Recognition



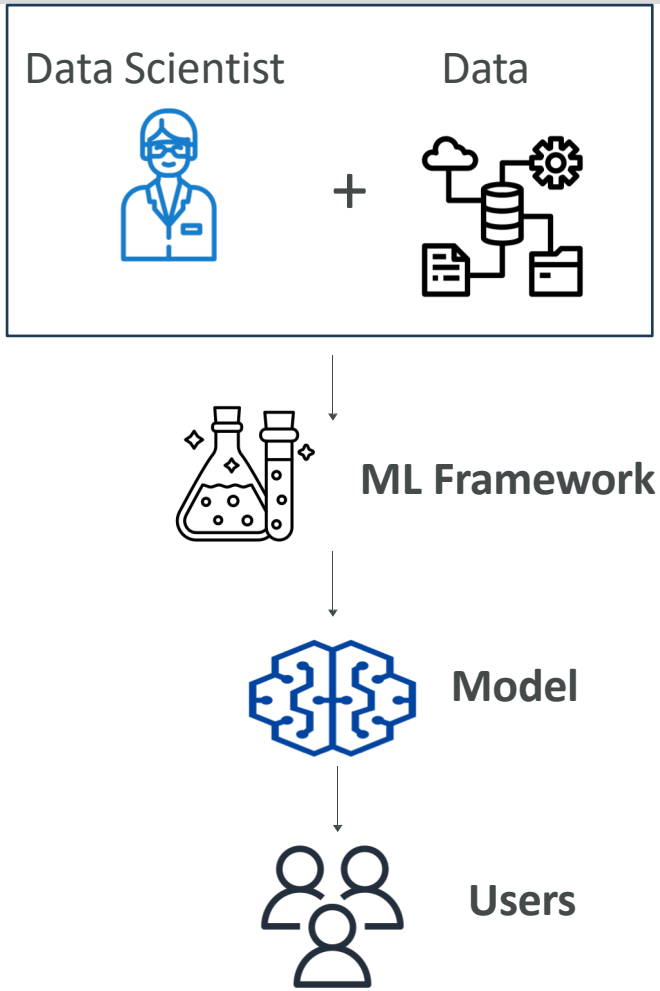
Fraud Detection



Intelligent Document Processing (IDP)

AI Components

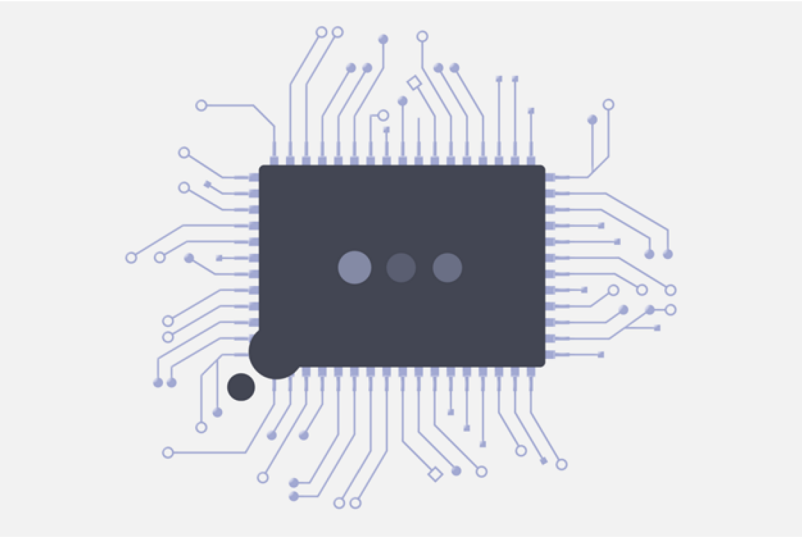
- Data Layer – collect vast amount of data
- ML Framework and Algorithm Layer – data scientists and engineer work together to understand use cases, requirements, and frameworks that can solve them
- Model Layer – implement a model and train it, we have the structure, the parameters and functions, optimizer function
- Application Layer – how to serve the model, and its capabilities for your users

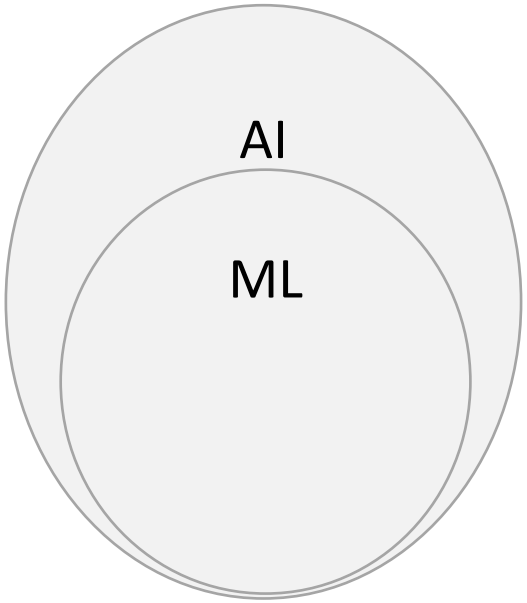


What is Machine Learning?

Machine learning is a branch of **artificial intelligence (AI)** and computer science which focuses on the use of **data** and **algorithms** to imitate the way that humans learn, gradually improving its accuracy.

How AI and ML Related?





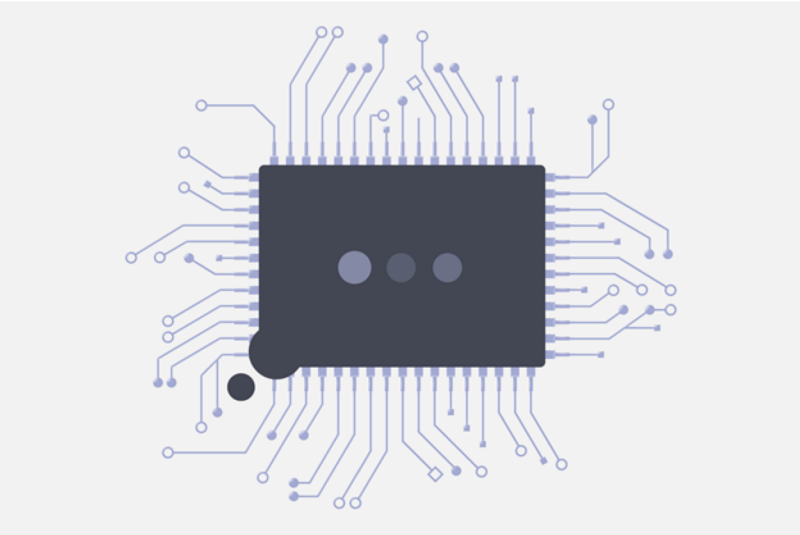
Artificial Intelligence

is a discipline

Machine Learning

is a subfield

AI != ML



Artificial Intelligence

≠

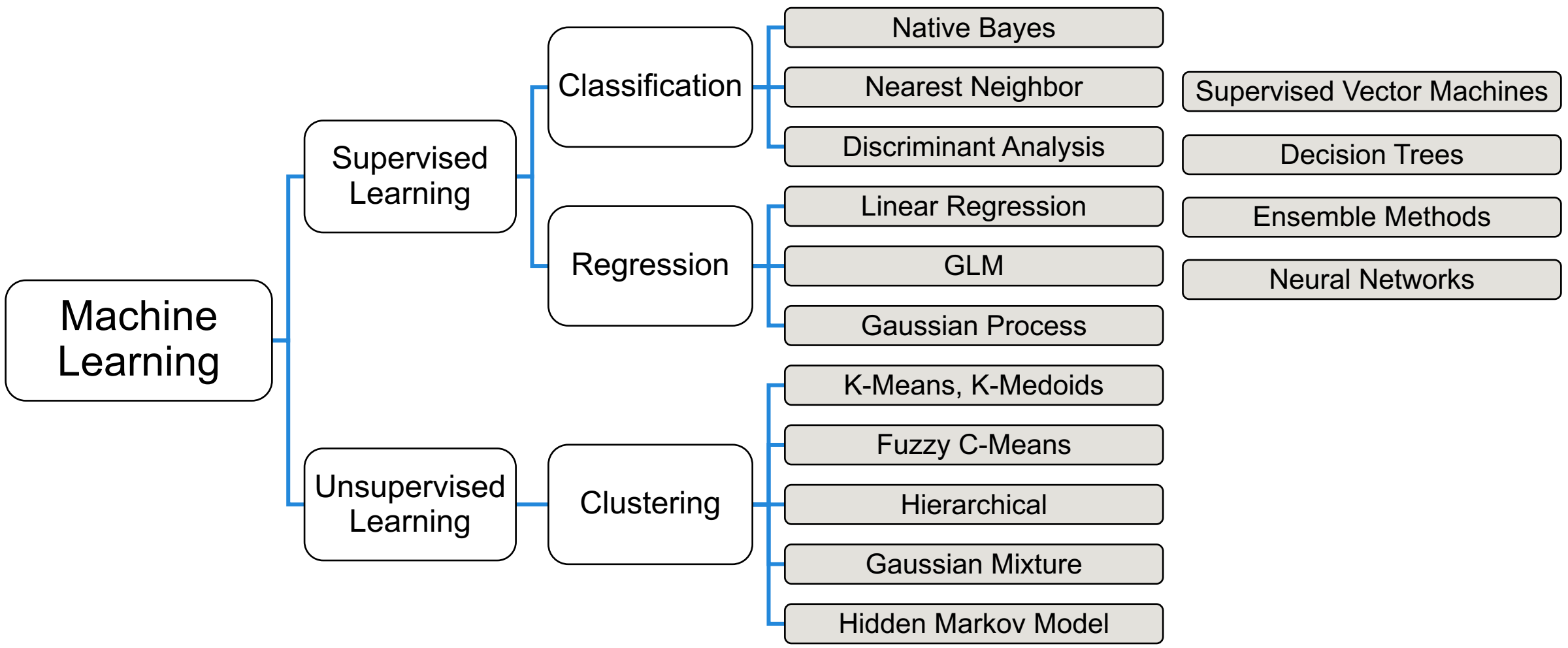
Machine Learning

ML Models

It is a program or system that trains it self from input data and create a trained model.

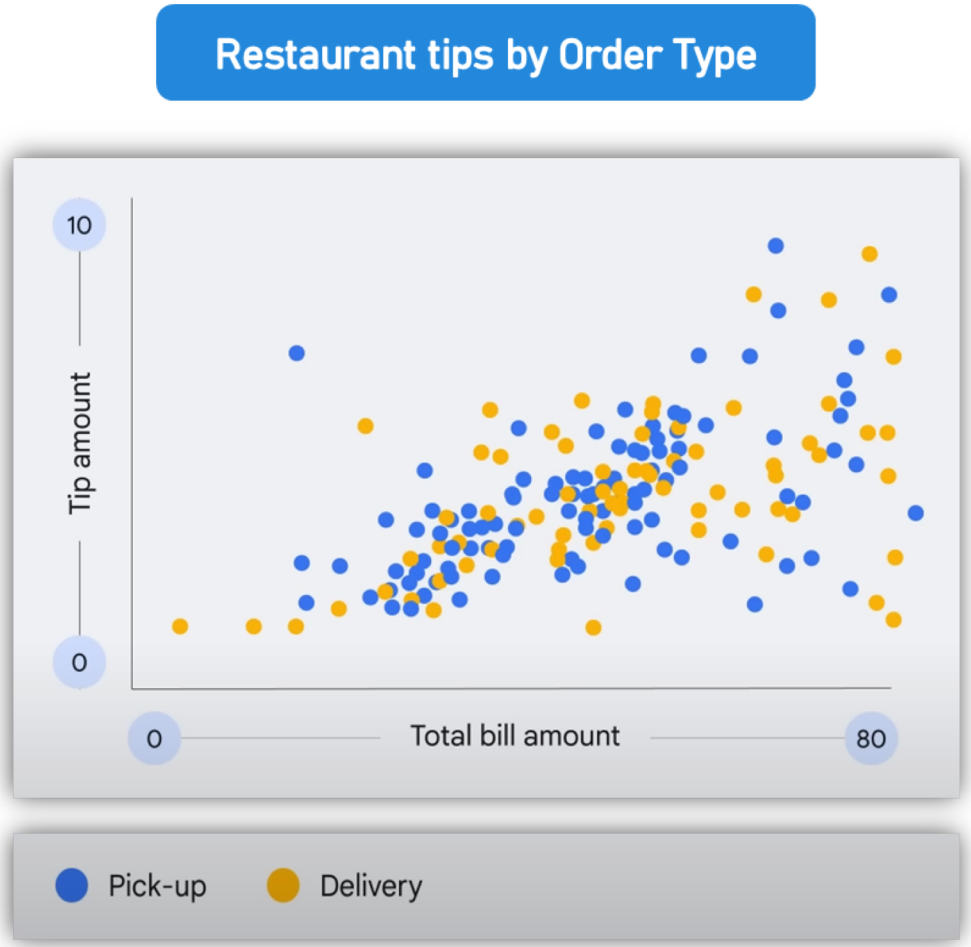
- Unsupervised ML models
- supervised ML models

ML Models


Machine Learning
Supervised Learning
Unsupervised Learning
Classification
Regression
Clustering
Native Bayes
Nearest Neighbor
Discriminant Analysis
Linear Regression
GLM
Gaussian Process
K-Means, K-Medoids
Fuzzy C-Means
Hierarchical
Gaussian Mixture
Hidden Markov Model
Supervised Vector Machines
Decision Trees
Ensemble Methods
Neural Networks

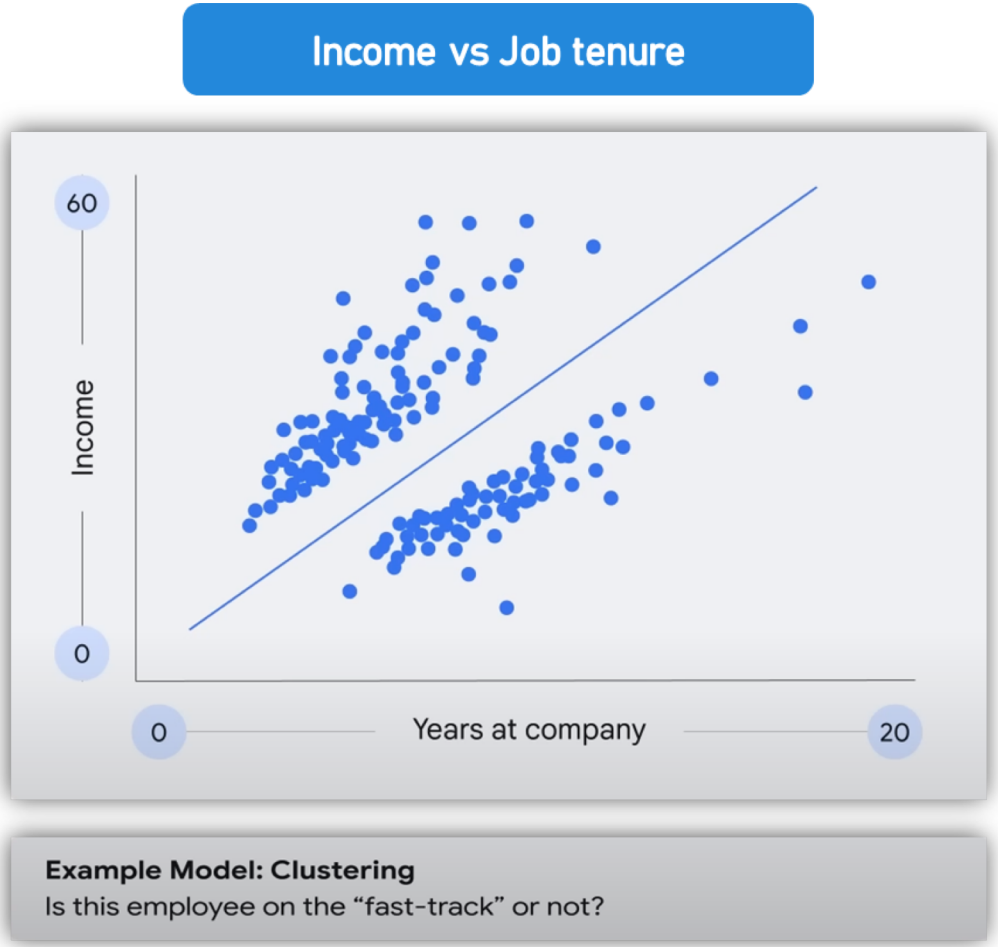
Supervised ML

- Supervised learning implies the data is already labeled
- In supervised learning we are learning from past examples to predict future values.



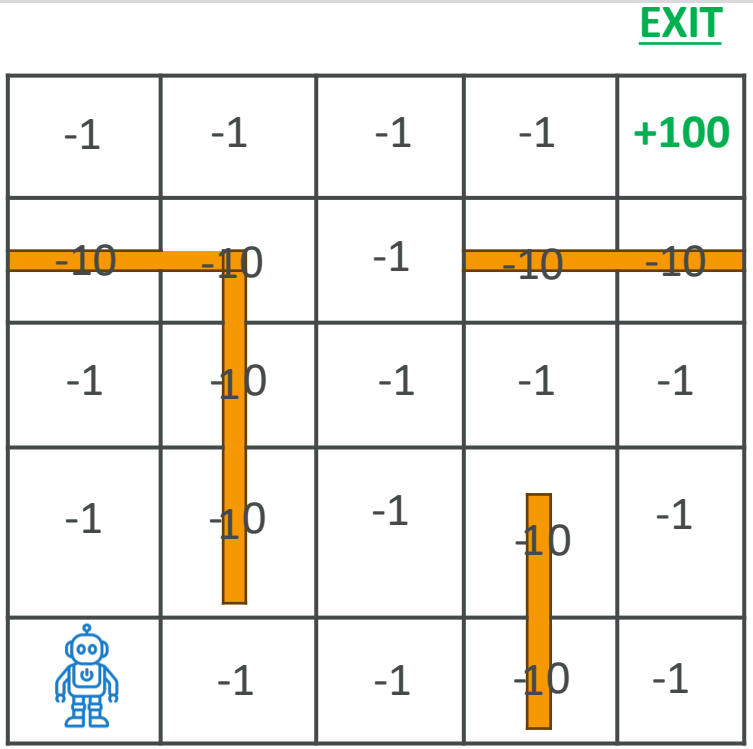
Unsupervised ML

- Unsupervised learning implies the data is not labeled
- Unsupervised problems are all about looking at the raw data, and seeing if it naturally falls into groups



What is Reinforcement Learning (RL)?

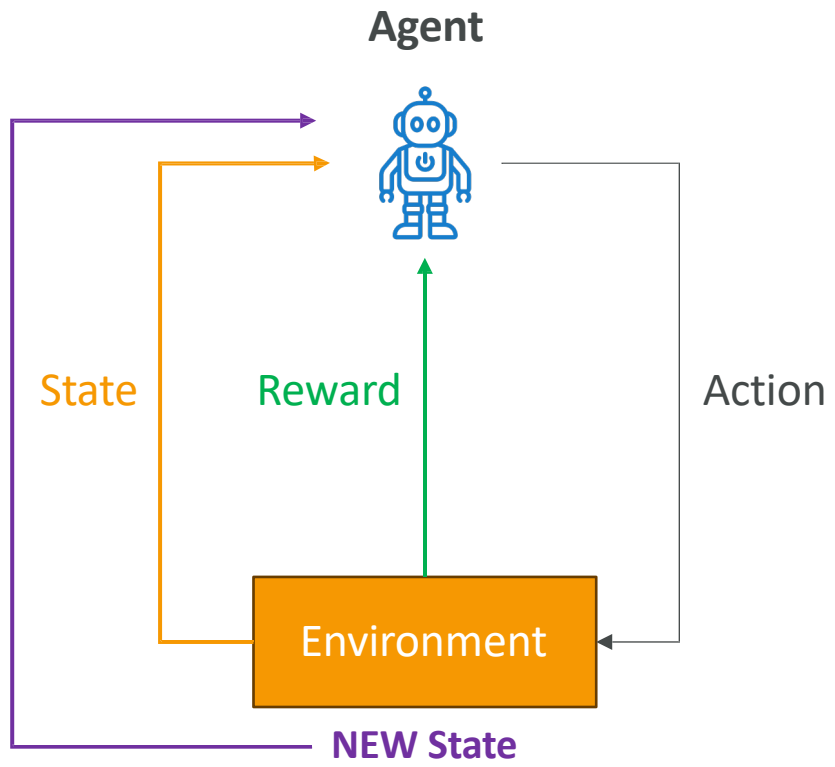
- A type of Machine Learning where an agent learns to make decisions by performing actions in an environment to maximize cumulative rewards
- Key Concepts
  - Agent – the learner or decision-maker
  - Environment – the external system the agent interacts with
  - Action – the choices made by the agent
  - Reward – the feedback from the environment based on the agent's actions
  - State – the current situation of the environment
  - Policy – the strategy the agent uses to determine actions based on the state



Simulate many times
Learn from mistakes
Learn from successes

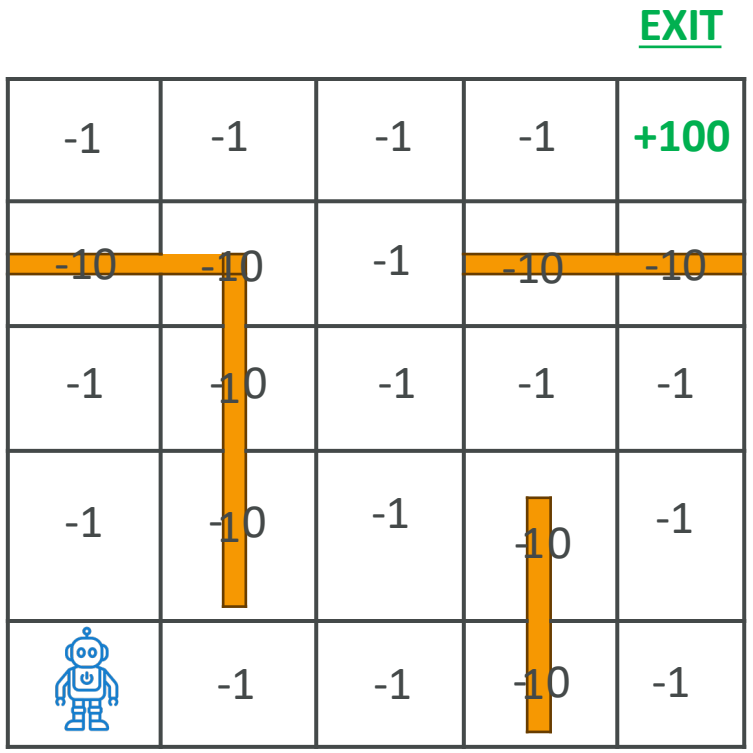
How Does Reinforcement Learning Work?

- Learning Process
  - The Agent observes the current State of the Environment
  - It selects an Action based on its Policy
  - The environment transitions to a new State and provides a Reward
  - The Agent updates its Policy to improve future decisions
- Goal: Maximize cumulative reward over time



Example: Reinforcement Learning in Action

- Scenario: training a robot to navigate a maze
- Steps: robot (Agent) observes its position (State)
  - Chooses a direction to move (Action)
  - Receives a reward (-1 for taking a step, -10 for hitting a wall, +100 for going to the exit)
  - Updates its Policy based on the Reward and new position
- Outcome: the robot learns to navigate the maze efficiently over time



Applications of Reinforcement Learning

- Gaming – teaching AI to play complex games (e.g., Chess)
- Robotics – navigating and manipulating objects in dynamic environments
- Finance – portfolio management and trading strategies
- Healthcare – optimizing treatment plans
- Autonomous Vehicles – path planning and decision-making

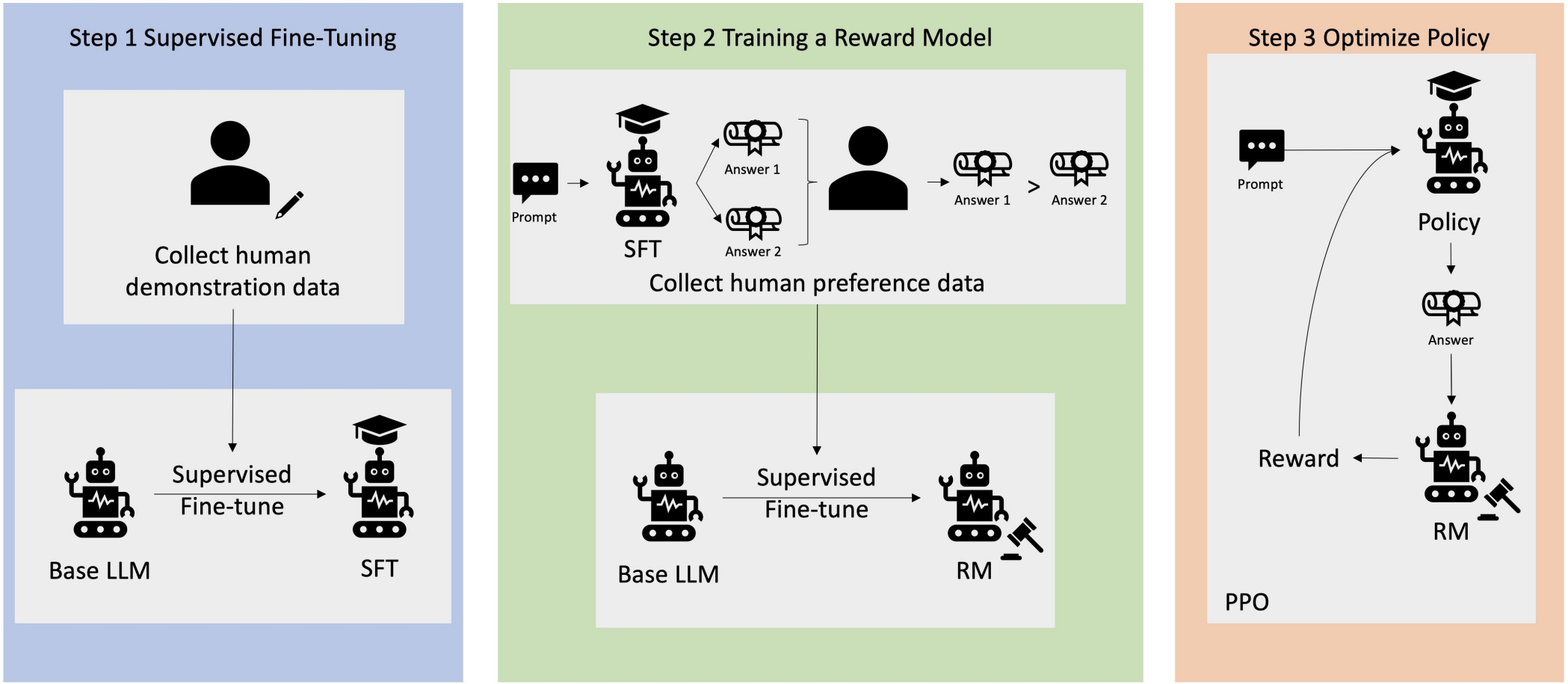
What is RLHF?

- RLHF = Reinforcement Learning from Human Feedback
- Use human feedback to help ML models to self-learn more efficiently
- In Reinforcement Learning there's a reward function
- RLHF incorporates human feedback in the reward function, to be more aligned with human goals, wants and needs
  - **What is RLHF?**assess the quality of the model's responses
- RLHF is used throughout GenAI applications including LLM Models
- RLHF significantly enhances the model performance
- Example: grading text translations from "technically correct" to "human"

RLHF Example:
Internal company knowledge chatbot

- Data collection
  - Set of human-generated prompts and responses are created
  - “Where is the location of the HR department in Boston?”
- Supervised fine-tuning of a language model
  - Fine-tune an existing model with internal knowledge
  - Then the model creates responses for the human-generated prompts
  - Responses are mathematically compared to human-generated answers
- Build a separate reward model
  - Humans can indicate which response they prefer from the same prompt
  - The reward model can now estimate how a human would prefer a prompt response
- Optimize the language model with the reward-based model
  - Use the reward model as a reward function for RL
  - This part can be fully automated

RLHF Process

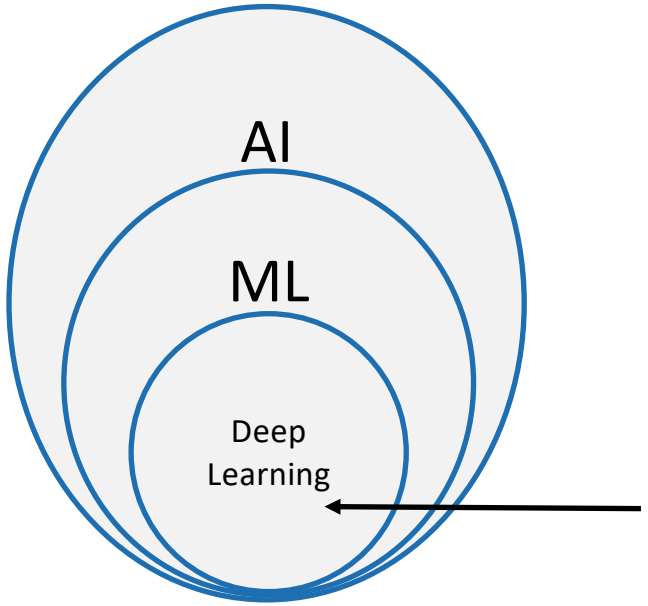


https://aws.amazon.com/what-is/reinforcement-learning-from-human-feedback/

Deep Learning

Deep Learning

Deep learning is a subset of ML



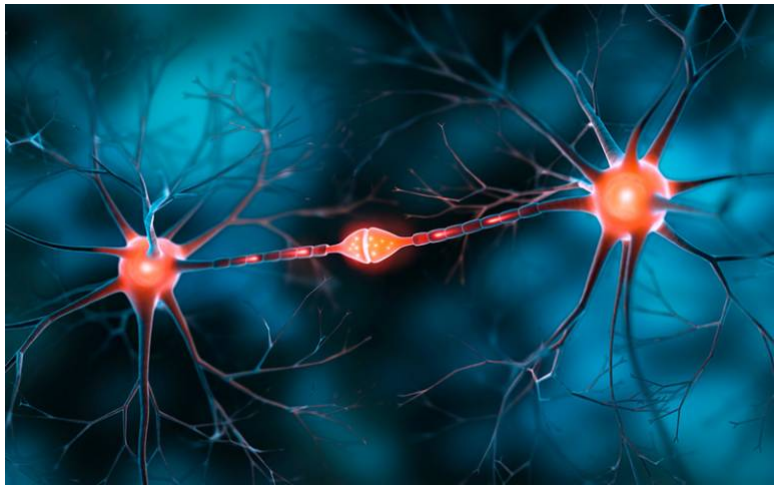
Machine Learning

- Supervised Learning
- Unsupervised Learning
- Reinforcement Learning
- Deep Learning

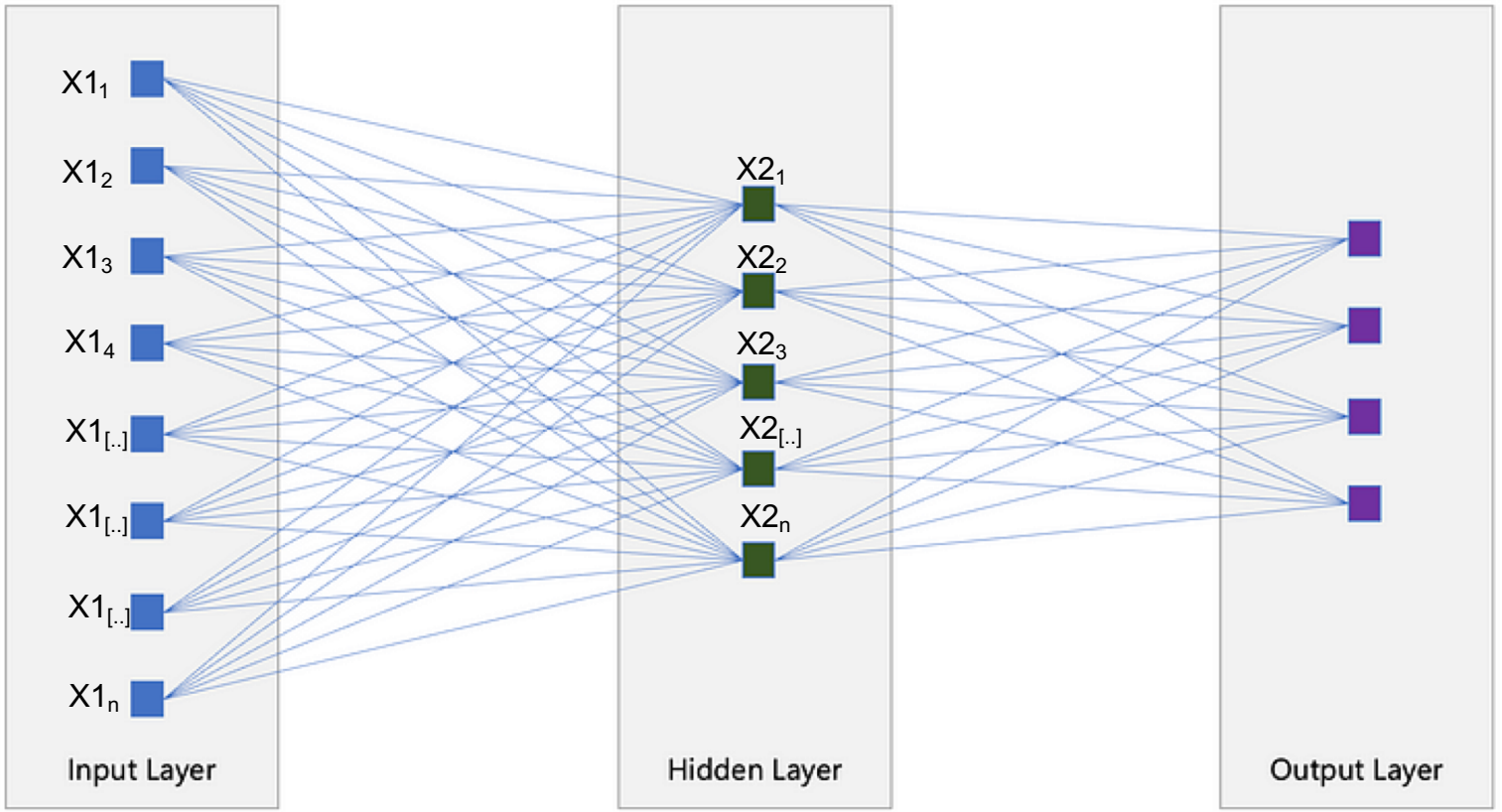
Deep Learning

Deep learning uses Artificial Neural Networks - allowing them to process more complex patterns than traditional machine learning

- Uses neurons and synapses (like our brain) to train a model
- Deep Learning because there's more than one layer of learning
  - Ex: Computer Vision – image classification, object detection, image segmentation
  - Ex: Natural Language Processing (NLP) – text classification, sentiment analysis, machine translation, language generation Large amount of input data Requires GPU (Graphical Processing Unit)



Artificial Neural Networks



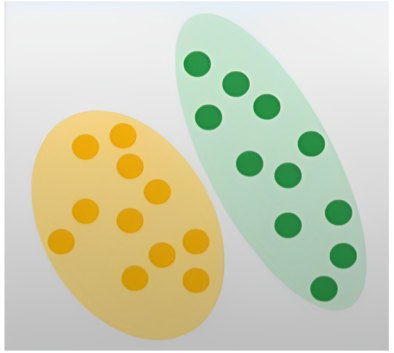
Deep Learning Model Types

Discriminative

- Used to classify or predict
- Typically trained on a dataset of labeled data
- Learns the relationship between the features of the data points and the labels

Generative

- Used to classify or predict
- Typically trained on a dataset of labeled data
- Learns the relationship between the features of the data points and the labels



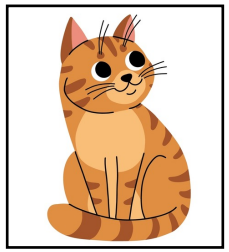
Deep Learning Model Types

Discriminative Technique

Classify

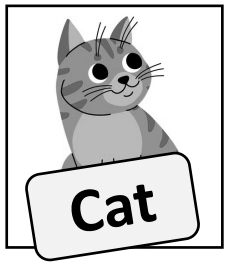
Discriminative Technique (classify as a dog or a cat)

Discriminative Technique



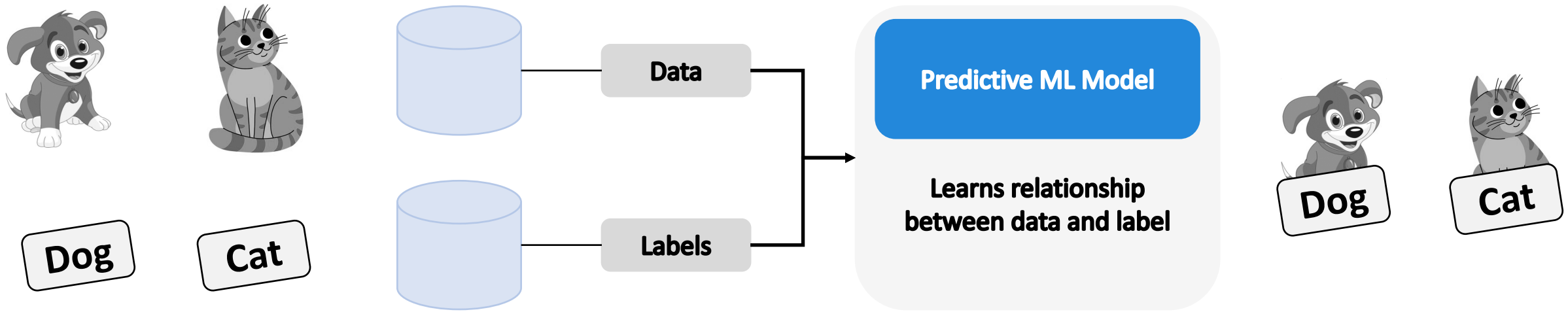
Classify

Discriminative Technique (classify as a dog or a cat)



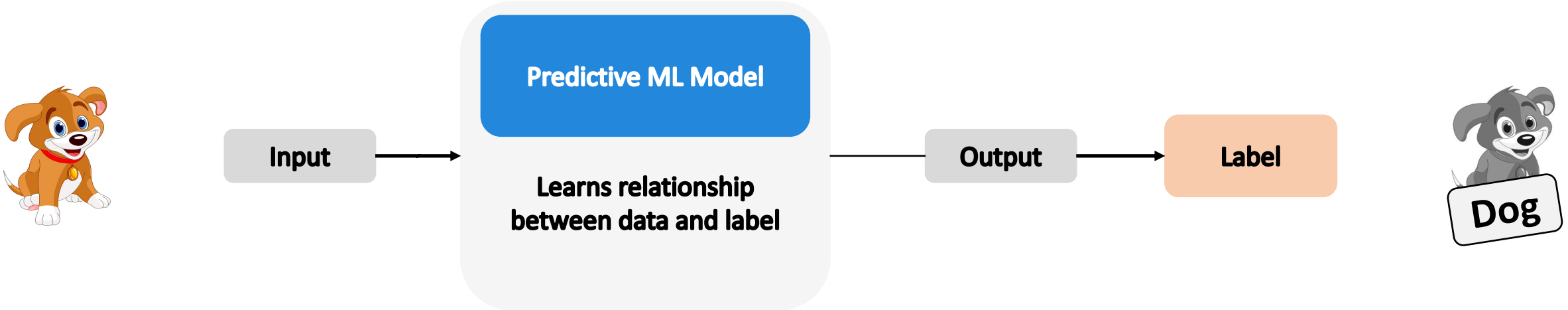
Deep Learning Model Types

Training Phase:



Deep Learning Model Types

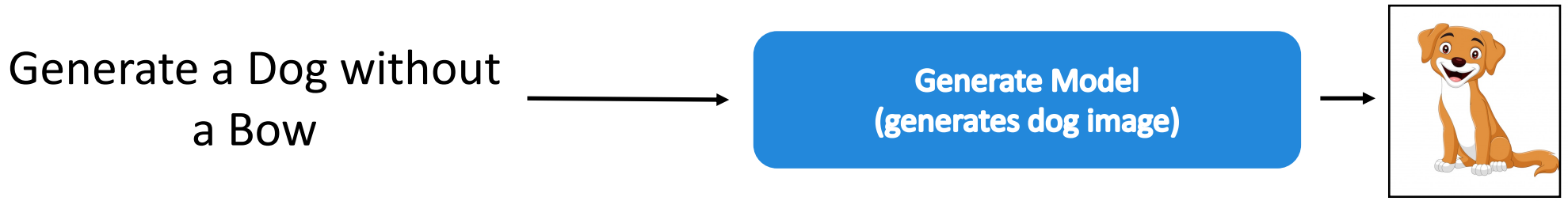
Execution Phase:



Deep Learning Model Types

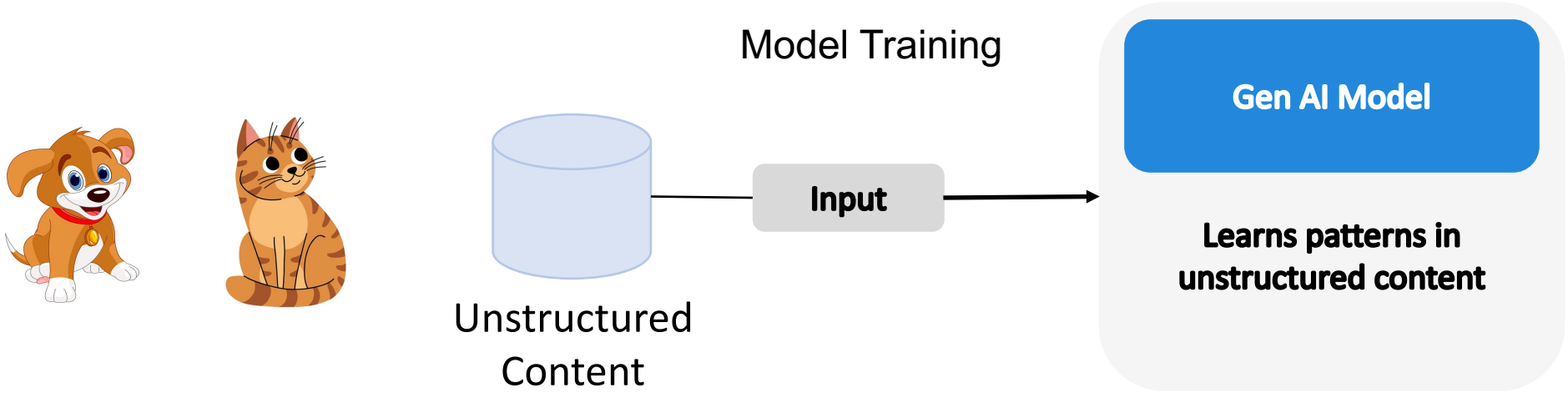
Generate Technique

Generate a Dog without a Bow



Deep Learning Model Types

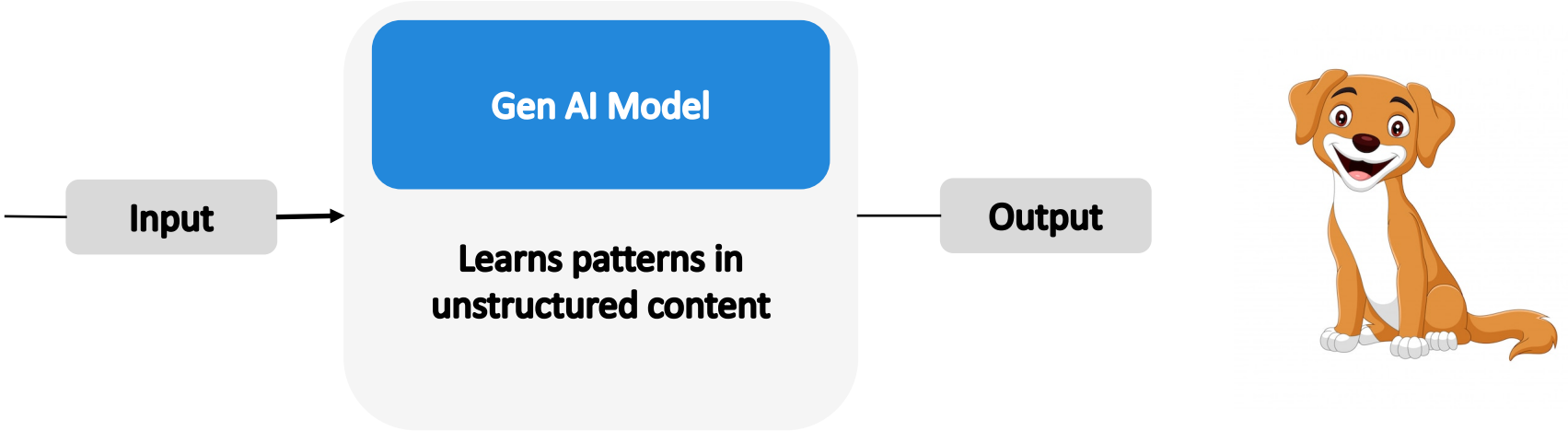
Training Phase:



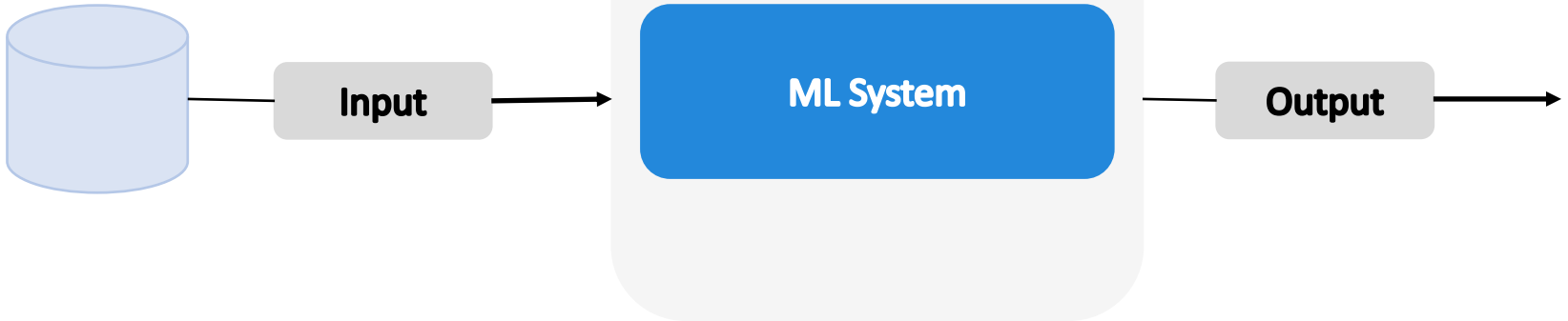
Deep Learning Model Types

Execution Phase:

Generate a dog without bow



ML System Processing



Not GenAI when y is a:

- Number
- Discrete
- Class
- Probability

Is GenAI when y is:

- Natural language
- Image
- Audio

Training Data

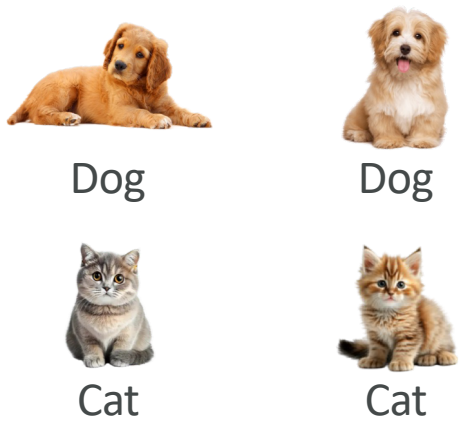
Training Data

- To train our model we must have good data
- Garbage in => Garbage out
- Most critical stage to build a good model
- Several options to model our data, which will impact the types of algorithms we can use to train our models
- Labeled vs. Unlabeled Data
- Structured vs. Unstructured Data

Labeled vs Unlabeled Data

Labeled Data

- Data includes both input features and corresponding output labels
- Example: dataset with images of animals where each image is labeled with the corresponding animal type (e.g., cat, dog)
- Use case: Supervised Learning, where the model is trained to map inputs to known outputs



Unlabeled Data

- Data includes only input features without any output labels
- Example: a collection of images without any associated labels
- Use case: Unsupervised Learning, where the model tries to find patterns or structures in the data

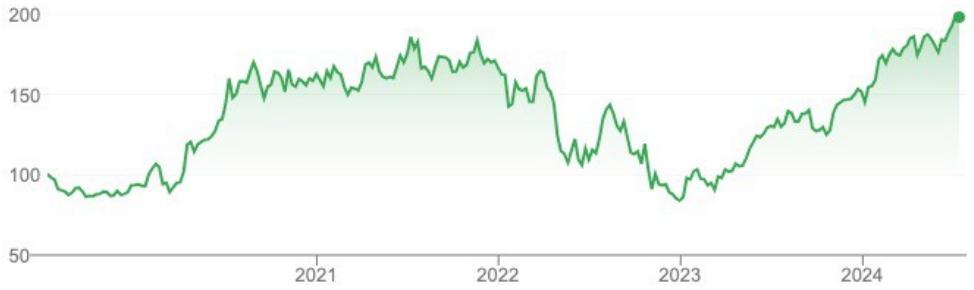
Structured Data

- Data is organized in a structured format, often in rows and columns (like Excel)
- **Tabular Data**
  - Data is arranged in a table with rows representing records and columns representing features
  - Example: customers database with fields such as name, age, and total purchase amount

Customer_ID	Name	Age	Purchase_Amount
1	Alice	30	$200
2	Bob	45	$300

Structured Data

- Data is organized in a structured format, often in rows and columns (like Excel)
- **Time Series Data**
  - Data points collected or recorded at successive points in time
  - Example: Stock prices recorded daily over a year



Date	Stock Price
01-07-2024	$197.20
02-07-2024	$200

Unstructured Data

- Data that doesn't follow a specific structure and is often text-heavy or multimedia content
- Text Data
  - Unstructured text such as articles, social media posts, or customer reviews
  - Example: a collection of product reviews from an e-commerce site
- Image Data
  - Data in the form of images, which can vary widely in format and content
  - Example: images used for object recognition tasks

Review: Attended a yoga class at the new studio. The instructor was excellent, and the facility was well-maintained. Loved the variety of classes offered. Only downside was the parking situation.

GAI – Use Case

Real Estate Trained Model

Data Set

Home prices

size of house (square feet)	# of bedrooms	# of bathrooms	newly renovated	price (1000$)
523	1	2	N	115
645	1	3	N	150
708	2	1	N	210
1034	3	3	Y	280
2290	4	4	N	355
2545	4	5	Y	440

Deep learning Model Training

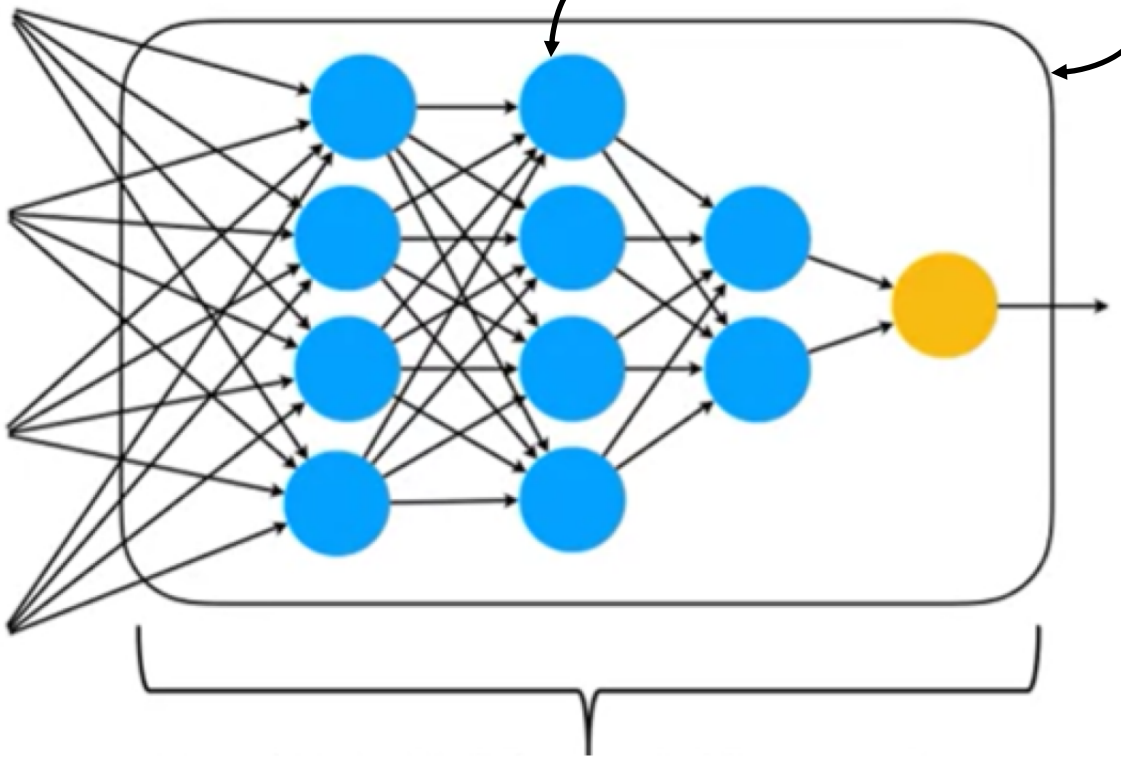
Features:

size

# of bedrooms

# of bathrooms

newly renovated



Neural networks were originally inspired by the brain, but the details of how they work are almost completely unrelated to how biological brains work.

Question (Prompt)

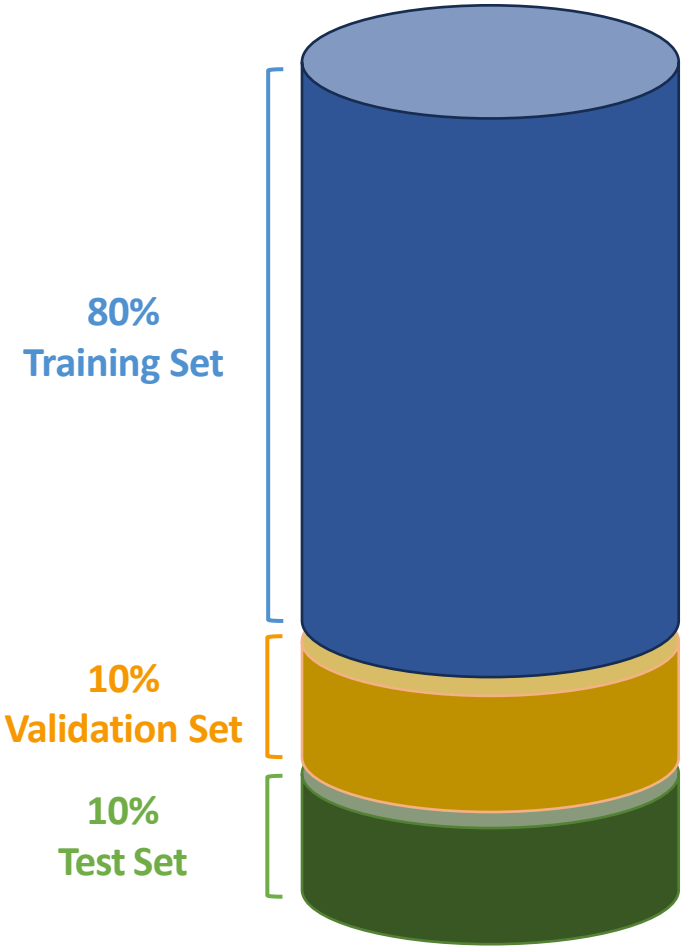
I am looking to buy a 2-bedroom house. how much can I expect to pay for a newly renovated 2-bedroom, 1-bathroom house?

Model Response

Based on the provided Dataset, for a newly renovated 2-bedroom, 1-bathroom house, you can expect to pay approximately $210,000.

Training vs Validation vs Test Set

- Training Set
  - Used to train the model
  - Percentage: typically, 60-80% of the dataset
  - Example: 800 labeled images from a dataset of 1000 images
- Validation Set
  - Used to tune model parameters and validate performance
  - Percentage: typically, 10-20% of the dataset
  - Example: 100 labeled images for hyperparameter tuning (tune the settings of the algorithm to make it more efficient)
- Test Set
  - Used to evaluate the final model performance
  - Percentage: typically, 10-20% of the dataset
  - Example: 100 labeled images to test the model's accuracy

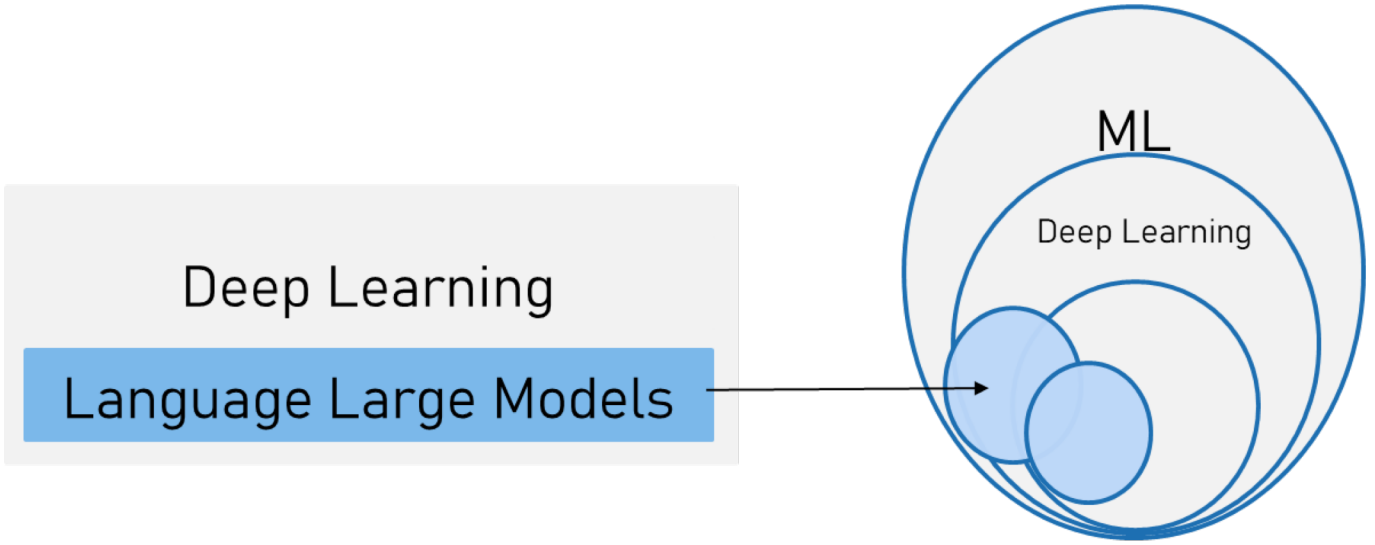


What is Generative AI?

- GenAI is a type of Artificial Intelligence that creates new content based on what it has learned from existing content.
- The process of learning from existing content is called training and results in the creation of a statistical model (LLM).
- When given a prompt, GenAI uses this statistical model (LLM) to predict what an expected response might be-and this generates new content.

Large Language Models

- Large language models are advanced AI systems with **billions of parameters** trained on **extensive datasets**, capable of **understanding and generating human-like** text across various languages and applications.



LLM Evolution

Why LLM models are big?

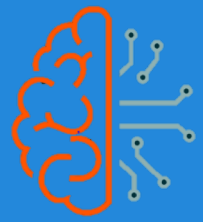
Llama 13B Model Card

- CCNet [67%]
- C4 [15%]
- GitHub [4.5%]
- Wikipedia [4.5%]
- Books [4.5%]
- ArXiv [2.5%]
- Stack
- Exchange[2%]
- 20 languages
- 2 Trillion Tokens

COMMON CRAWL

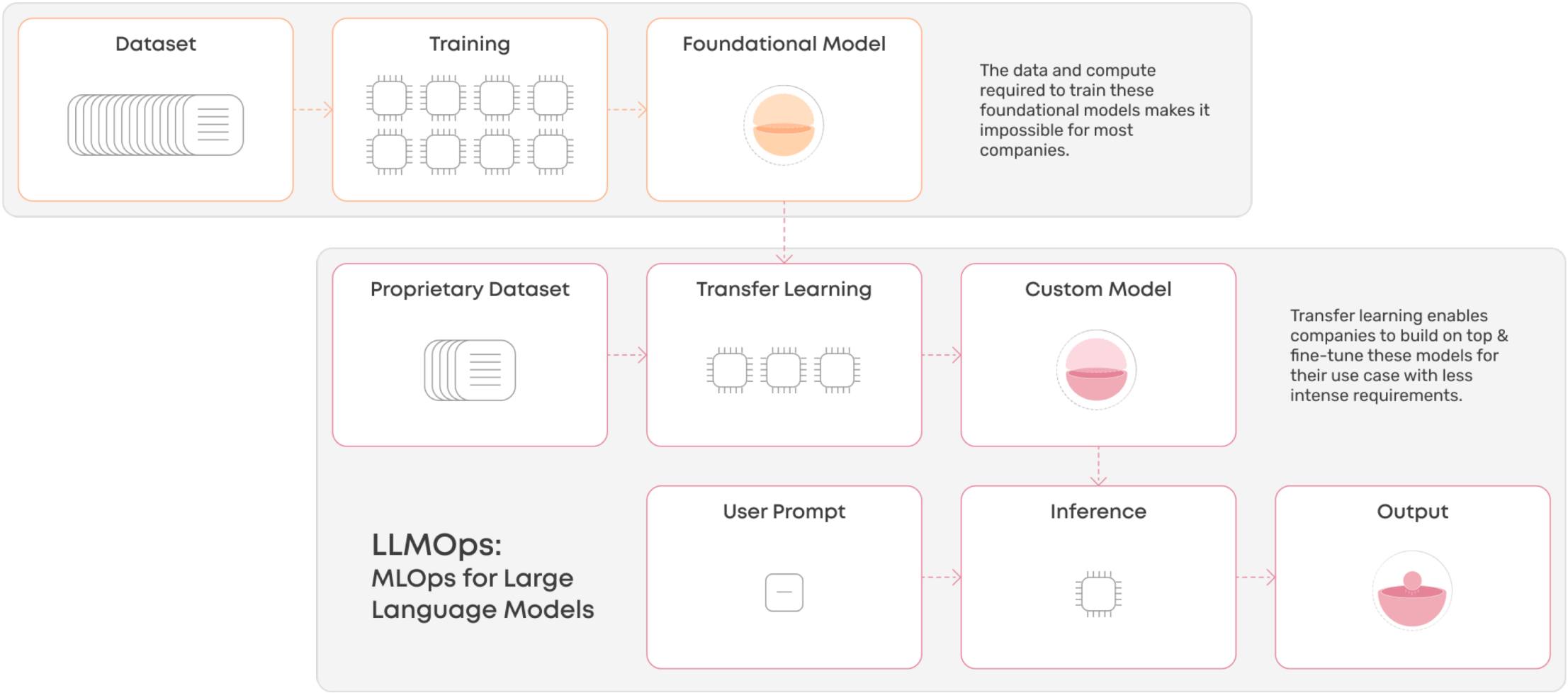
- Open source since 2007, available till June 2023
- 3 - 5 billion new pages added every month
- Every LLM training this corpus
- It contains 3.1 billion web pages or 370 TiB
- Page captures are from 44 million hosts or 35 million registered domains
- 82% raw tokens used in GPT-3

Types of LLM Models



API	Open Source	Proprietary
• No or limited visibility into the code and data, but easier to deploy	• Offer the most transparency and flexibility, but require time and expertise to setup	• Built in-house from scratch for the organization
Bard OpenAI	LLaMA	Bloomberg

How LLM created & Fine-tuned?

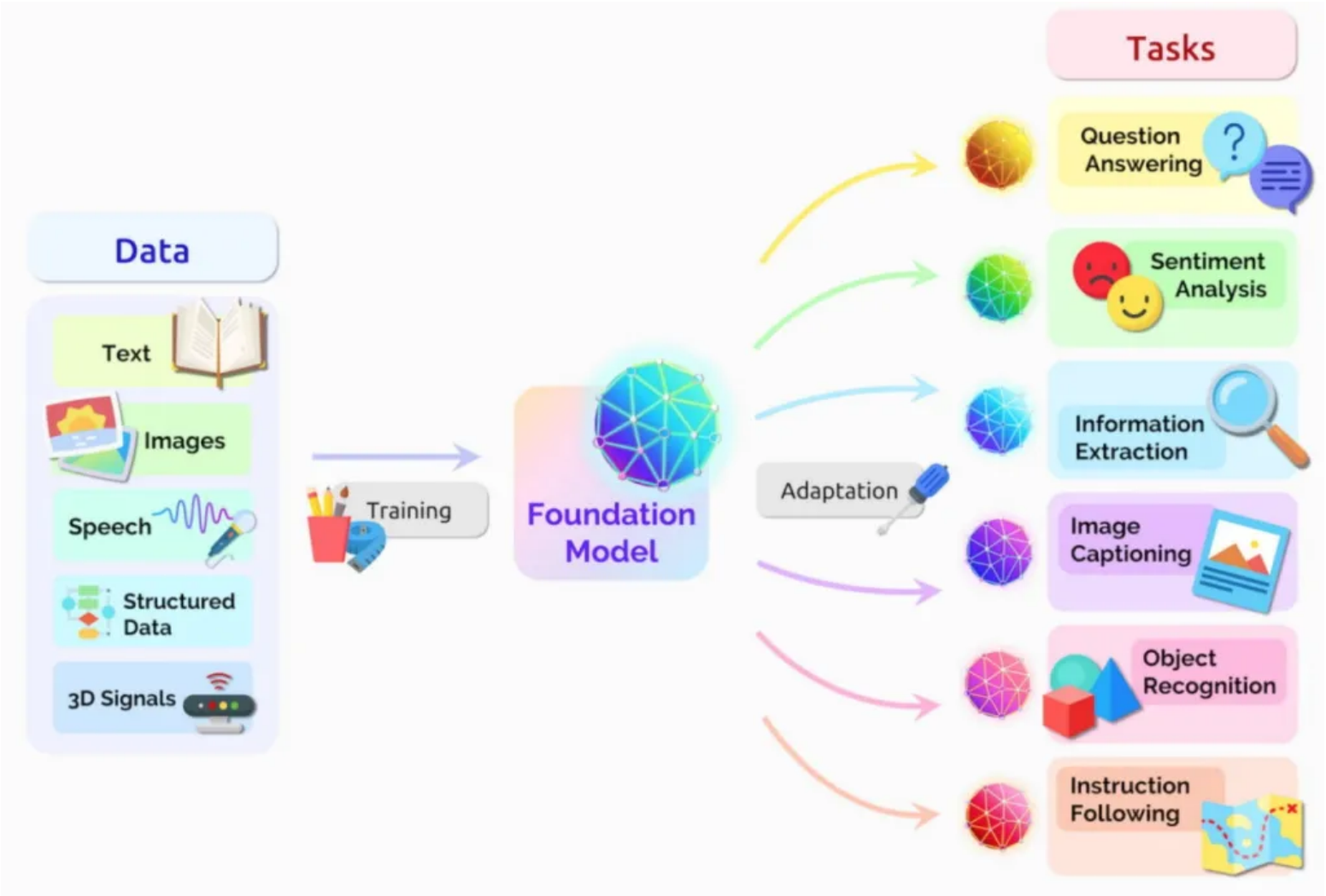


LLM

- LLM is a type of AI algorithm that uses deep learning techniques and massively large data sets to understand, summarize, generate and predict new content
- LLM are commonly used in NLP applications where a user inputs a query in natural language to generate a result.
- Modern LLMs emerged in 2017, used transformer models Some LLMs are referred to as foundation models
- Different architecture in ML history

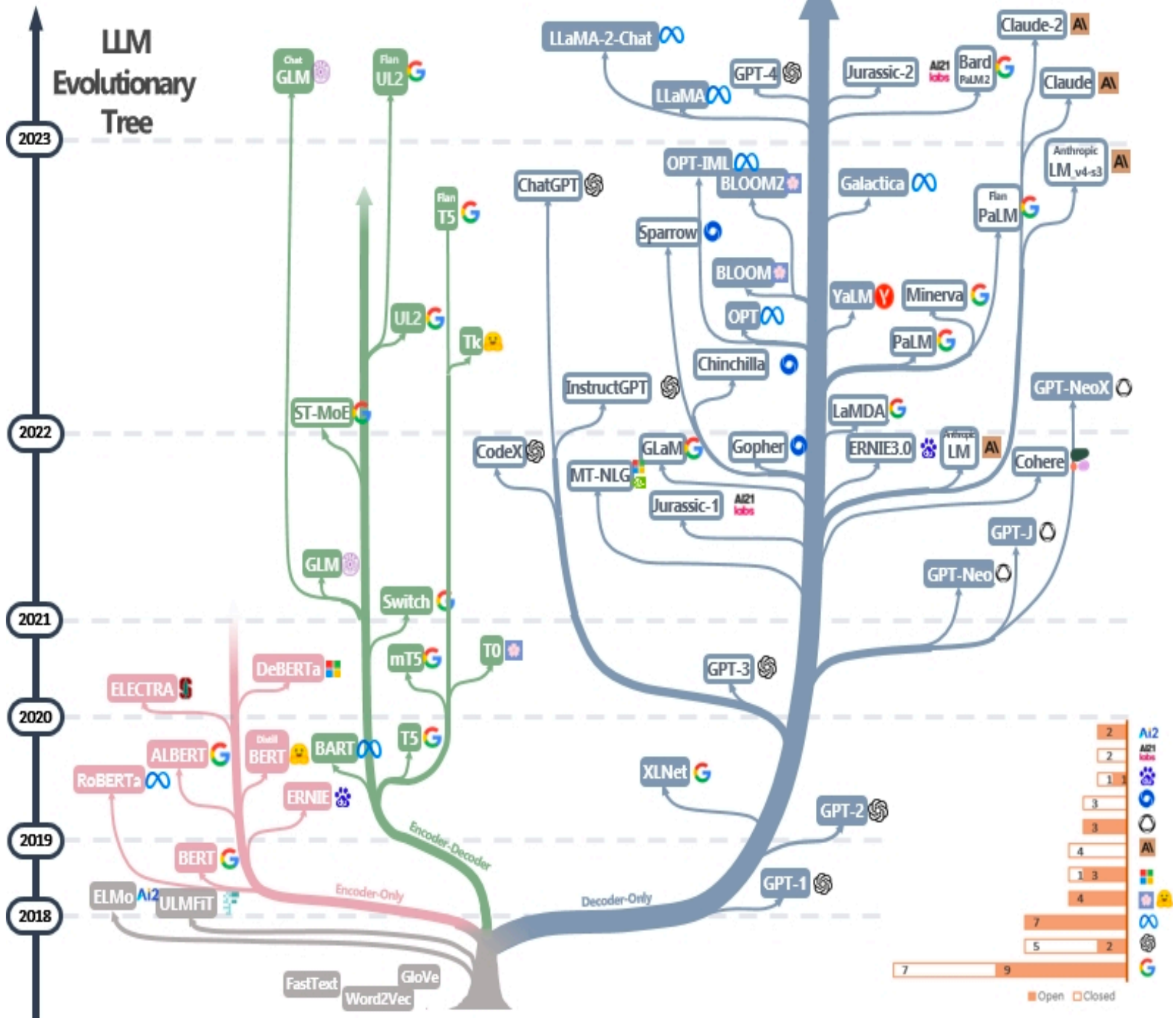
Major Improvements
RNN
LSTM
Transformer
Attention
Transfer Learning
Human Feedback

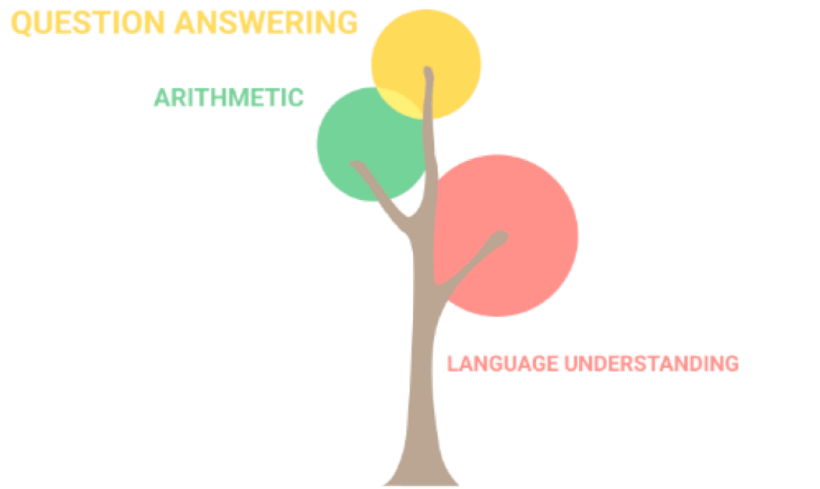
What is Foundational Models



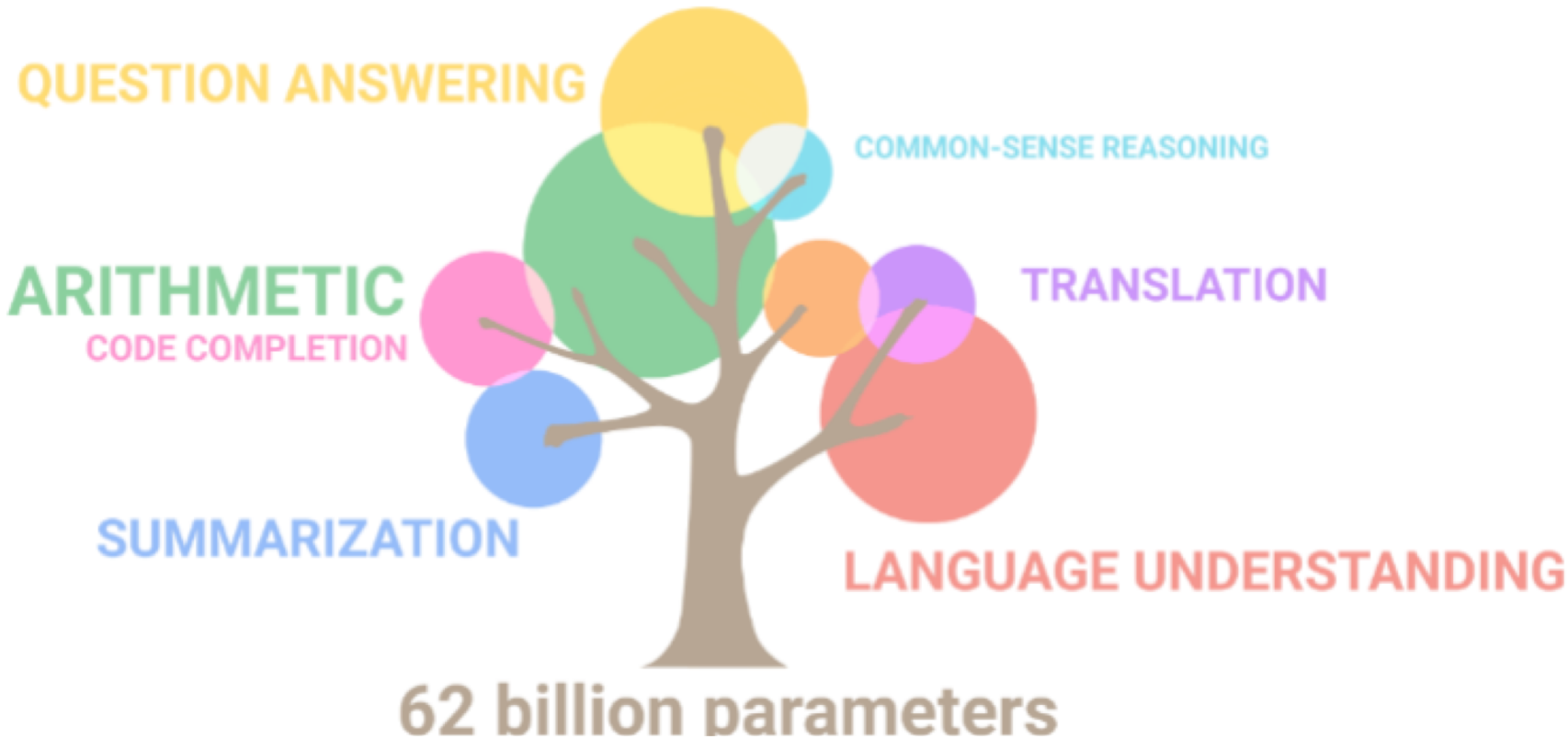
Example Foundation Models

Model	Description
GPT (Generative Pre-trained Transformer)	Autoregressive model by OpenAI for text generation, summarization, and conversation. Forms the basis of ChatGPT.
BERT (Bidirectional Encoder Representations from Transformers)	Google's encoder-only model focused on deep language understanding for tasks like classification and QA.
T5 (Text-to-Text Transfer Transformer)	Google's model that frames all NLP tasks as a text to text problem flexible and powerful for both understanding and generation.
RoBERTa (Robustly Optimized BERT Pretraining Approach)	Facebook's optimized version of BERT with better performance due to improved training strategies and data.
Claude	Autoregressive model by Anthropic focused on safety, alignment, and high-quality dialogue.
LLaMA (Large Language Model Meta AI)	Meta's open-source LLM series designed for research and customization popular in fine-tuning and local deployment.
Jurassic	AI21 Labs' large-scale language model available via API optimized for creative generation and enterprise use.
Cohere	Transformer-based model built for enterprise NLP offers strong multilingual support, embeddings, and retrieval-augmented generation.


LLM Evolutionary Tree
2023
2022
2021
2020
2019
2018
Chat GLM
Flan UL2
LLaMA-2-Chat
GPT-4
LLaMA
Jurassic-2
Bard PaLM2
Claude-2
Claude
Anthropic LM_v4-s3
OPT-IML
BLOOMZ
ChatGPT
Galactica
Flan PaLM
Flan T5
Sparrow
BLOOM
YaLM
Minerva
UL2
OPT
Tk
PaLM
Chinchilla
InstructGPT
GPT-NeoX
LaMDA
ST-MoE
CodeX
GLaM
Gopher
ERNIE3.0
Anthropic LM
Cohere
MT-NLG
Jurassic-1
GPT-J
GLM
GPT-Neo
Switch
mT5
T0
GPT-3
DeBERTa
ELECTRA
T5
ALBERT
Distil BERT
BART
XLNet
RoBERTa
ERNIE
GPT-2
BERT
Encoder-Decoder
Encoder-Only
Decoder-Only
ELMo
ULMFiT
GPT-1
FastText
GloVe
Word2Vec
Open
Closed



8 billion parameters


QUESTION ANSWERING
COMMON-SENSE REASONING
ARITHMETIC
CODE COMPLETION
TRANSLATION
SUMMARIZATION
LANGUAGE UNDERSTANDING
62 billion parameters

QUESTION ANSWERING

SEMANTIC PARSING

PROVERBS

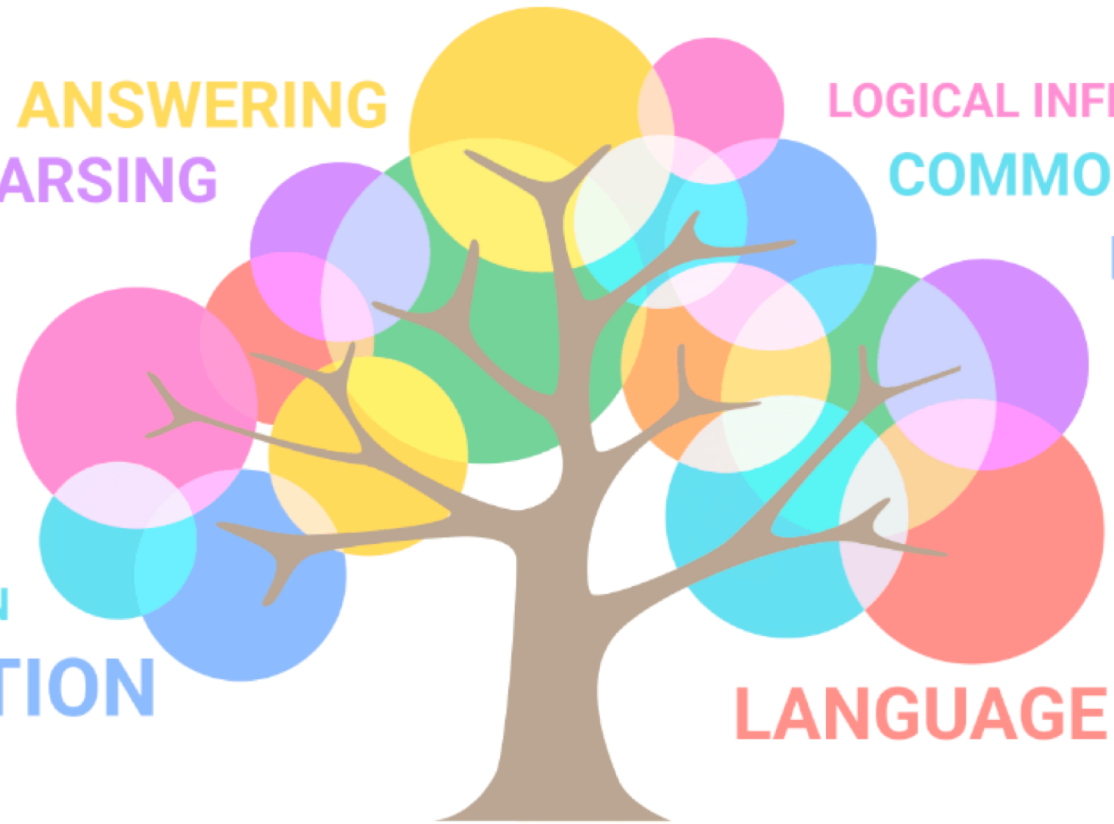
ARITHMETIC

CODE COMPLETION

GENERAL KNOWLEDGE

READING COMPREHENSION

SUMMARIZATION



LOGICAL INFERENCE CHAINS

COMMON-SENSE REASONING

PATTERN RECOGNITION

TRANSLATION

DIALOGUE

JOKE EXPLANATIONS

PHYSICS QA

LANGUAGE UNDERSTANDING

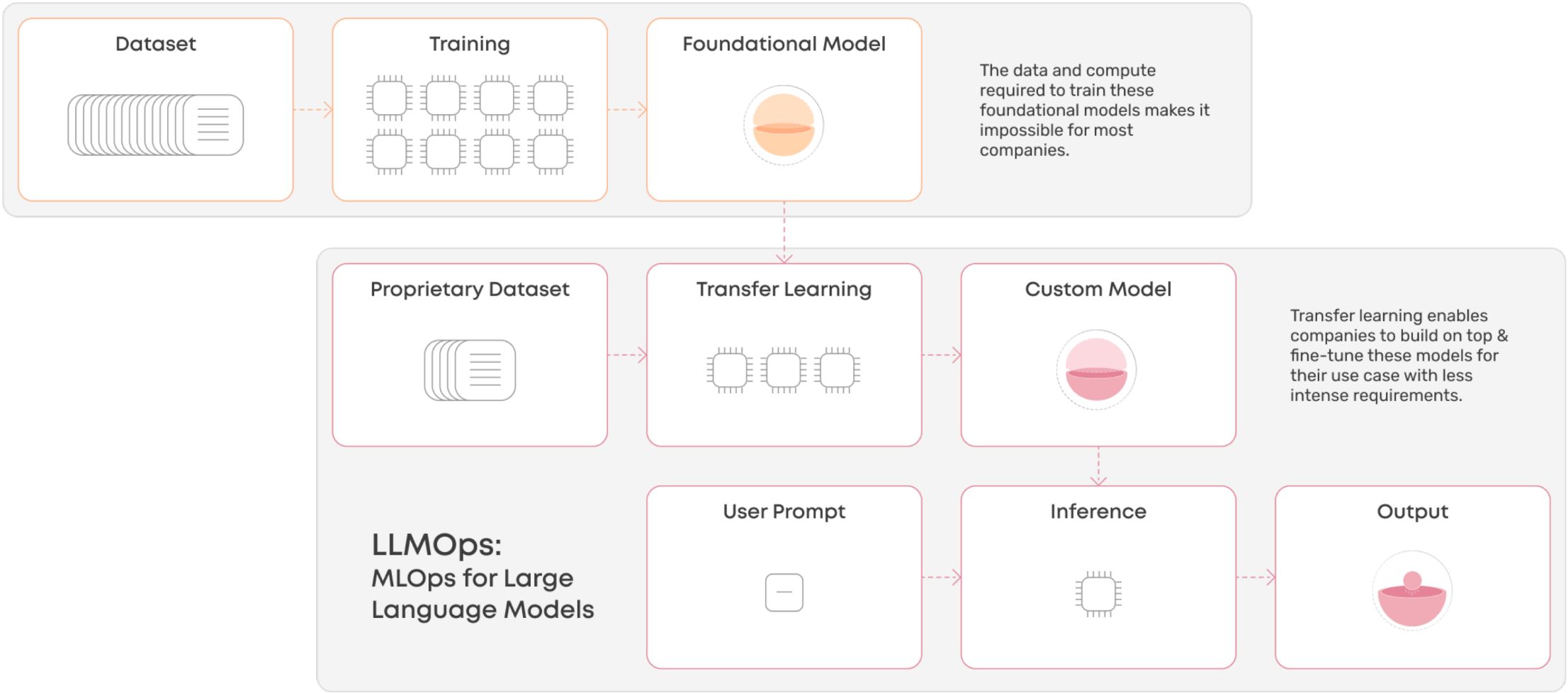
540 billion parameters

The generative AI Application Landscape

Application Layer

Text	Code	Image	Speech	Video	3D	Other
Marketing (content)						
Sales (email)						Gaming
Support (chat/email)	Code Generation	Image generation				RPA
General writing	Code documentation	Consumer/ Social				Music
Note taking	Text to SQL	Media/ Advertising				Audio
Other	Web app builders	Design	Voice Synthesis	Video editing /generation	3D models/ scenes	Biology & Chemistry

How LLM created & Fine-tuned?



Transformers

- A transformer is a deep learning architecture developed by Google and based on the multi-head attention mechanism, proposed in a 2017 paper "Attention Is All You Need".[1] It has no recurrent units, and thus requires less training time than previous recurrent neural architectures, such as long short-term memory (LSTM)

Attention Is All You Need [2017]

Transformers

Attention Is All You Need

Ashish Vaswani* Google Brain avaswani@google.com

Noam Shazeer* Google Brain noam@google.com

Niki Parmar* Google Research nikip@google.com

Jakob Uszkoreit* Google Research usz@google.com

Llion Jones* Google Research llion@google.com

Aidan N. Gomez* † University of Toronto aidan@cs.toronto.edu

Łukasz Kaiser* Google Brain lukaszkaiser@google.com

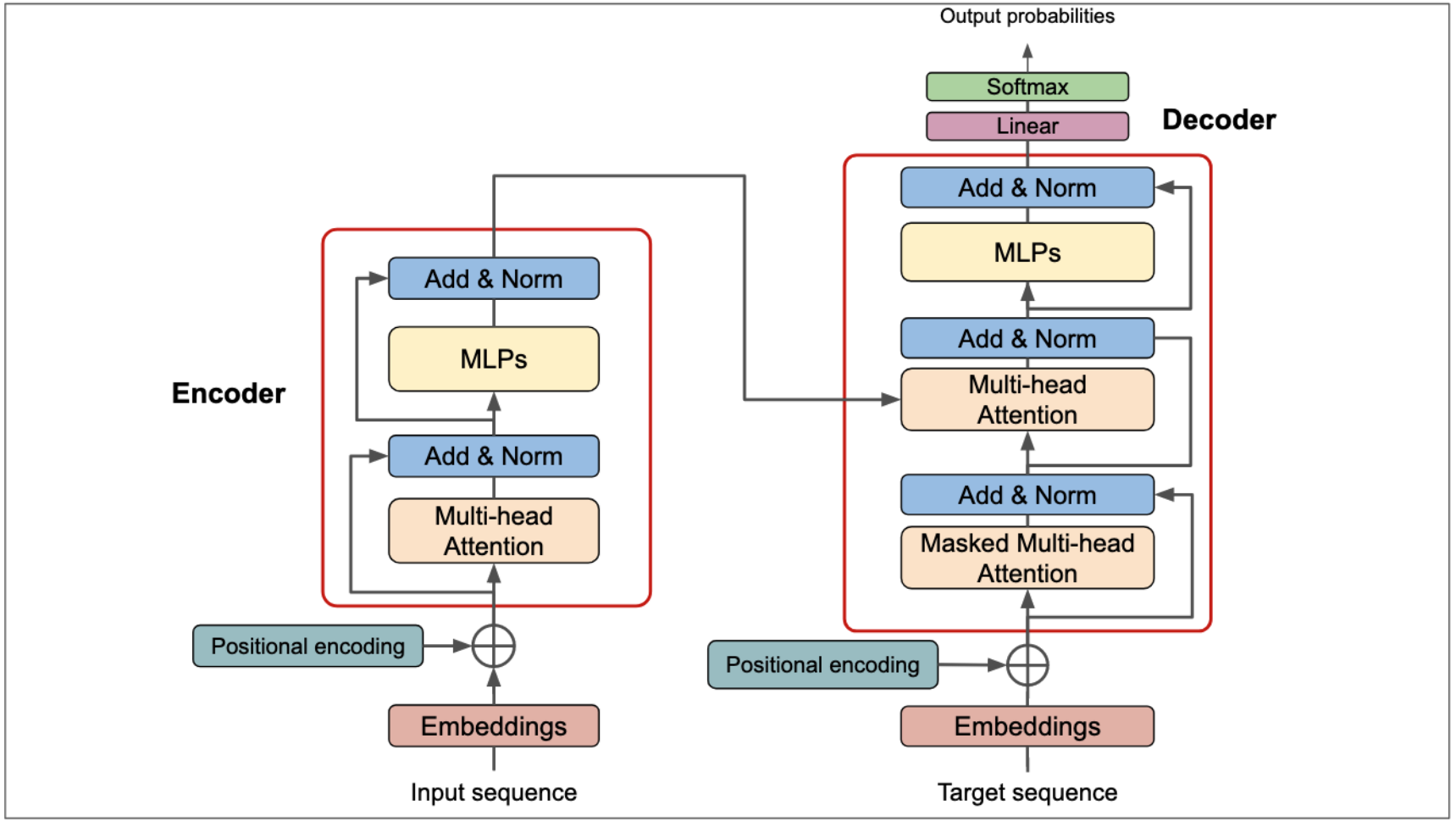
Illia Polosukhin* ‡ illia.polosukhin@gmail.com

Abstract

The dominant sequence transduction models are based on complex recurrent or convolutional neural networks that include an encoder and a decoder. The best performing models also connect the encoder and decoder through an attention mechanism. We propose a new simple network architecture, the Transformer, based solely on attention mechanisms, dispensing with recurrence and convolutions entirely. Experiments on two machine translation tasks show these models to

- Scale efficiently
- Parallel process
- Attention to input meaning

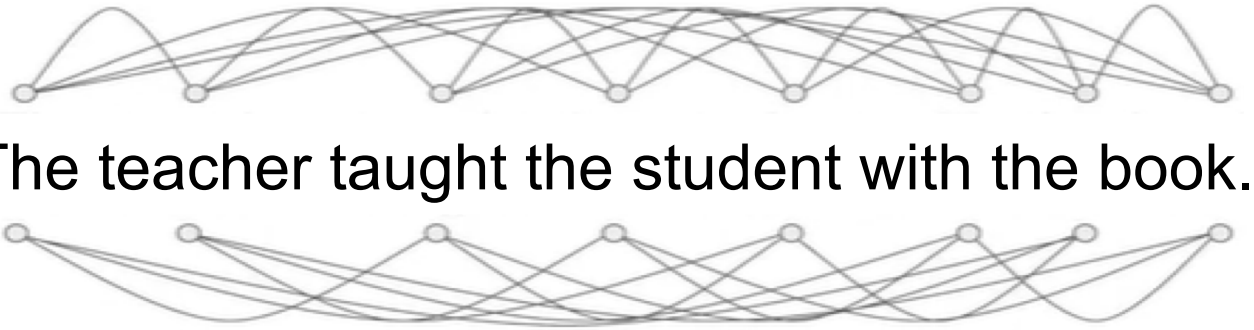
Transformers Architecture



Transformers

The teacher taught the student with the book.

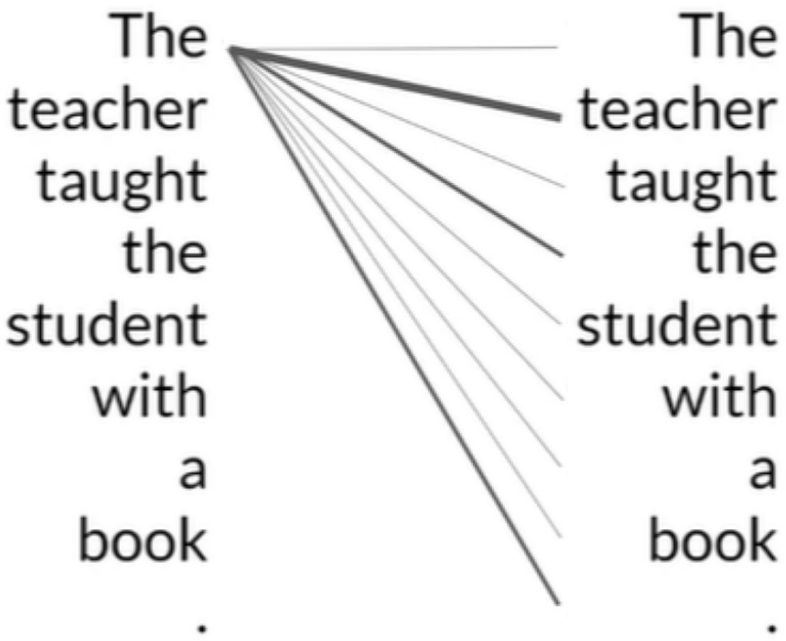
Transformers



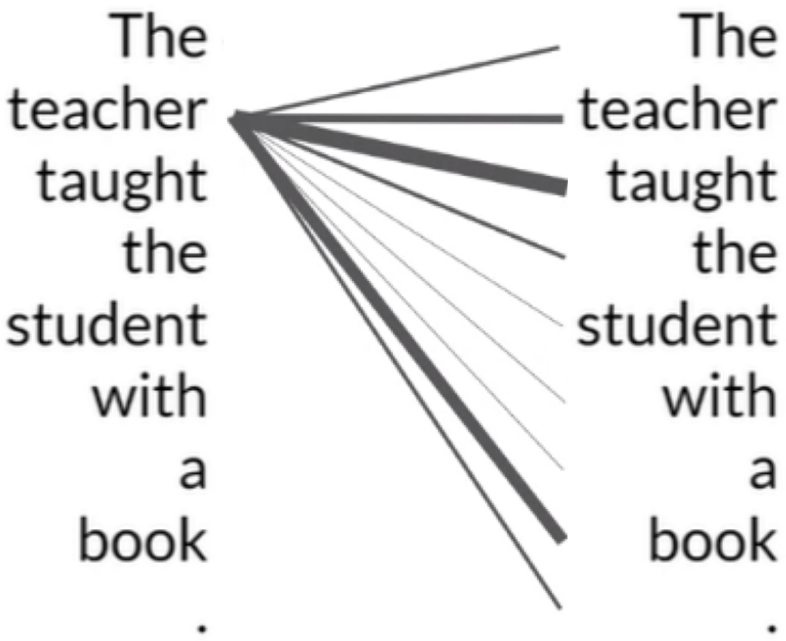
Transformers

The teacher taught the student with the book.

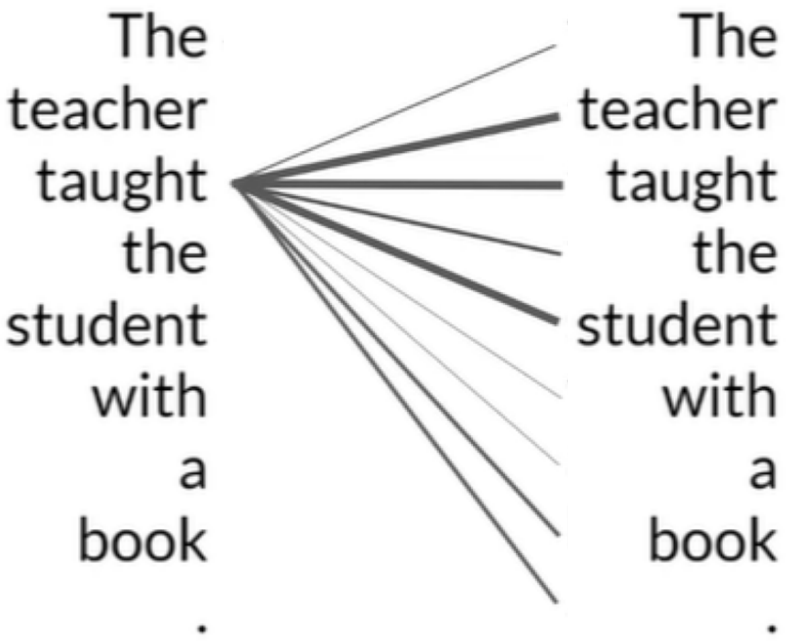
Self-attention



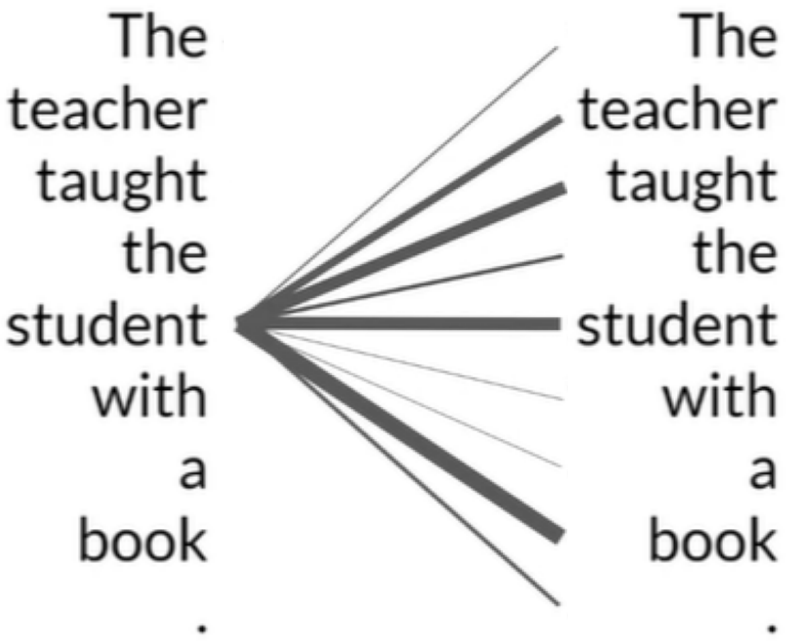
Self-attention



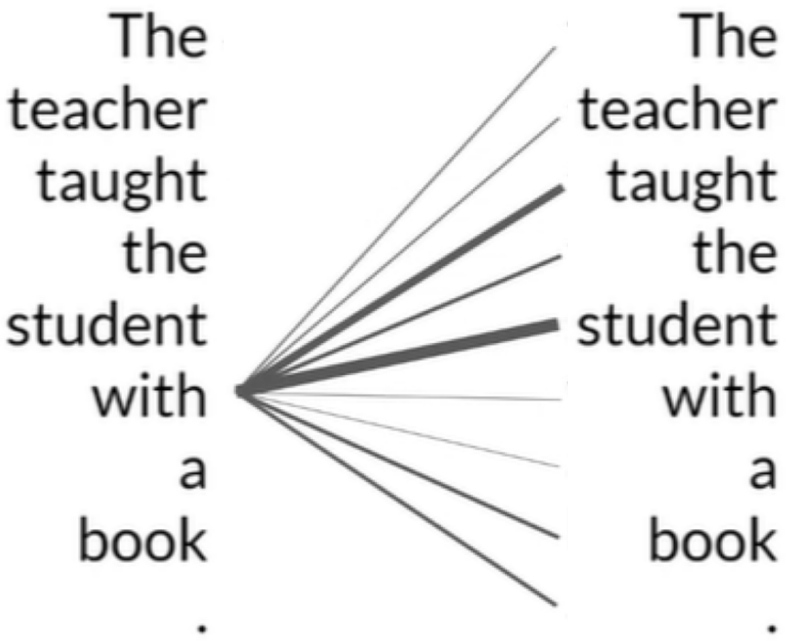
Self-attention



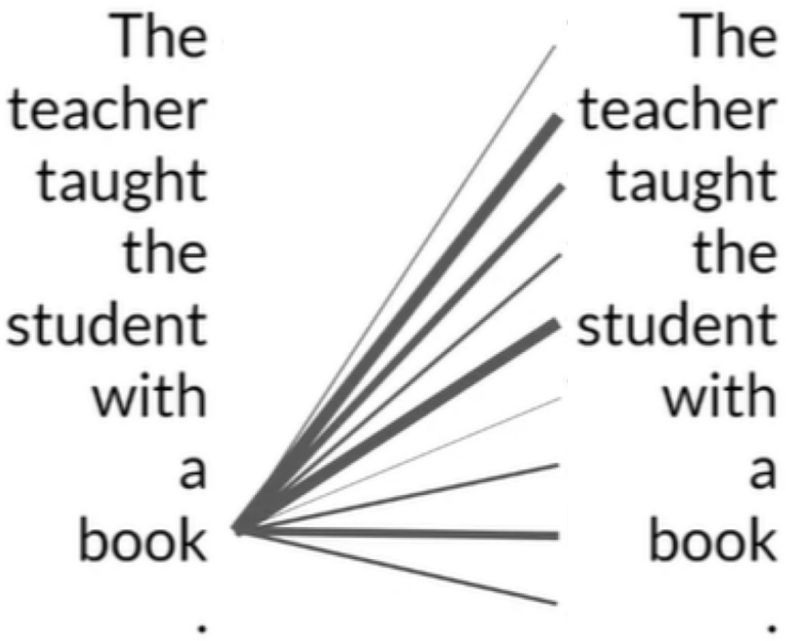
Self-attention



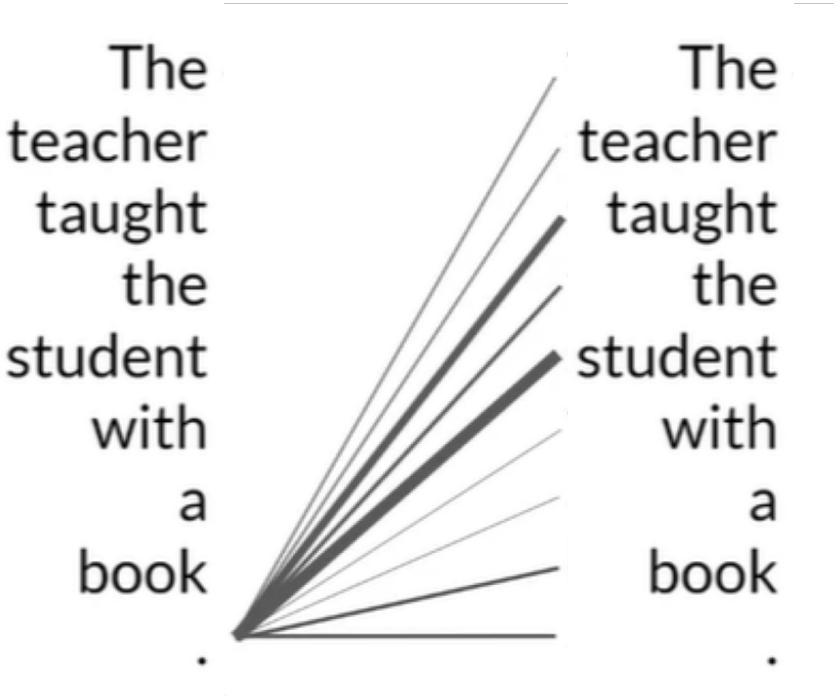
Self-attention



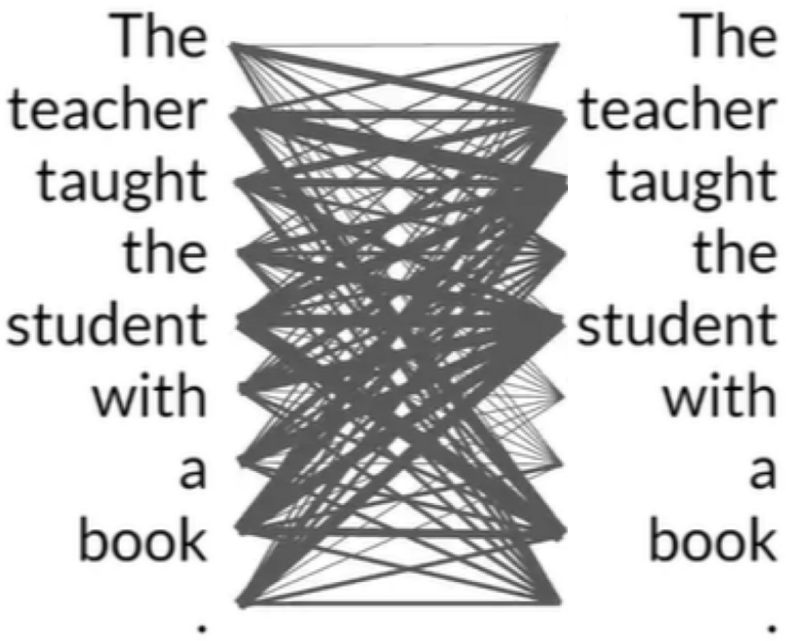
Self-attention



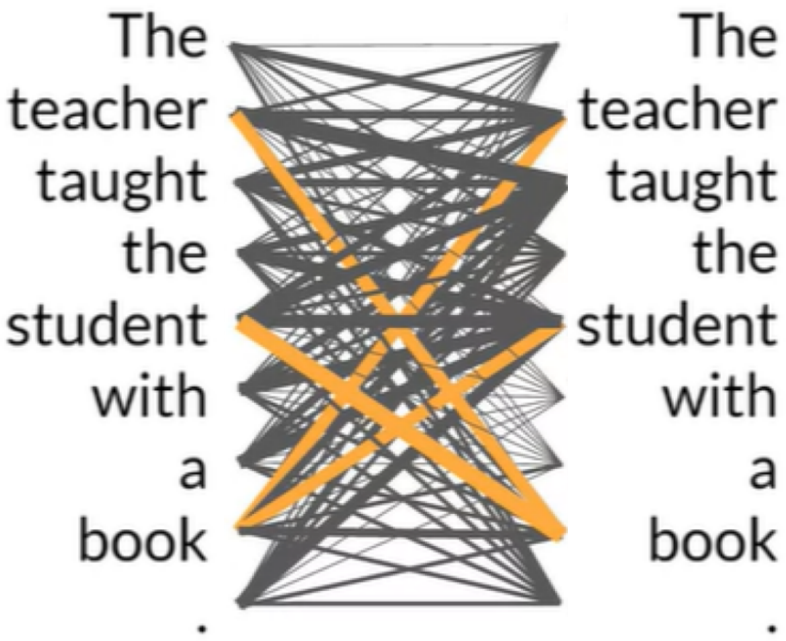
Self-attention



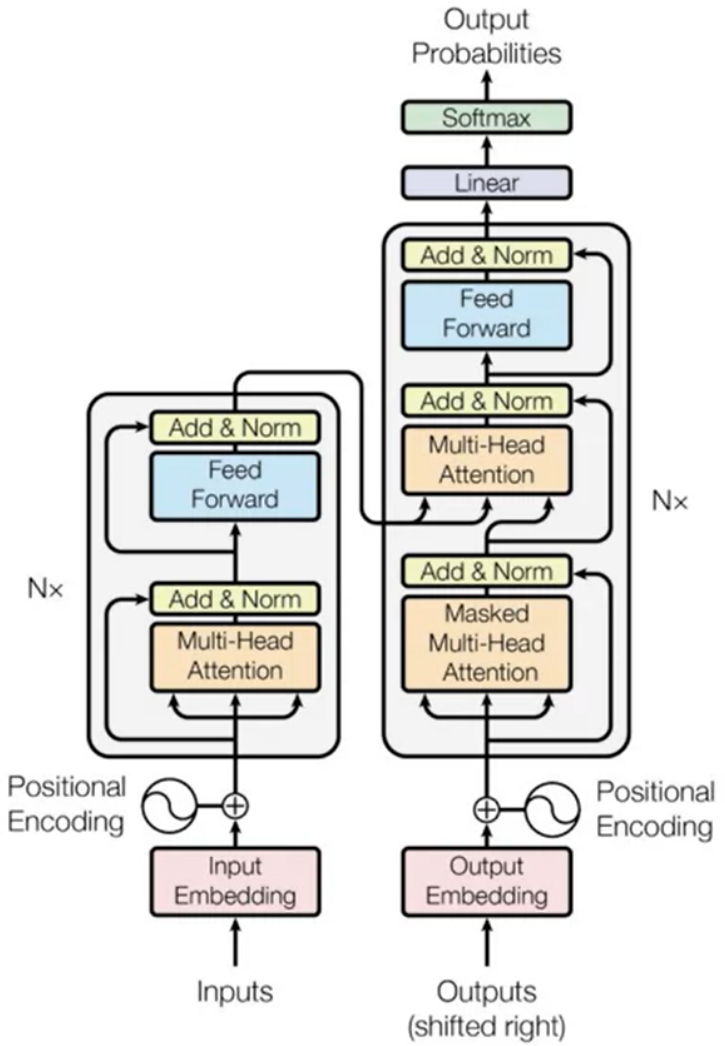
Self-attention



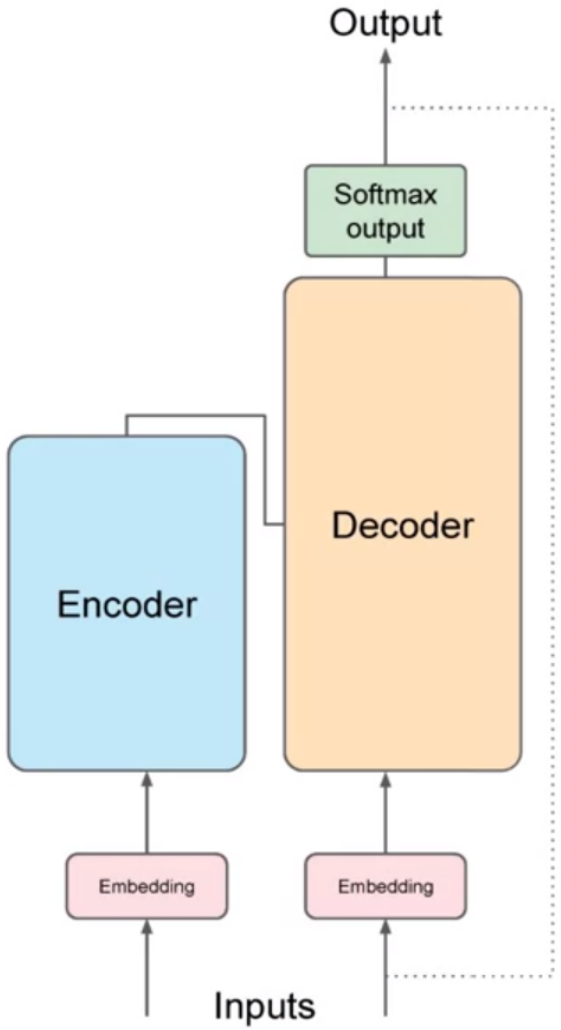
Self-attention



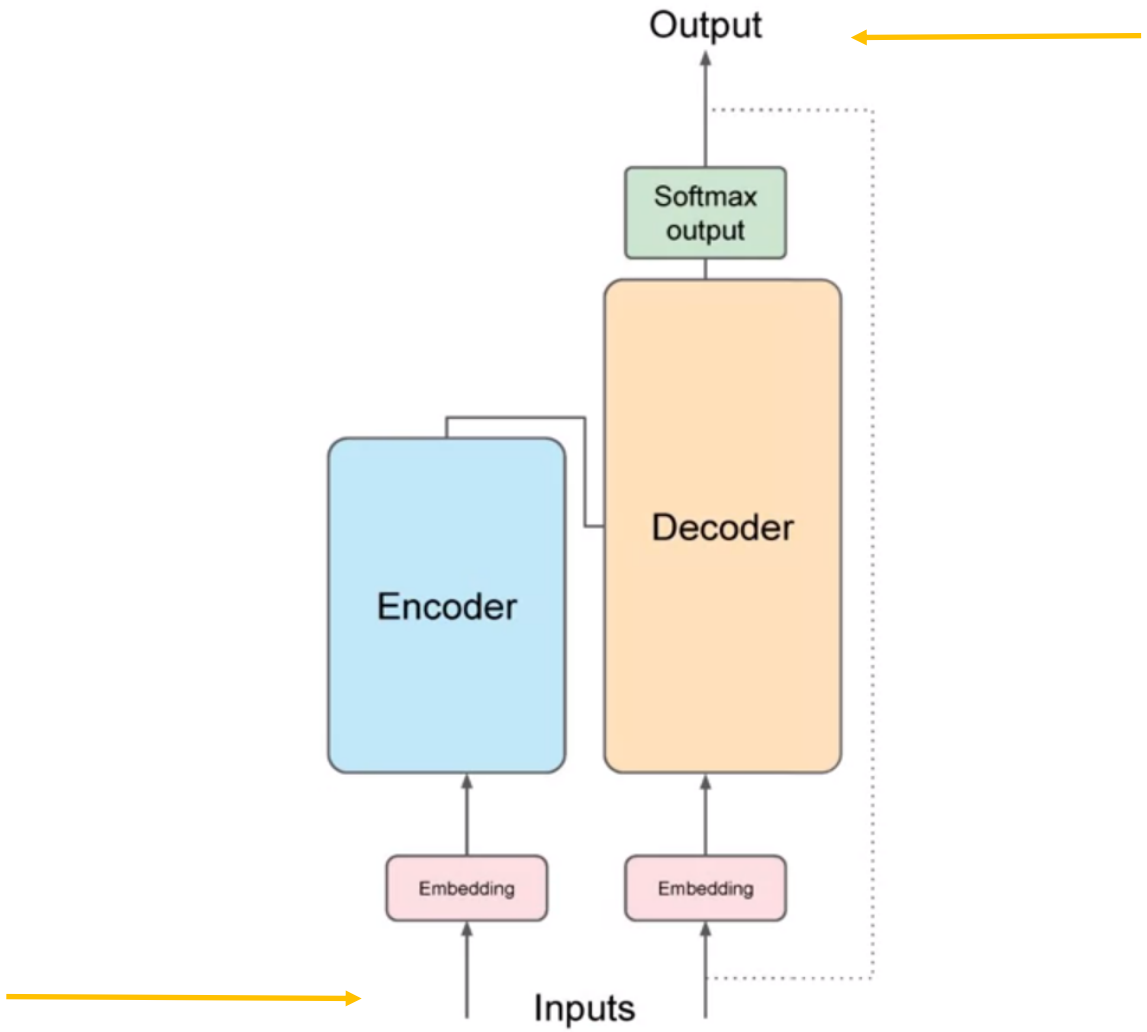
Transformers Architecture



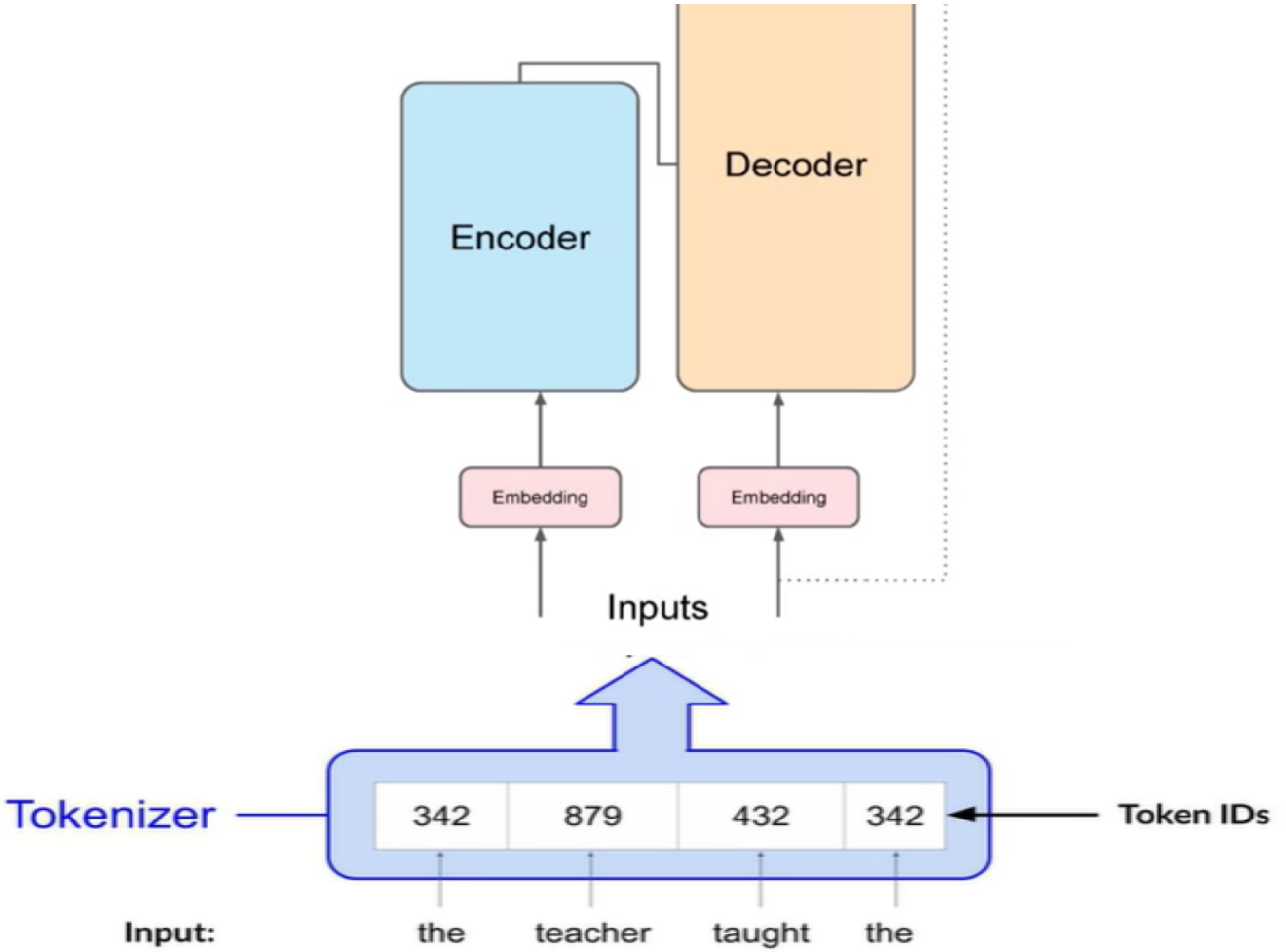
Transformers Architecture



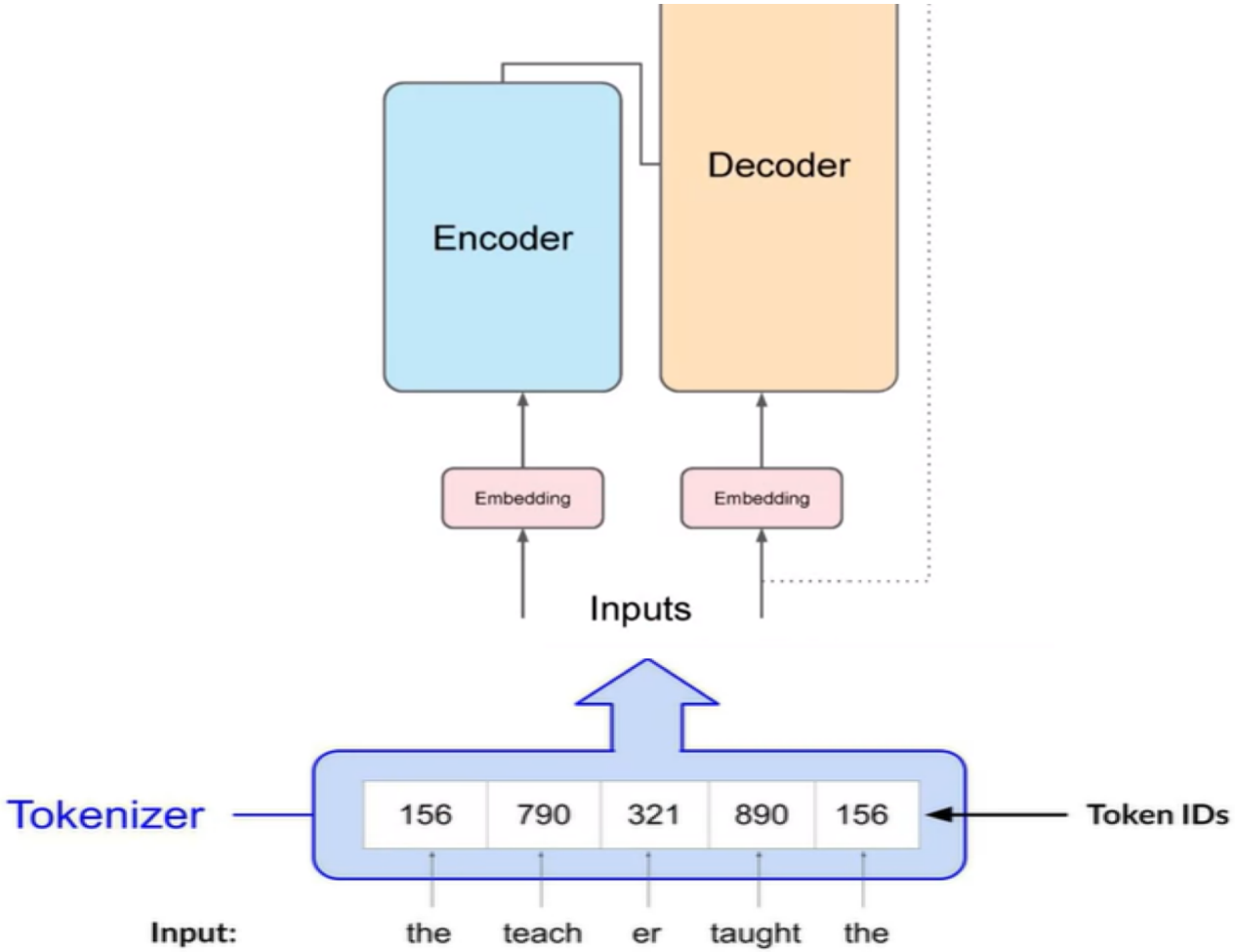
Transformers Architecture



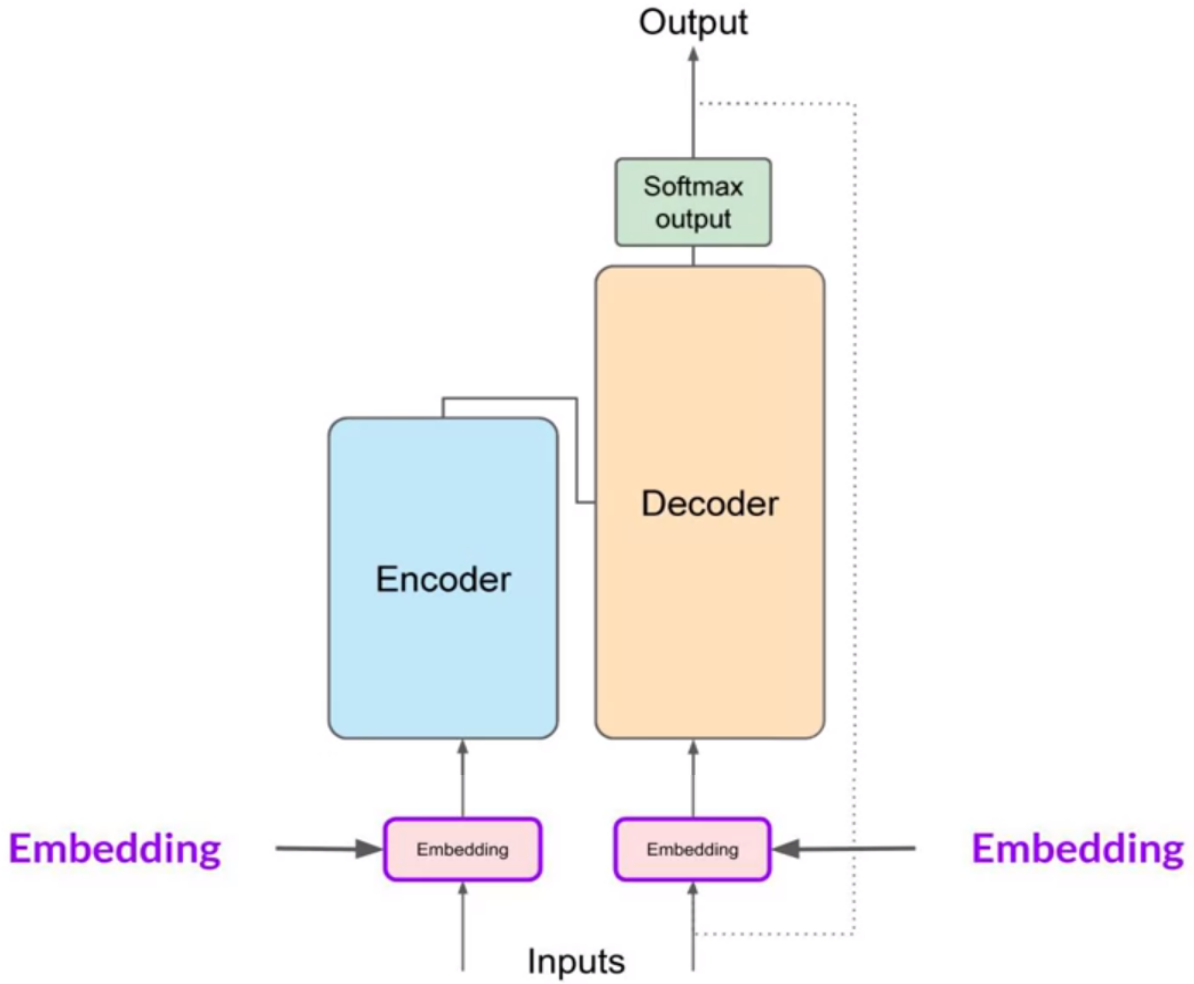
Transformers Architecture



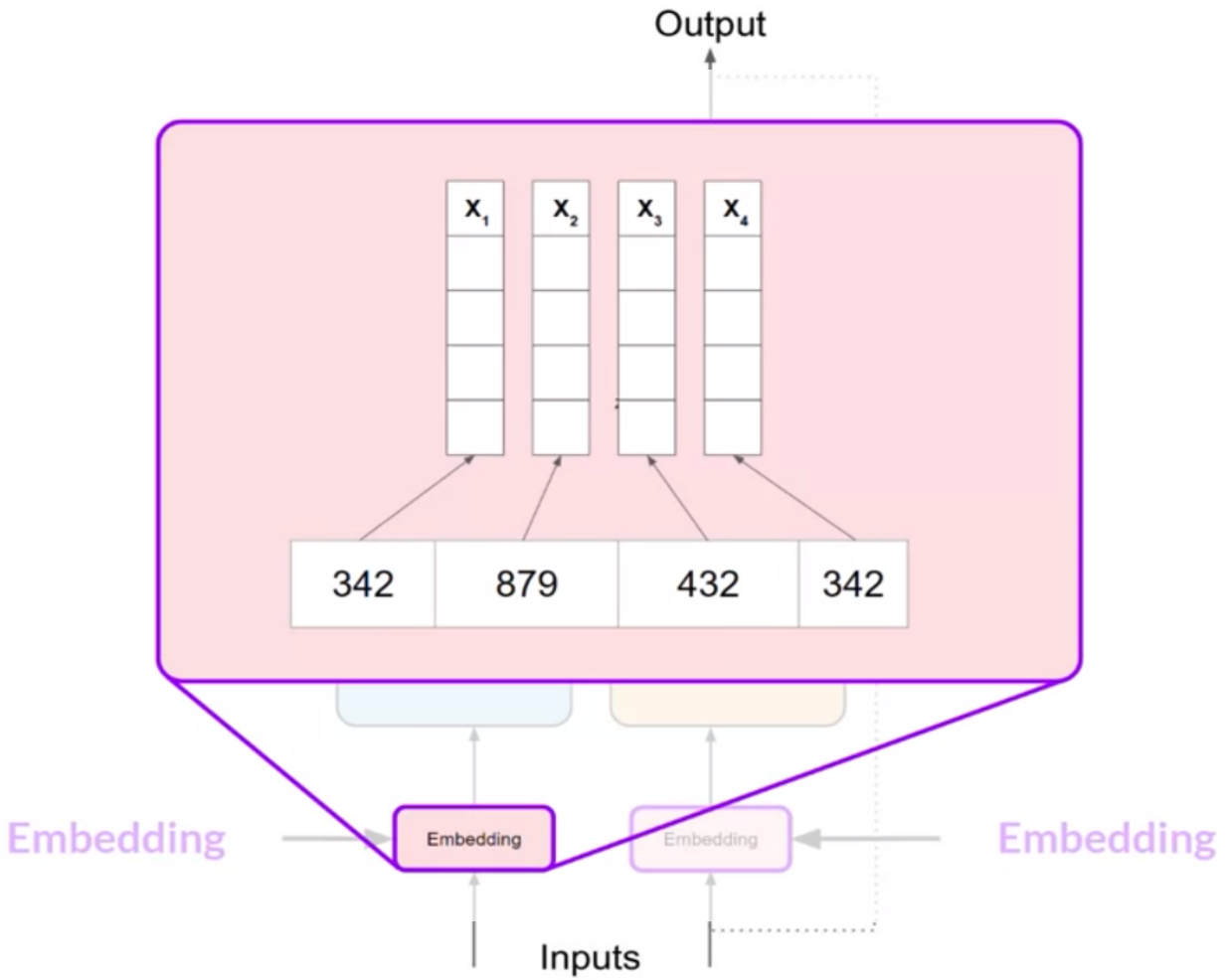
Transformers Architecture



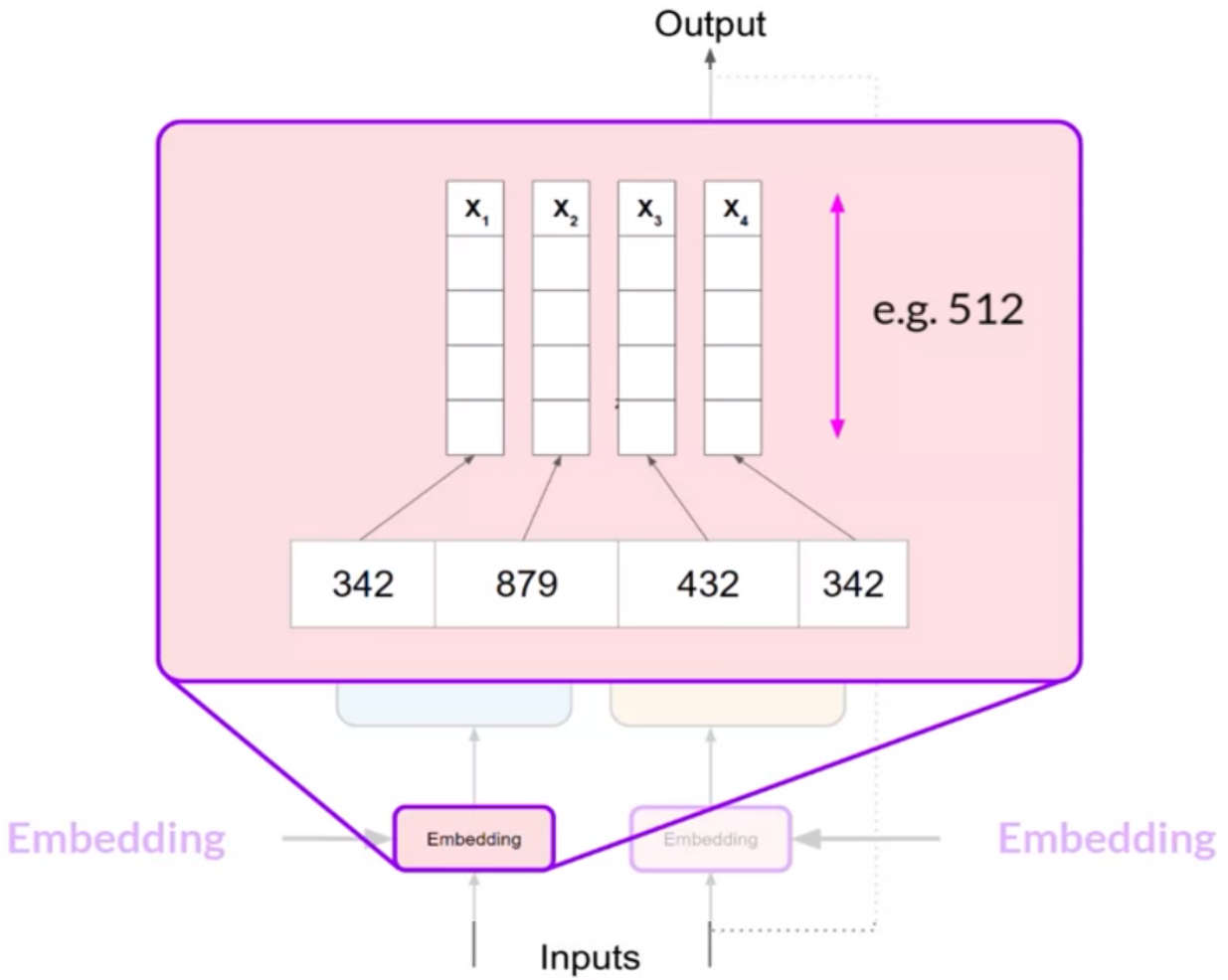
Transformers Architecture



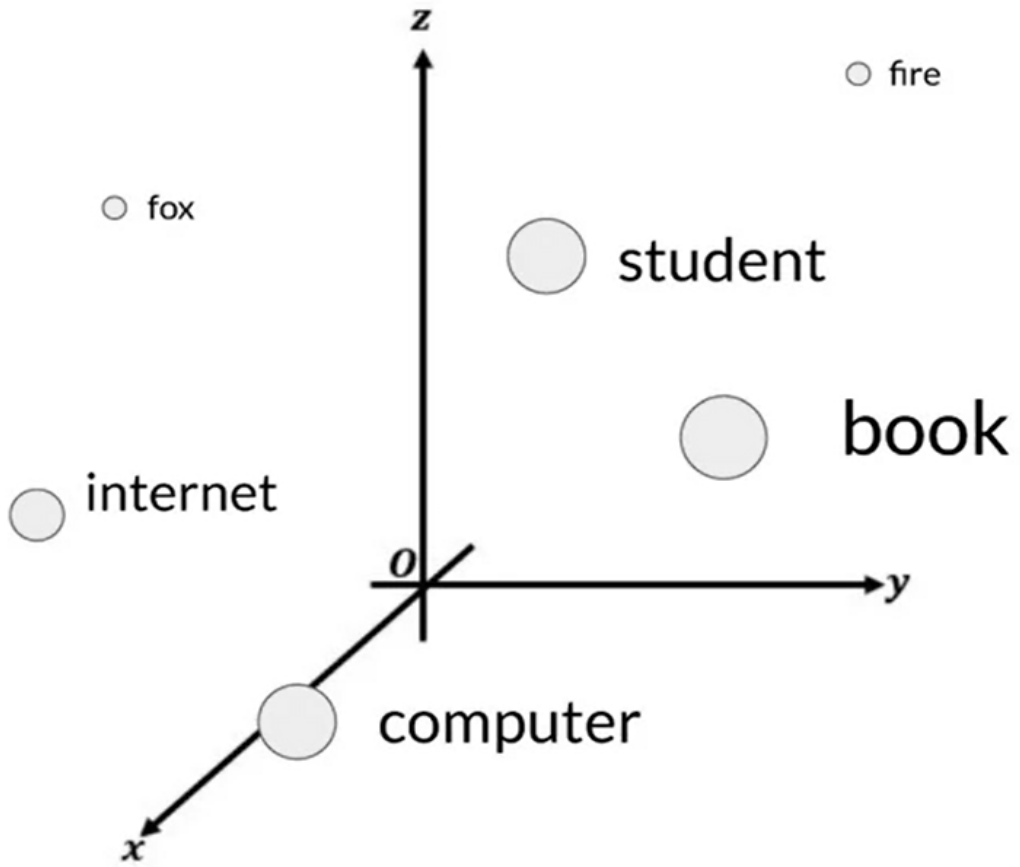
Transformers Architecture



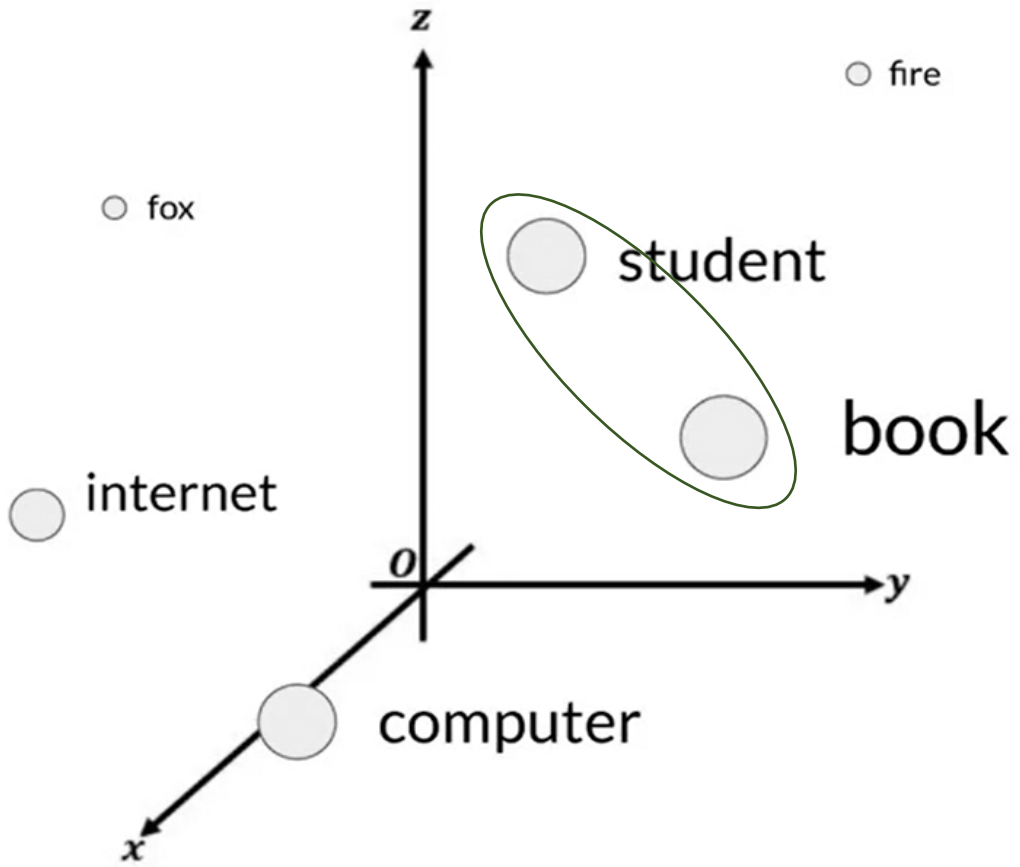
Transformers Architecture



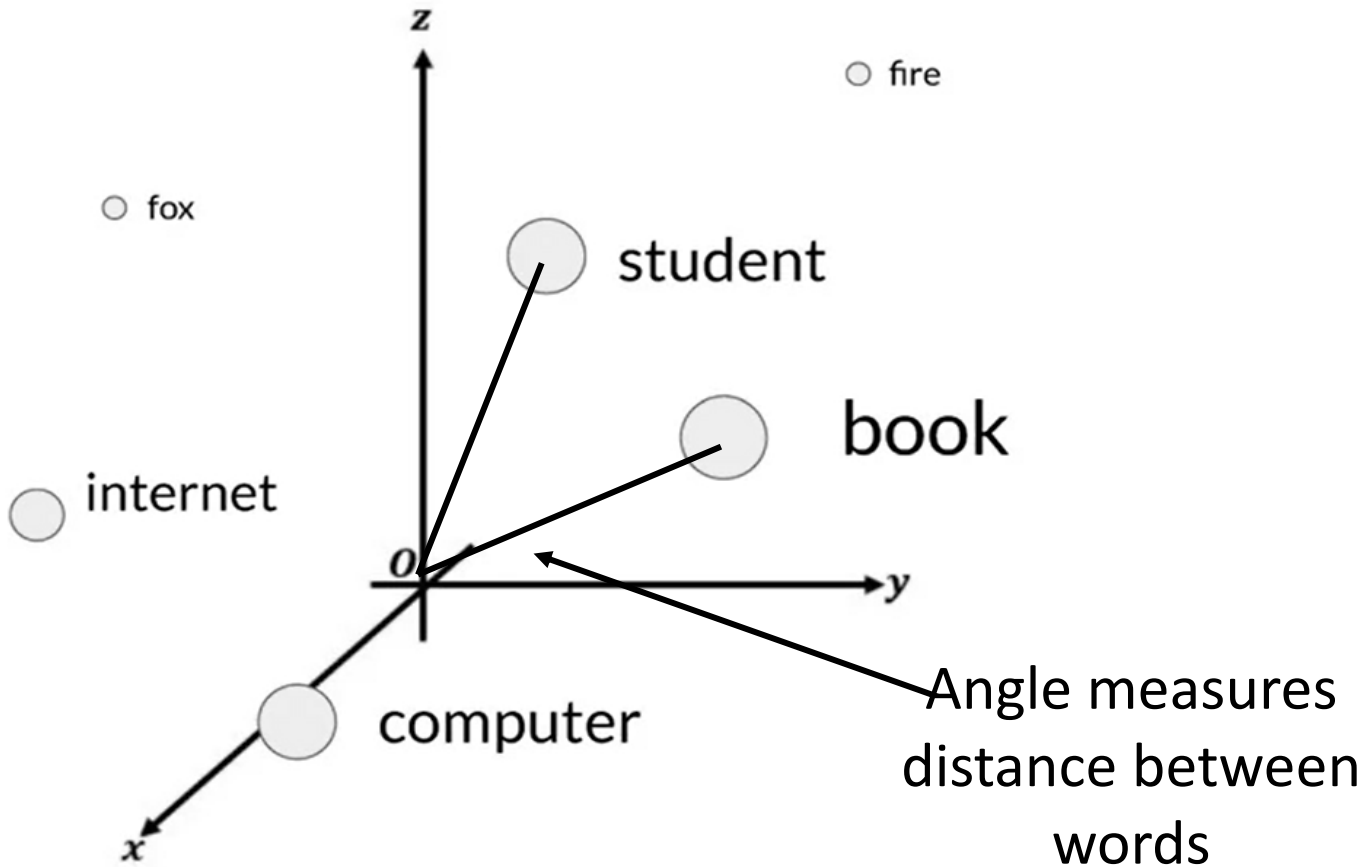
Transformers Architecture



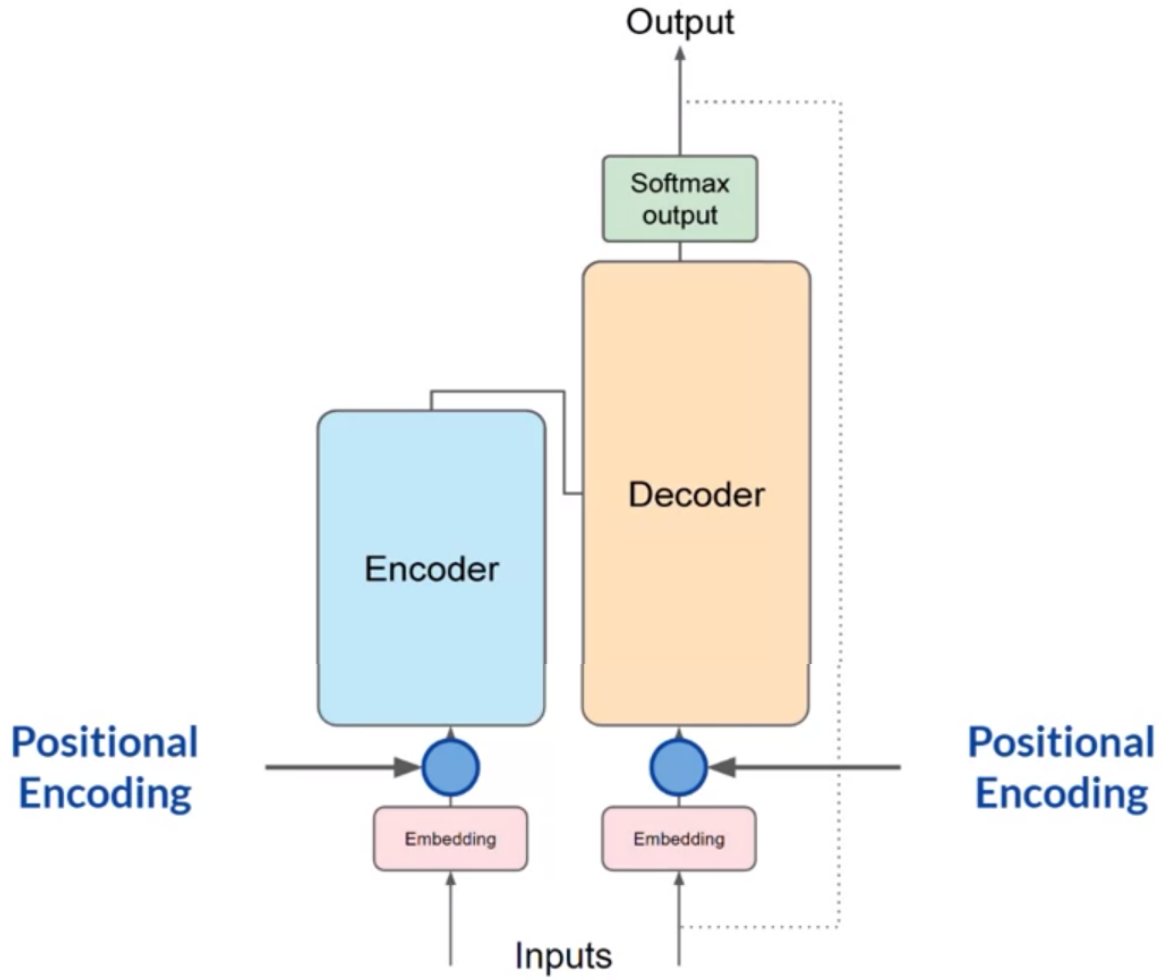
Transformers Architecture



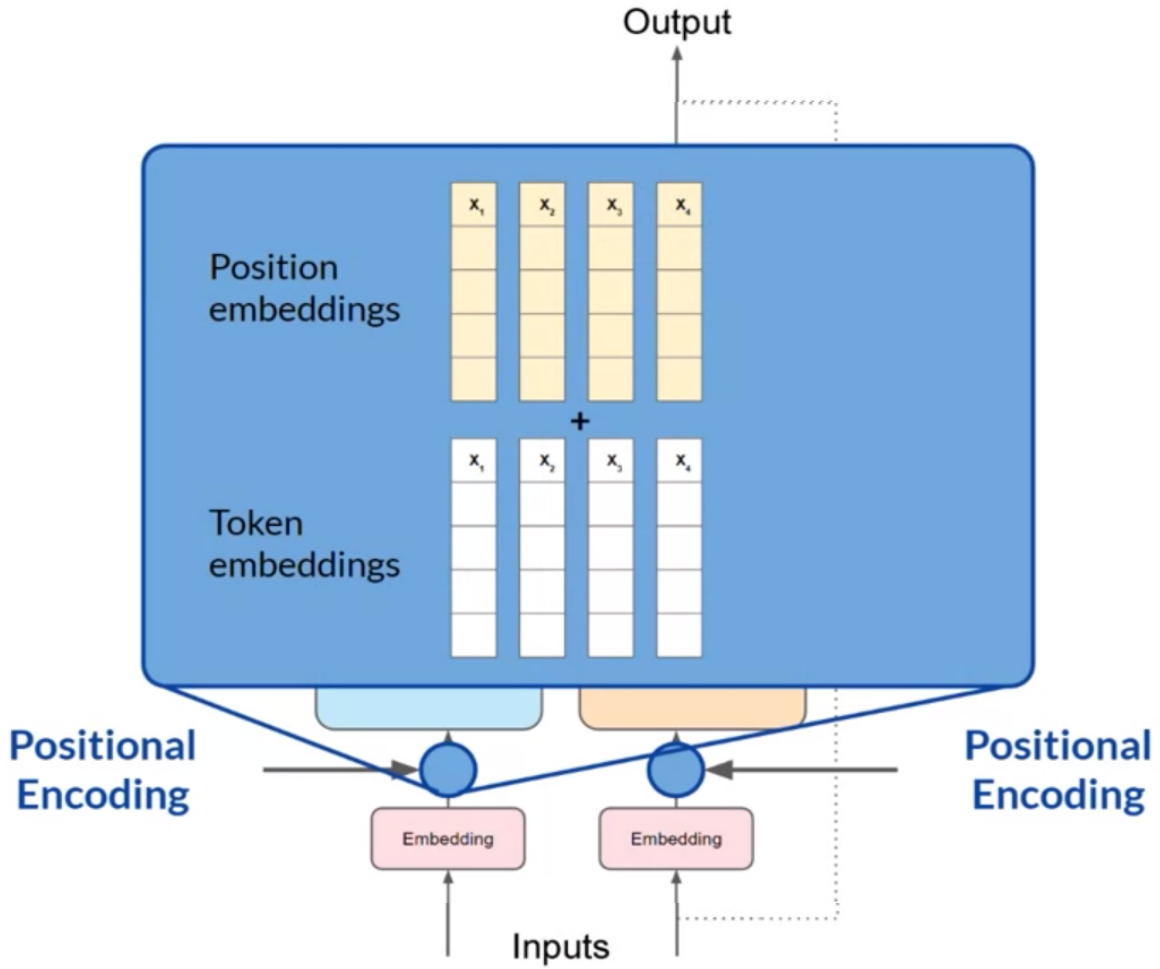
Transformers Architecture



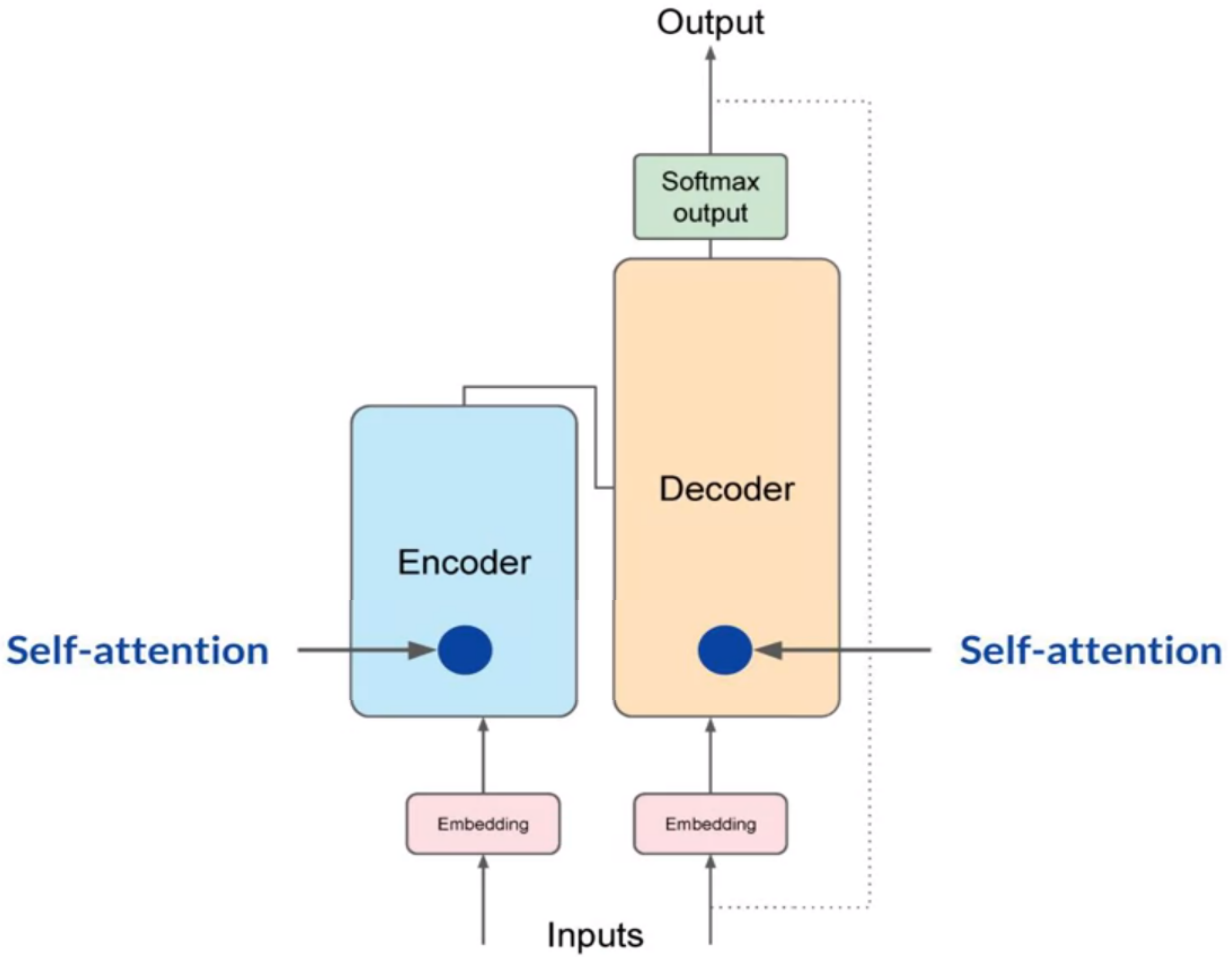
Transformers Architecture



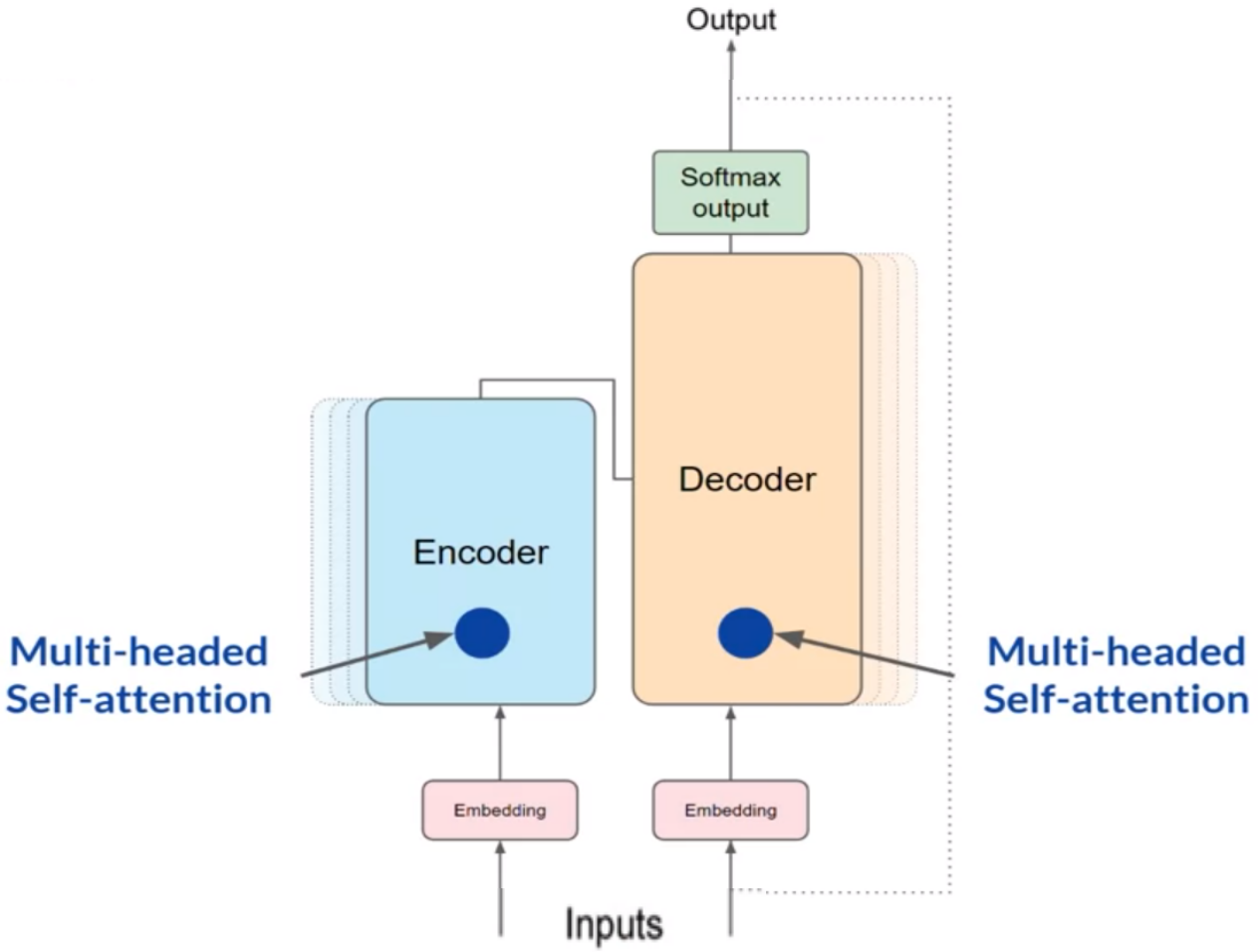
Transformers Architecture



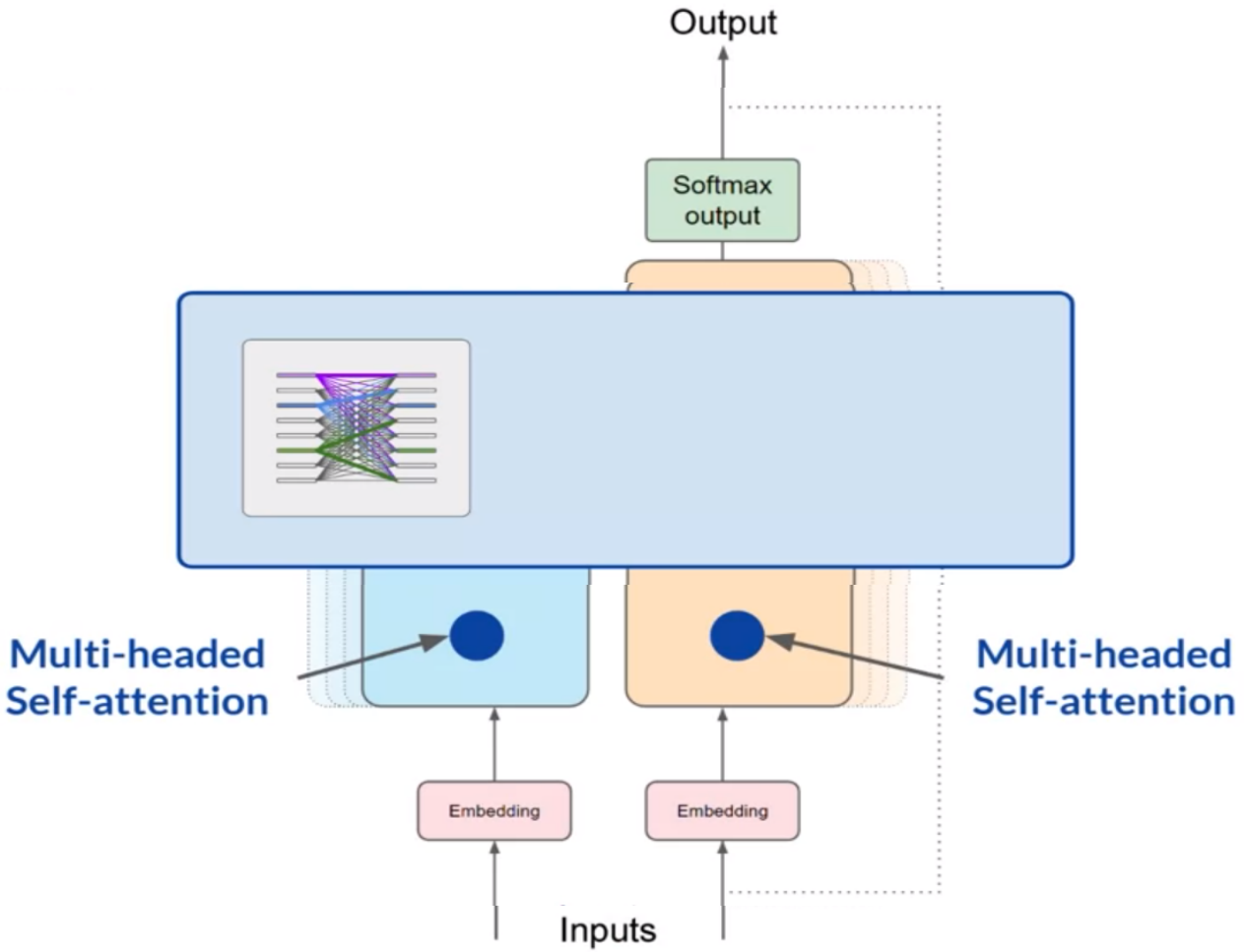
Transformers Architecture



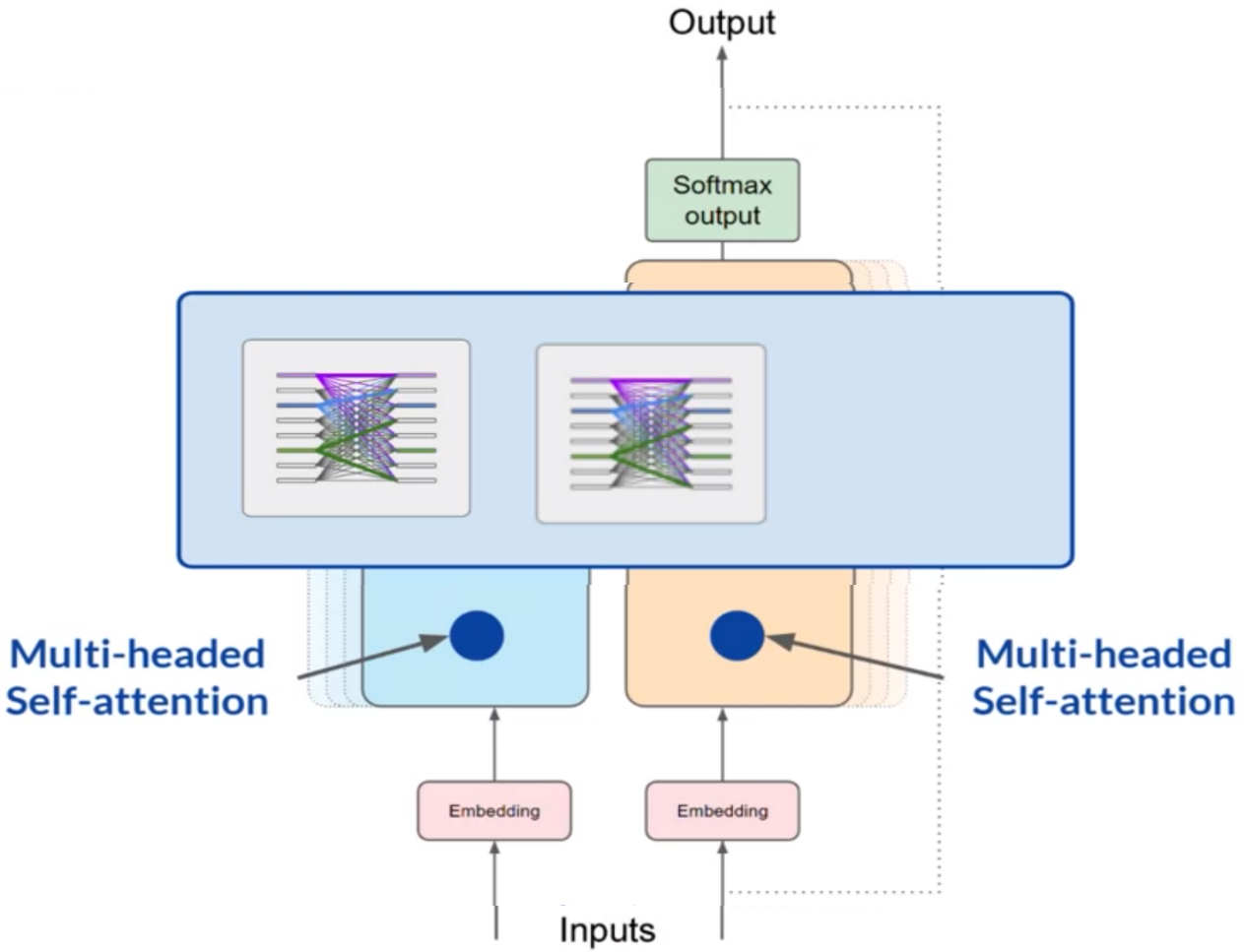
Transformers Architecture



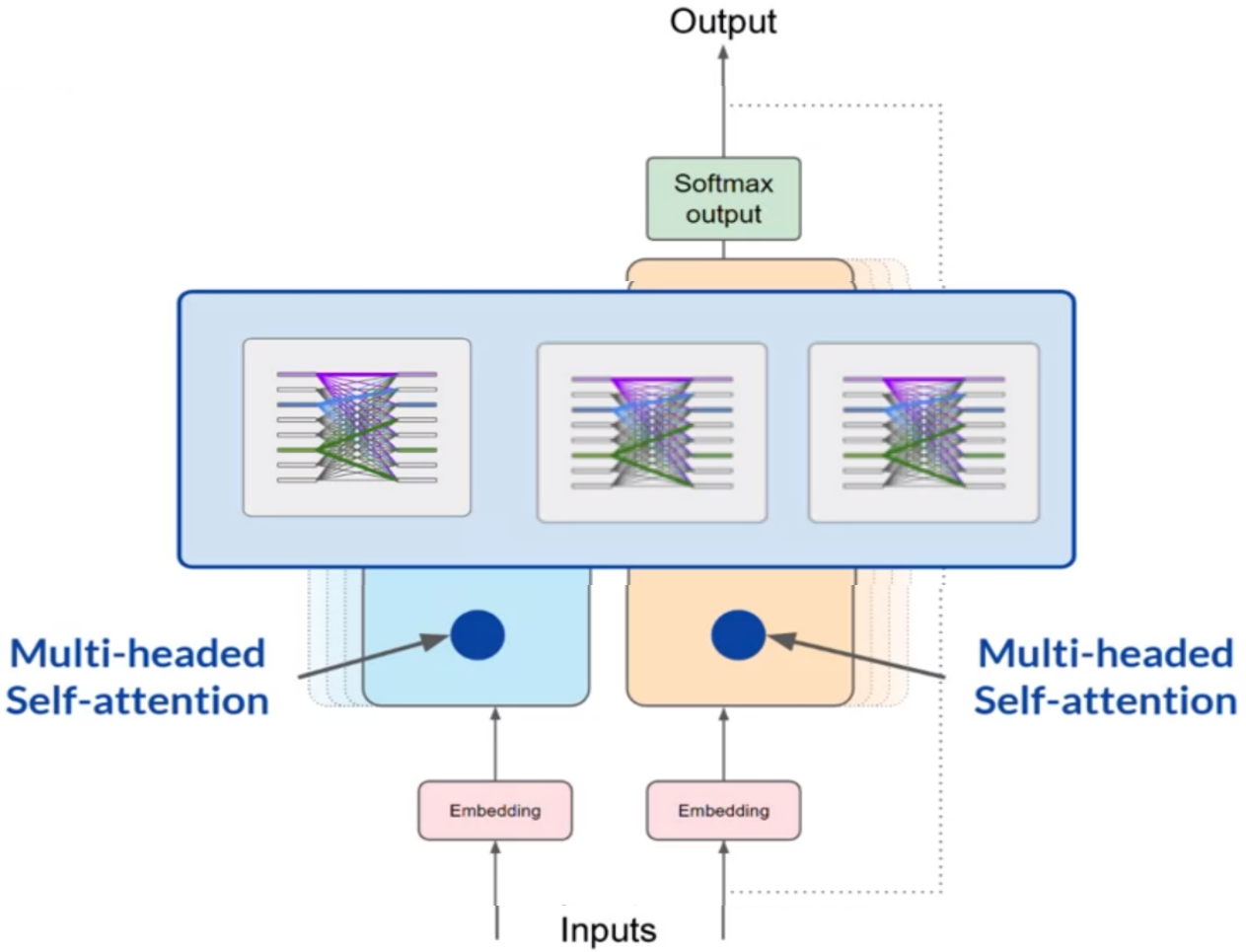
Transformers Architecture



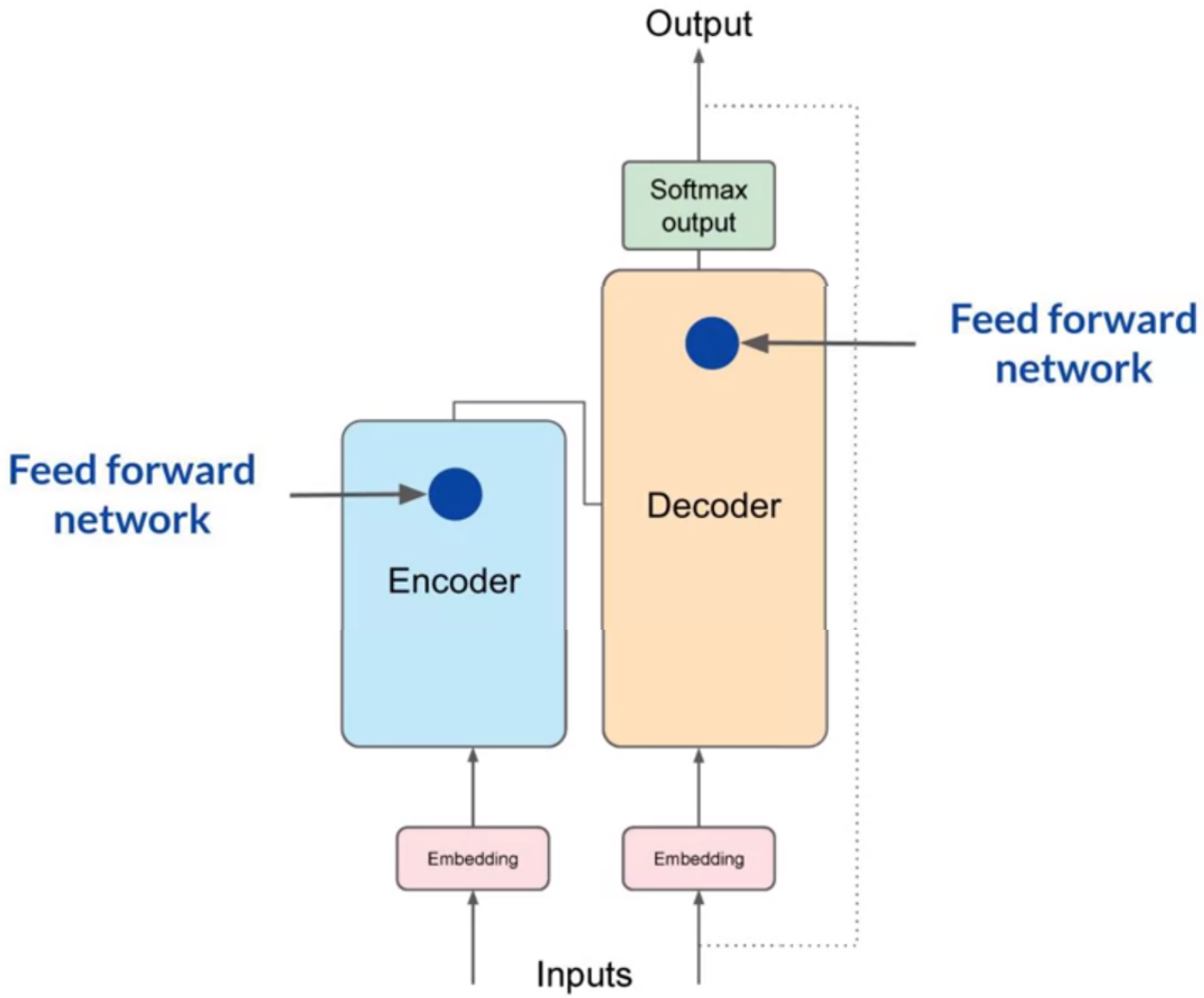
Transformers Architecture



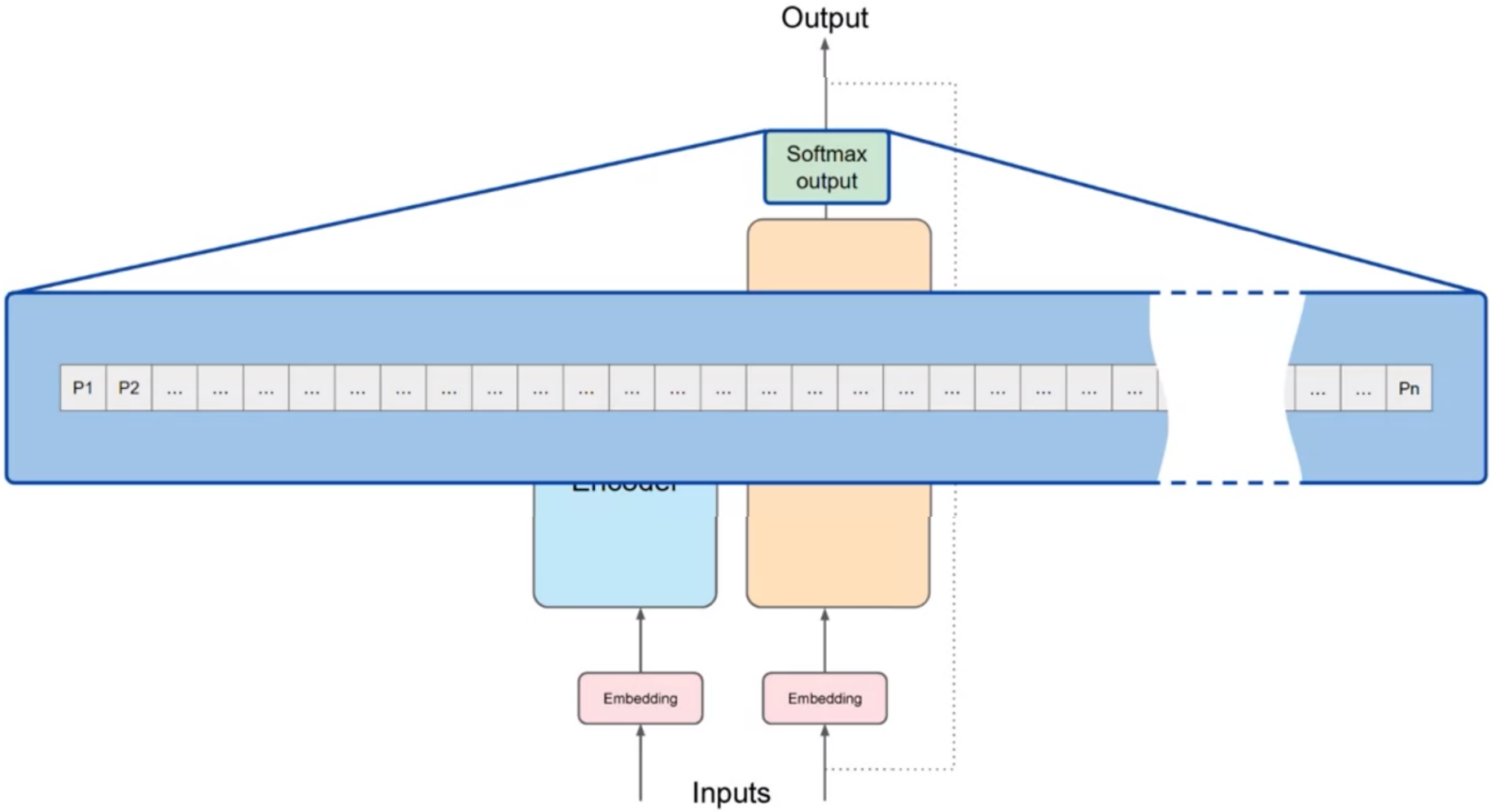
Transformers Architecture



Transformers Architecture



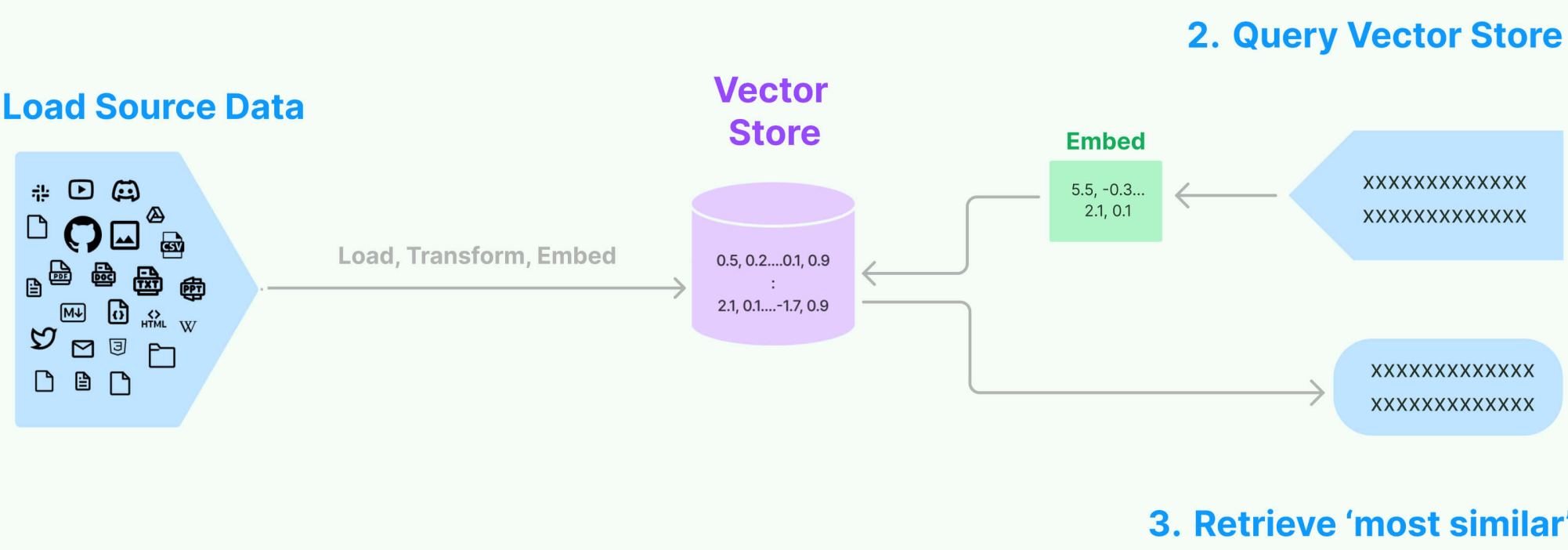
Transformers Architecture



Vector stores

- One of the most common ways to store and search over unstructured data is to embed it and store the resulting embedding vectors, and then at query time to embed the unstructured query and retrieve the embedding vectors that are 'most similar' to the embedded query.
- A vector store takes care of storing embedded data and performing vector search for you.

Vector Stores



Embedding

Context:
Revenue of Apple

Apple

Embedding

Context:
Revenue of Apple

Apple

Parameters

related_to_phones

Is_location

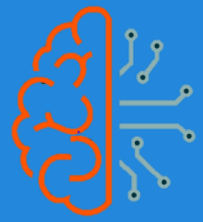
has_stock

revenue

is_fruit

calories

Embedding



Context:
Revenue of Apple

Parameters	Apple
related_to_phones	1
Is_location	0
has_stock	1
revenue	82
is_fruit	0
calories	0

Embedding

Context:
Revenue of Apple

Parameters	Apple
related_to_phones	1
Is_location	0
has_stock	1
revenue	82
is_fruit	0
calories	0

$$\begin{bmatrix} 1 \\ 0 \\ 1 \\ 82 \\ 0 \\ 0 \end{bmatrix}$$

Embedding

Context:
Calories in Apple

Parameters	Apple
related_to_phones	0
Is_location	0
has_stock	0
revenue	0
is_fruit	1
calories	95

$$\begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 1 \\ 95 \end{bmatrix}$$

Embedding

Parameters	Context: *Revenue of Apple* Apple	Context: *Calories in Apple* Apple	Context: *Nutrition in Orange* Orange
related_to_phones	1	0	0
Is_location	0	0	0
has_stock	1	0	0
revenue	82	0	0
is_fruit	0	1	1
calories	0	95	50

Embedding

Parameters
related_to_phones
Is_location
has_stock
revenue
is_fruit
calories

Context:
Revenue of Apple

Apple
1
0
1
82
0
0

Context:
Calories in Apple

Apple
0
0
0
0
1
95

Context:
Nutrition in Orange

Orange
0
0
0
0
1
50

Embedding

Parameters	Context: *Revenue of Apple* Apple	Context: *Calories in Apple* Apple	Context: *Nutrition in Orange* Orange	Context: *Employees in Samsung* Samsung
related_to_phones	1	0	0	1
Is_location	0	0	0	0
has_stock	1	0	0	1
revenue	82	0	0	76
is_fruit	0	1	1	0
calories	0	95	50	0

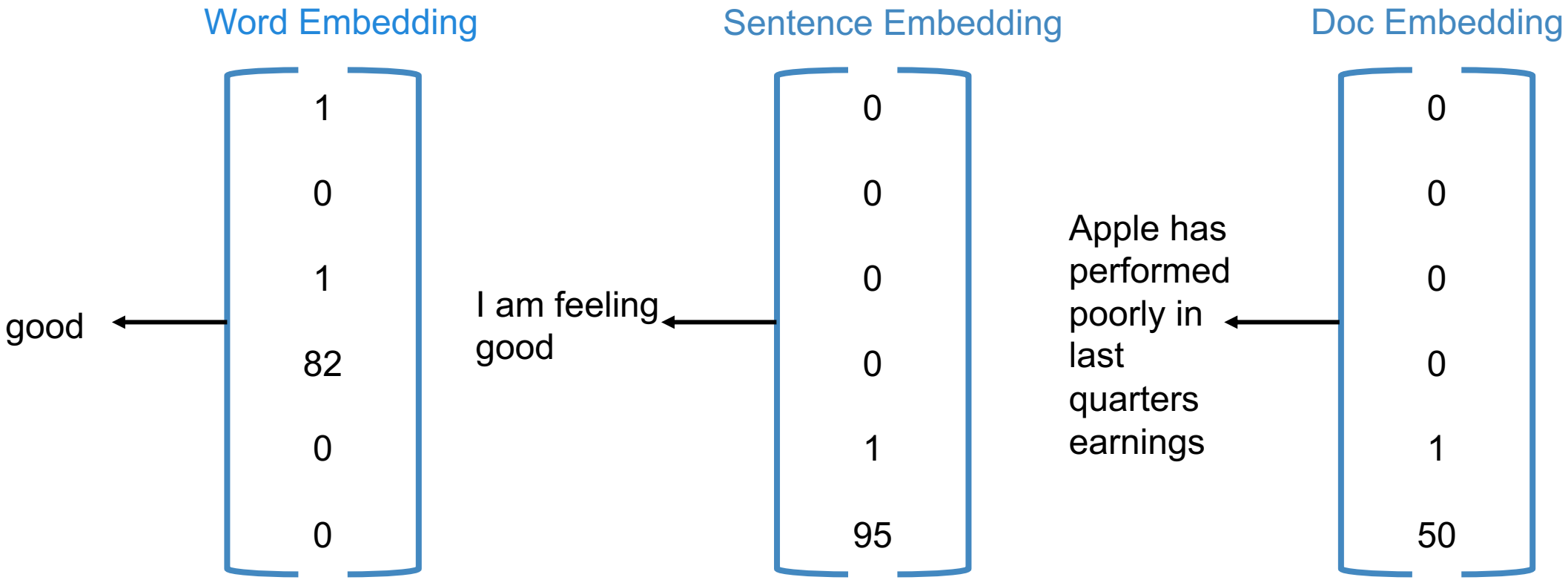
Embedding

Parameters	Context: *Revenue of Apple* — Apple	Context: *Calories in Apple* — Apple	Context: *Nutrition in Orange* — Orange	Context: *Employees in Samsung* — Samsung
related_to_phones	1	0	0	1
Is_location	0	0	0	0
has_stock	1	0	0	1
revenue	82	0	0	76
is_fruit	0	1	1	0
calories	0	95	50	0

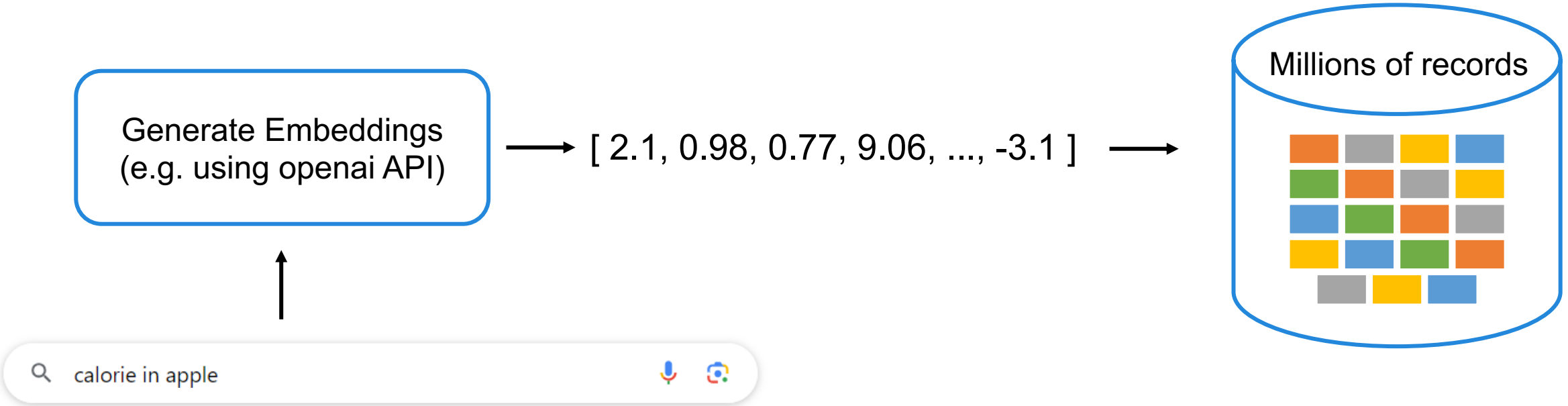
Word embedding techniques

CBOW, Skip gram	Based on transformer architecture	Based on LSTM
Word2vec	BERT	ELMo
GloVe	GPT	
fastText		

Embedding techniques



Traditional Database



Traditional Database

Query Vector

[2.1, 0.98, 0.77, 9.06,...,-3.1]

[2.0, 0.96, 0.67, 9-4, .., -3-7]

[3.45, 6.29, 0.53, 8.47 ..., 4.86]

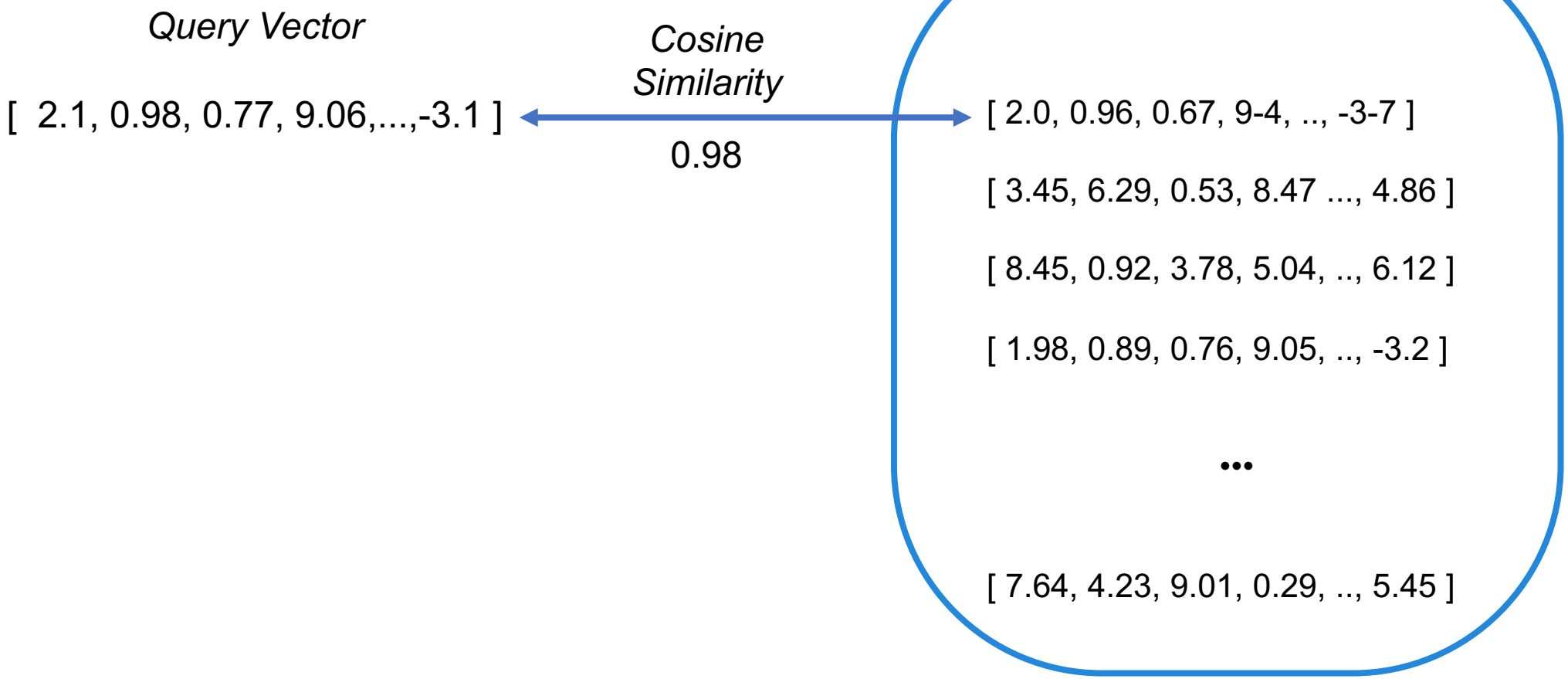
[8.45, 0.92, 3.78, 5.04, .., 6.12]

[1.98, 0.89, 0.76, 9.05, .., -3.2]

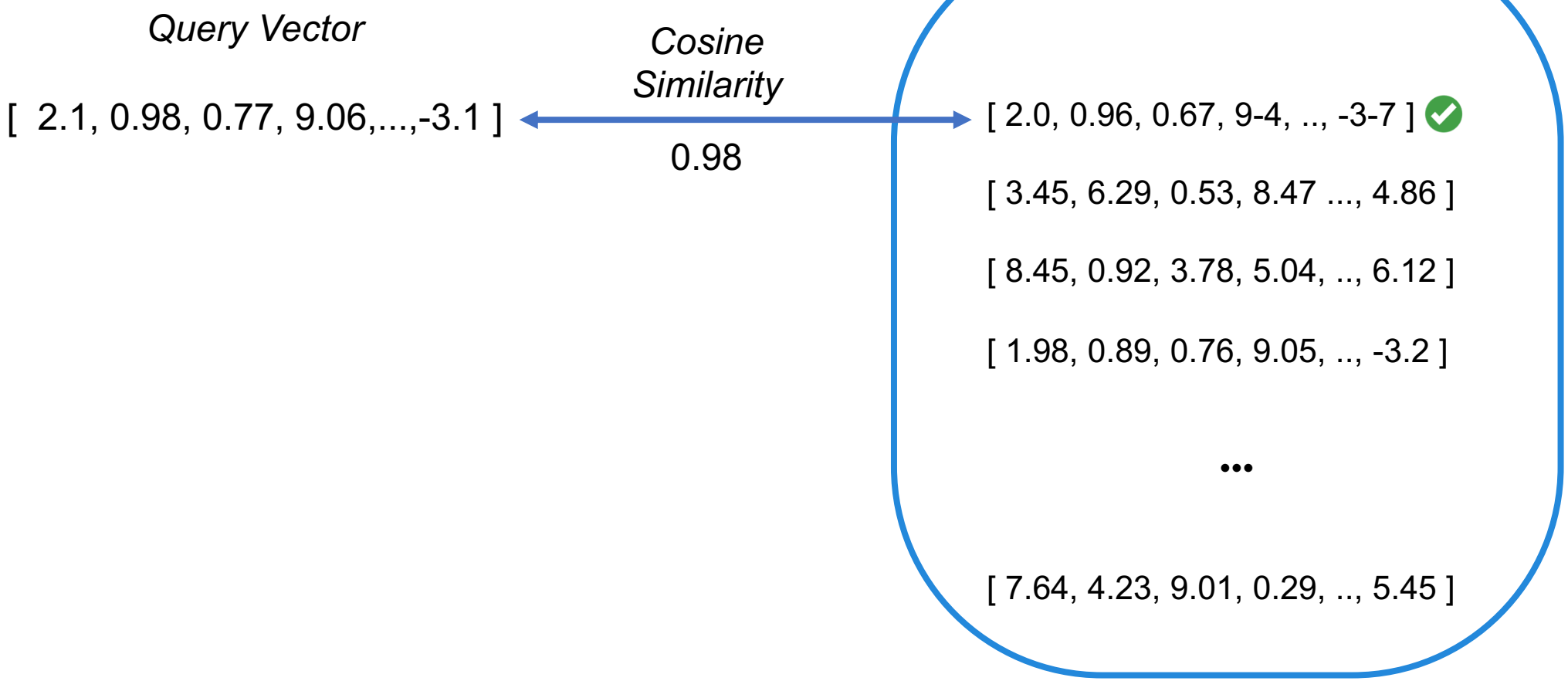
...

[7.64, 4.23, 9.01, 0.29, .., 5.45]

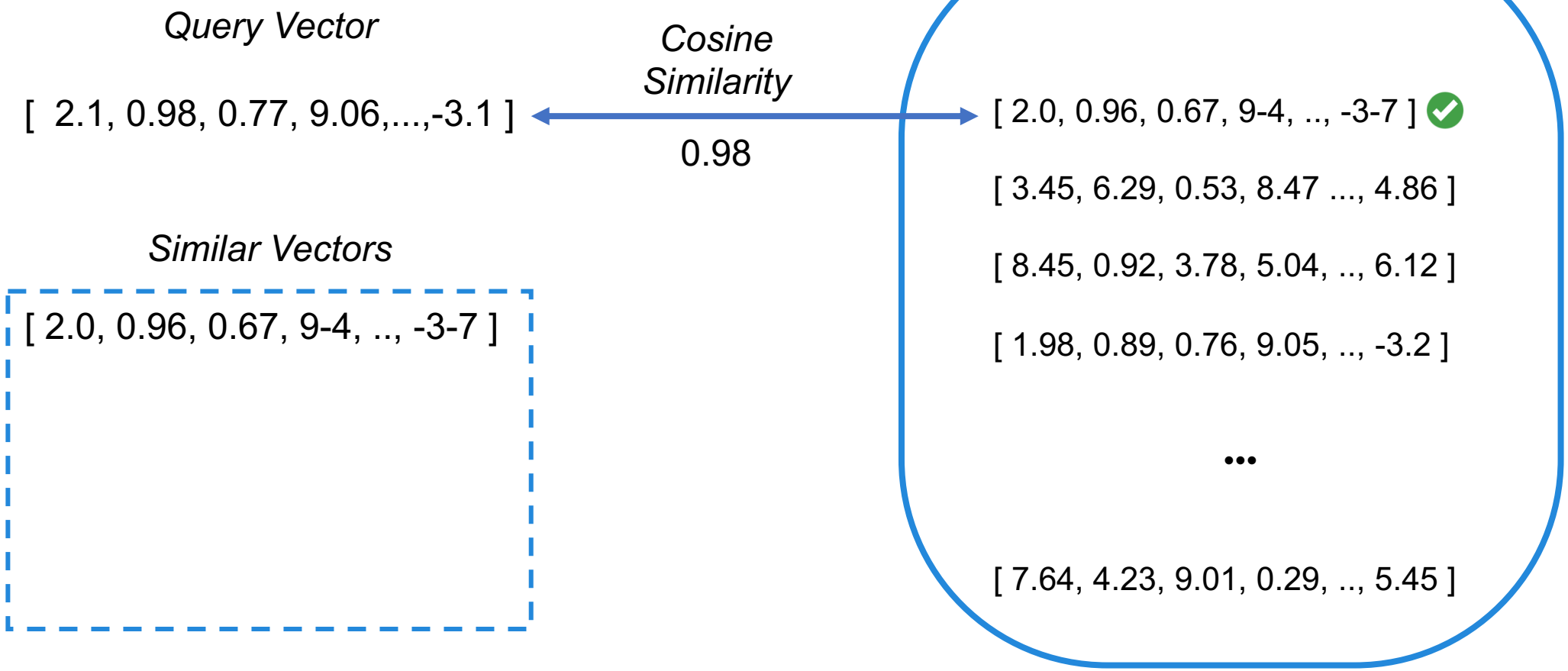
Traditional Database



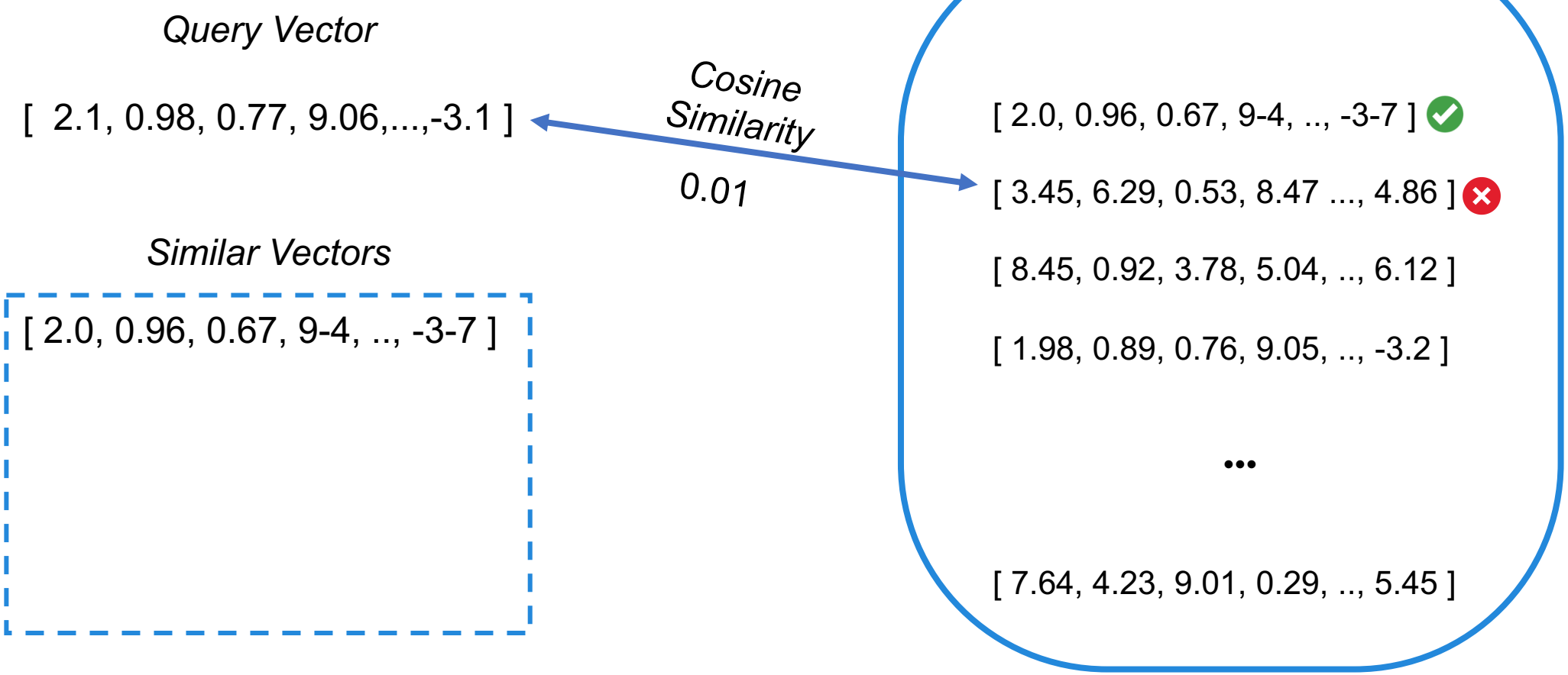
Traditional Database



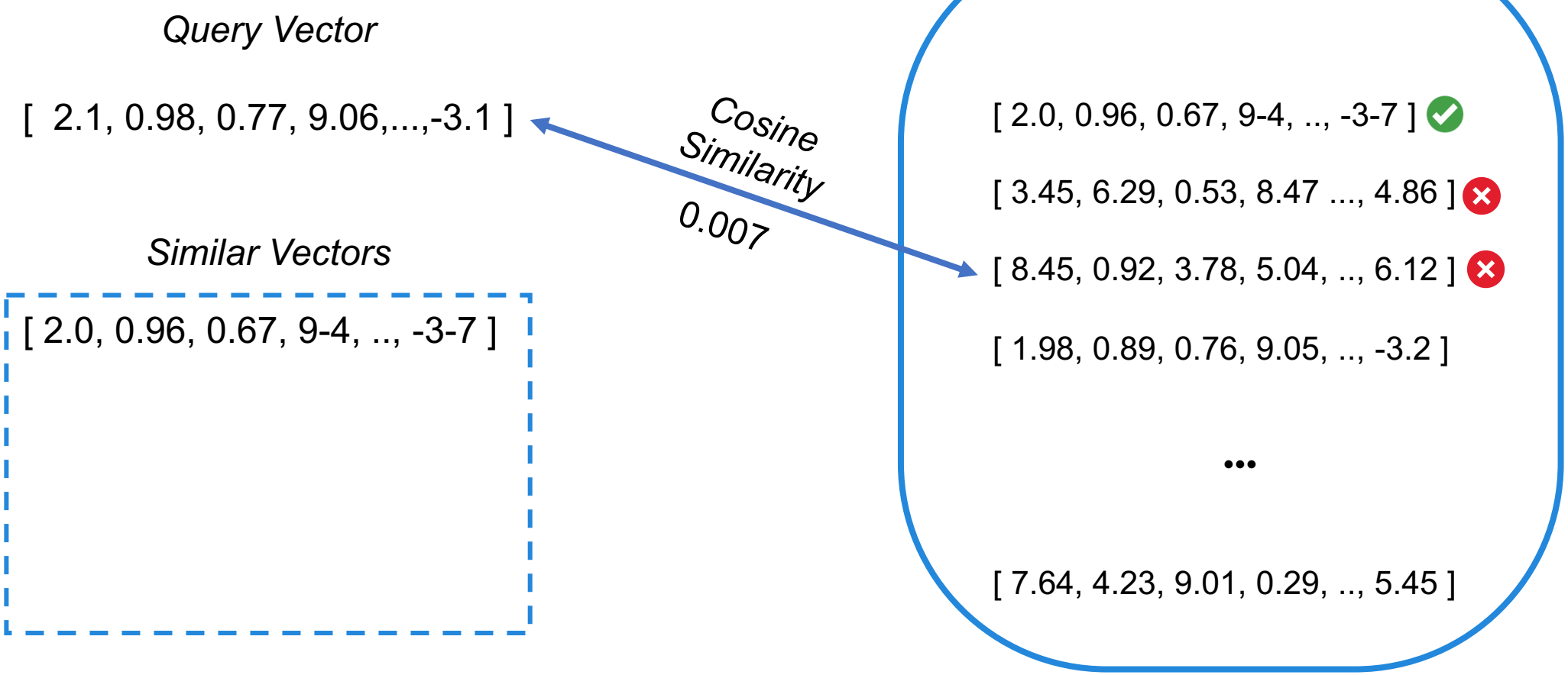
Traditional Database



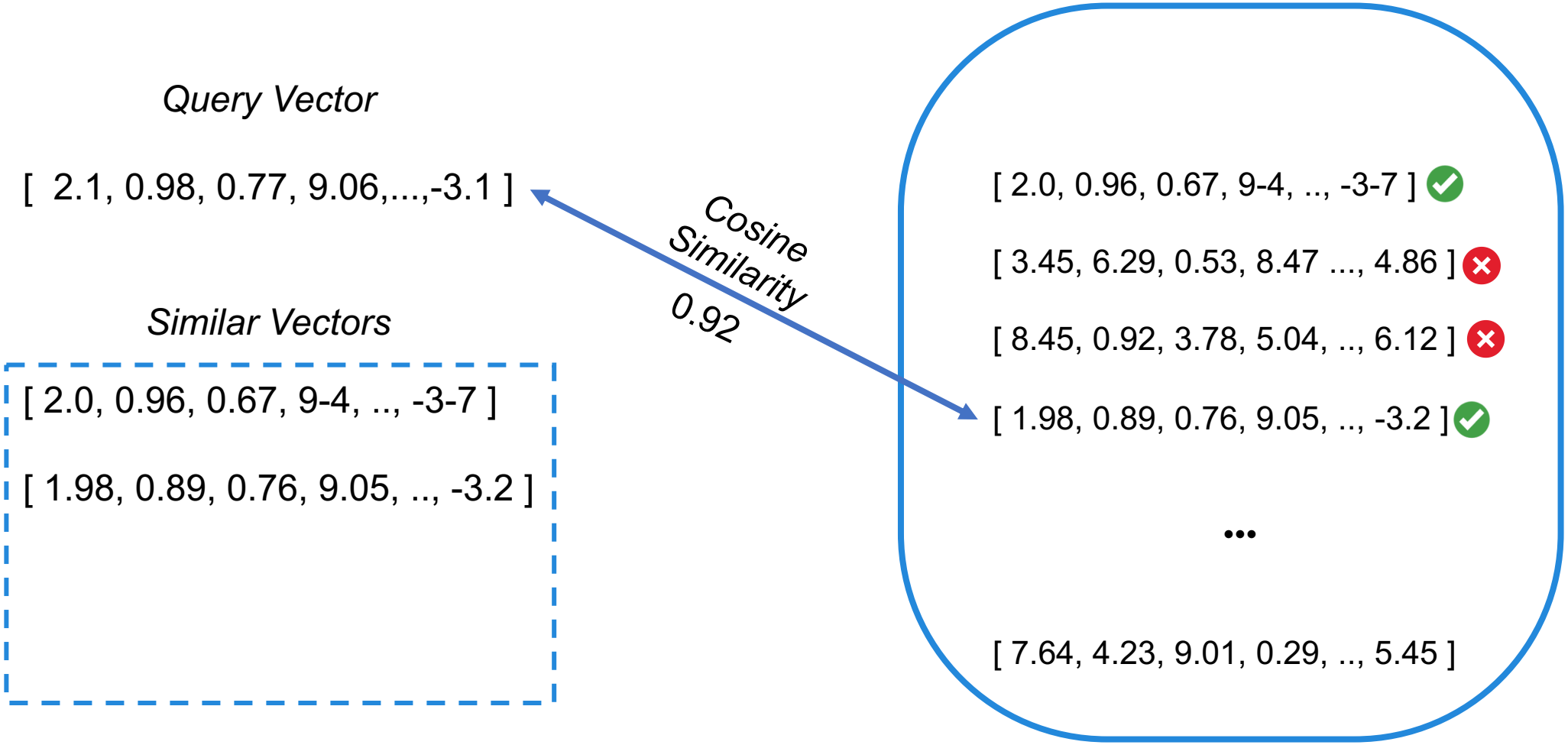
Traditional Database



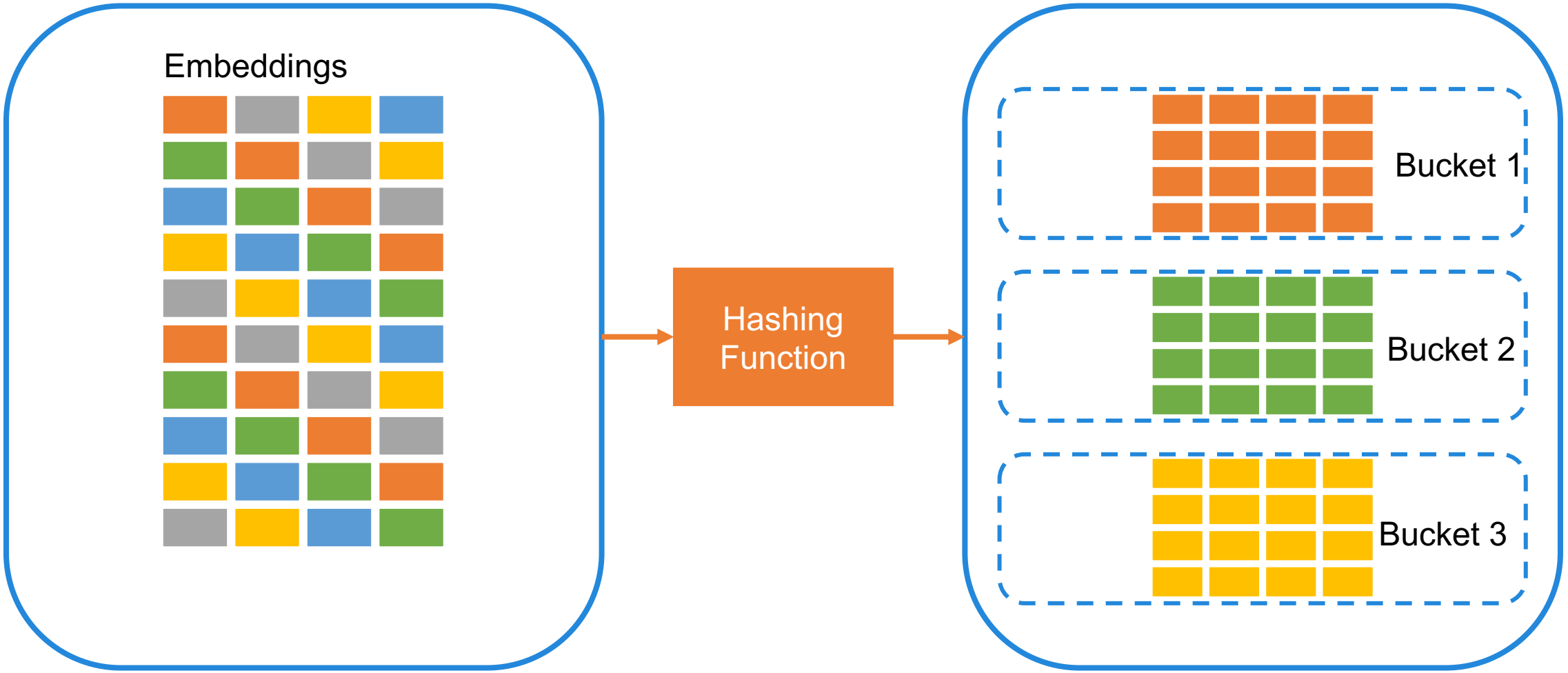
Traditional Database



Traditional Database



Traditional Database



Text AI LLMs

- GPT-3
- GPT-4
- LaMDA
- LLaMA
- Stanford Alpaca
- Google FLAN
- Poe
- Falcon LLM

Thank You