UNIVERSITY OF ALBERTA

NATIONAL BANK

From Data to Decisions

When (and When Not) to Use Generative AI in Business

Tony M. Yousefnezhad

tony@learningbymachine.com

Sr. Data Scientist at *National Bank of Canada*

Adjunct Professor at *University of Alberta*

OSS Contributor, *Learning By Machine*

Personal Website

Learning By Machine

DAMA Edmonton Chapter - April 2026

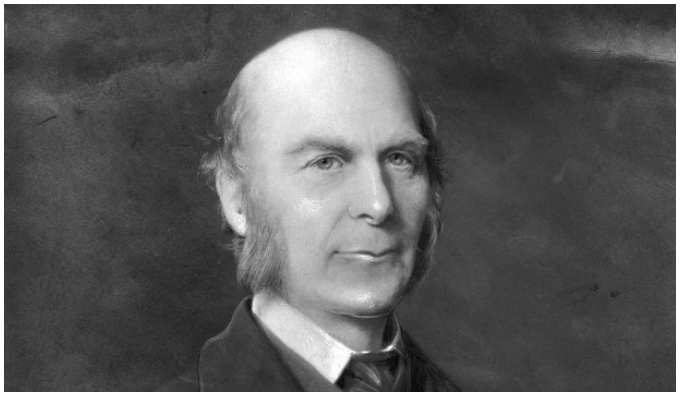
Outline

- The Regression Story
- Attention Is All You Need
- The Era After Large Language Models
  - Retrieval Augmented Generation (RAG)
  - Model Context Protocol (MCP)
- Conclusion

The Regression Story

Linear Regression

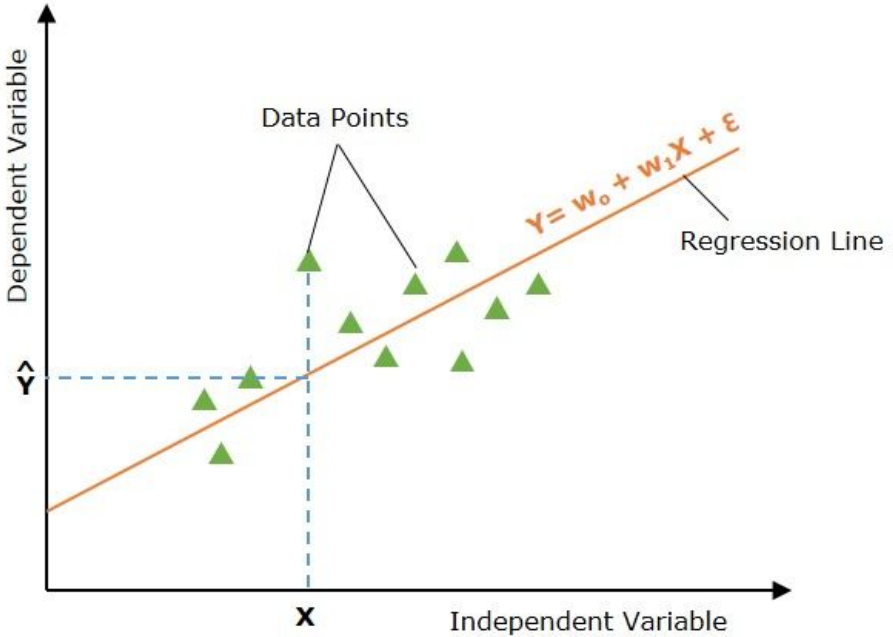
- It was first introduced by Francis Galton around **1886**

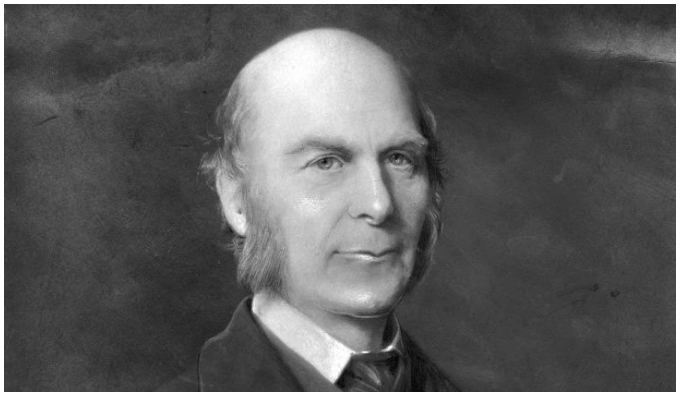


Sir Francis Galton (1860–1911)

Linear Regression

- It was first introduced by Francis Galton around **1886**
- Our objective is to learn a regression model from training data that can accurately predict outcomes for new samples (i.e., testing data) that were **not available** during training.

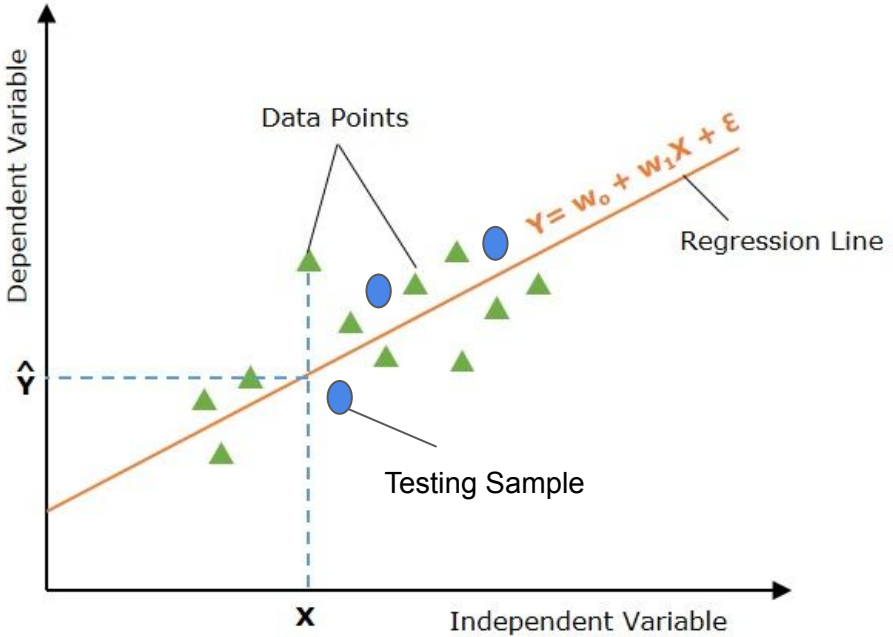


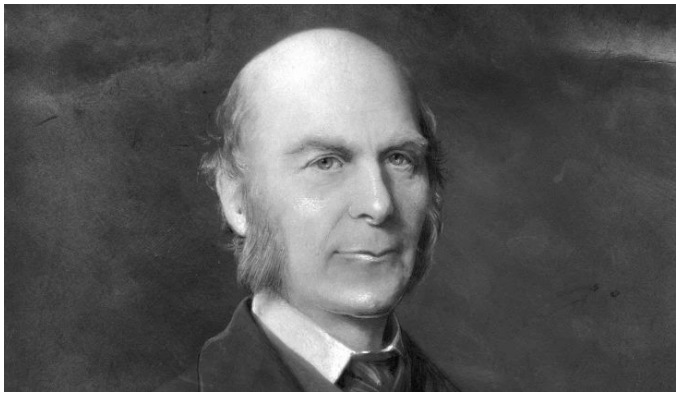


Sir Francis Galton (1860–1911)

Linear Regression

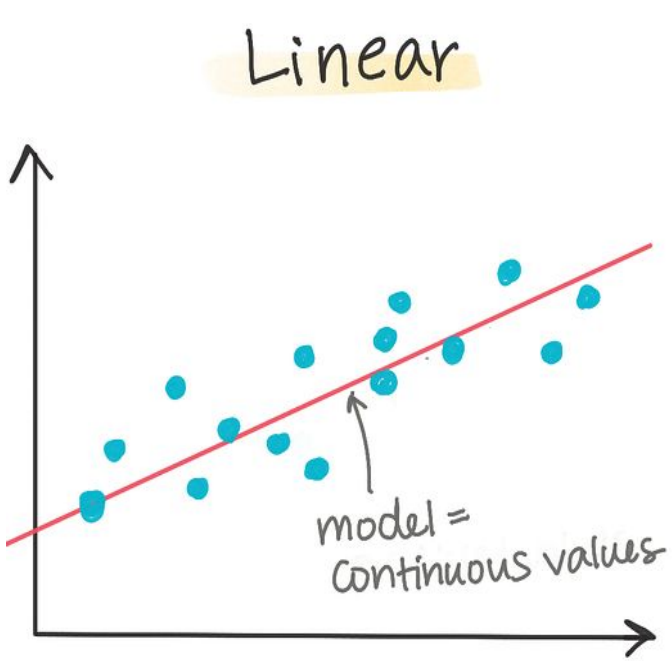
- It was first introduced by Francis Galton around **1886**
- Our objective is to learn a regression model from training data that can accurately predict outcomes for new samples (i.e., testing data) that were **not available** during training.

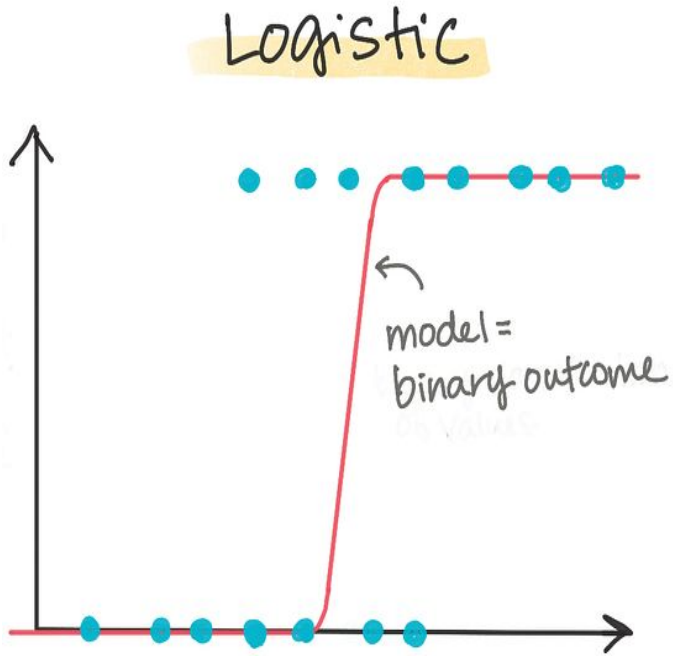




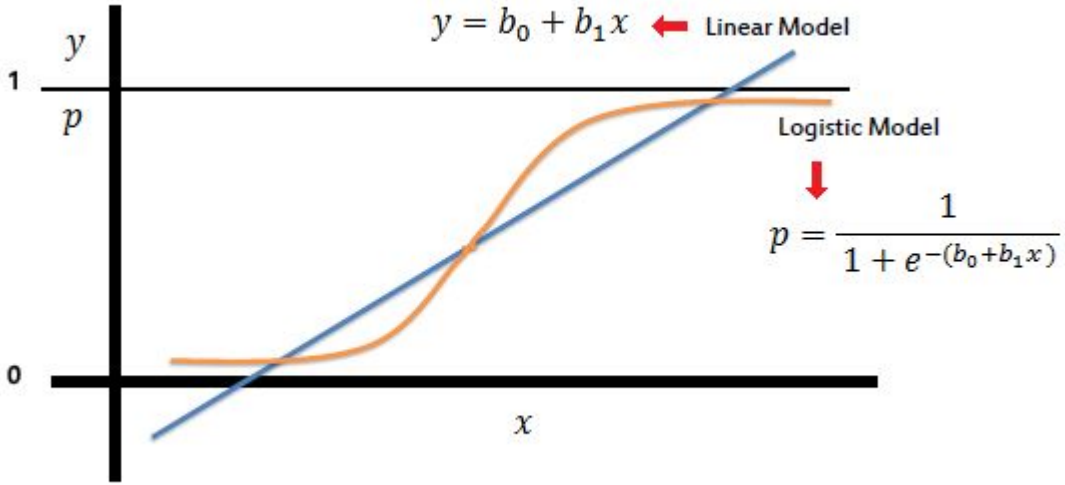
Sir Francis Galton (1860–1911)

Linear Regression vs. Logistic Regression





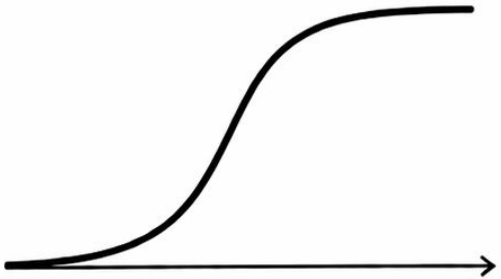
Linear Regression vs. Logistic Regression



Logistic Regression: Binary vs. Multi-class

SIGMOID FUNCTION

Used in binary classification
(Yes/No, Spam/Not Spam)



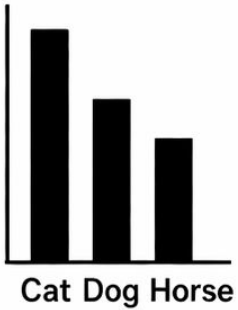
$$\sigma(x) = \frac{1}{1+e^{-x}}$$

Helps your model decide
between two outcomes

SOFTMAX FUNCTION

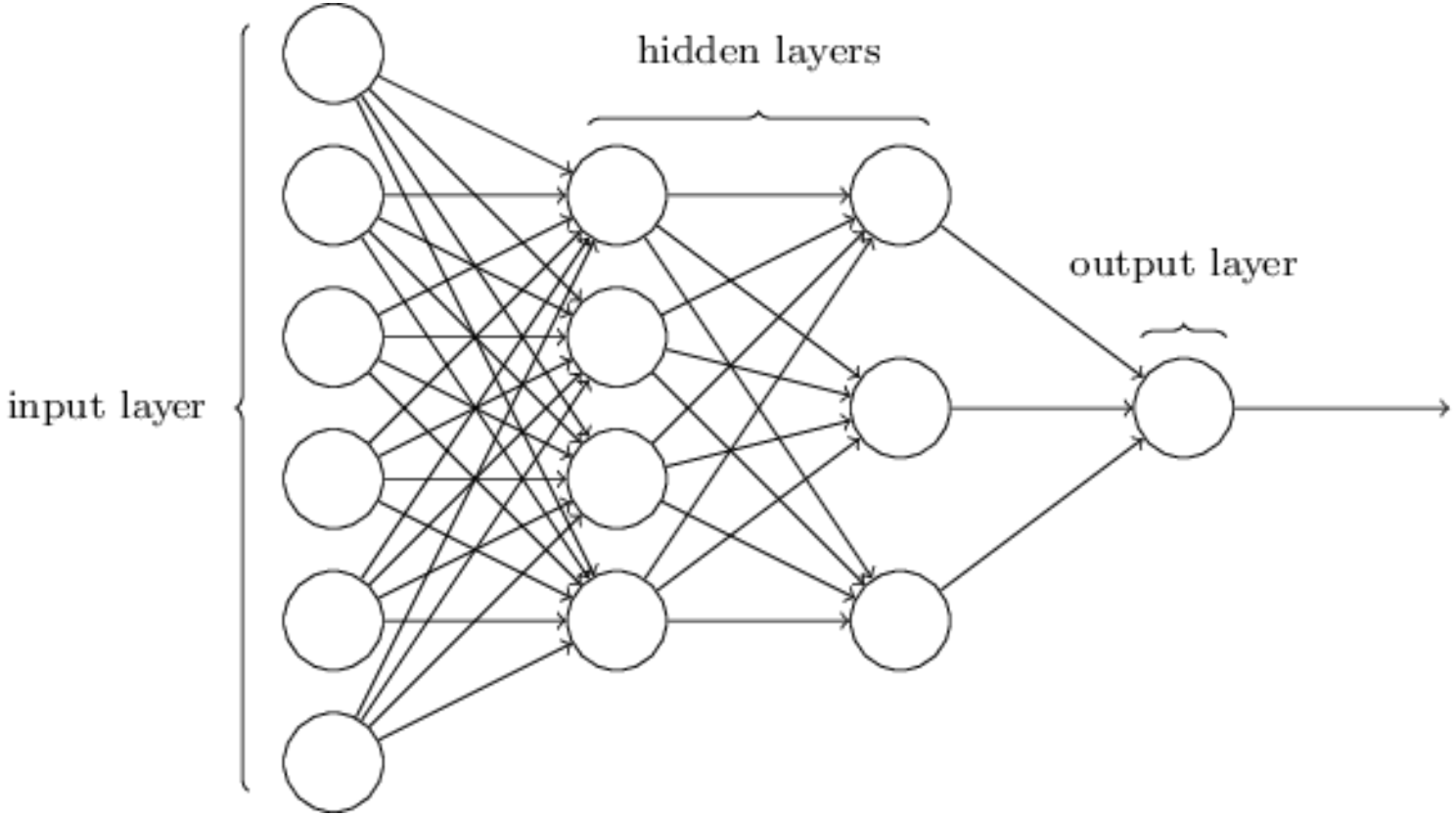
Used in multi-class classification
(Cat/Dog/Horse)

$$Softmax(x_i) = \frac{e^{x_i}}{\sum_j e^{x_j}}$$

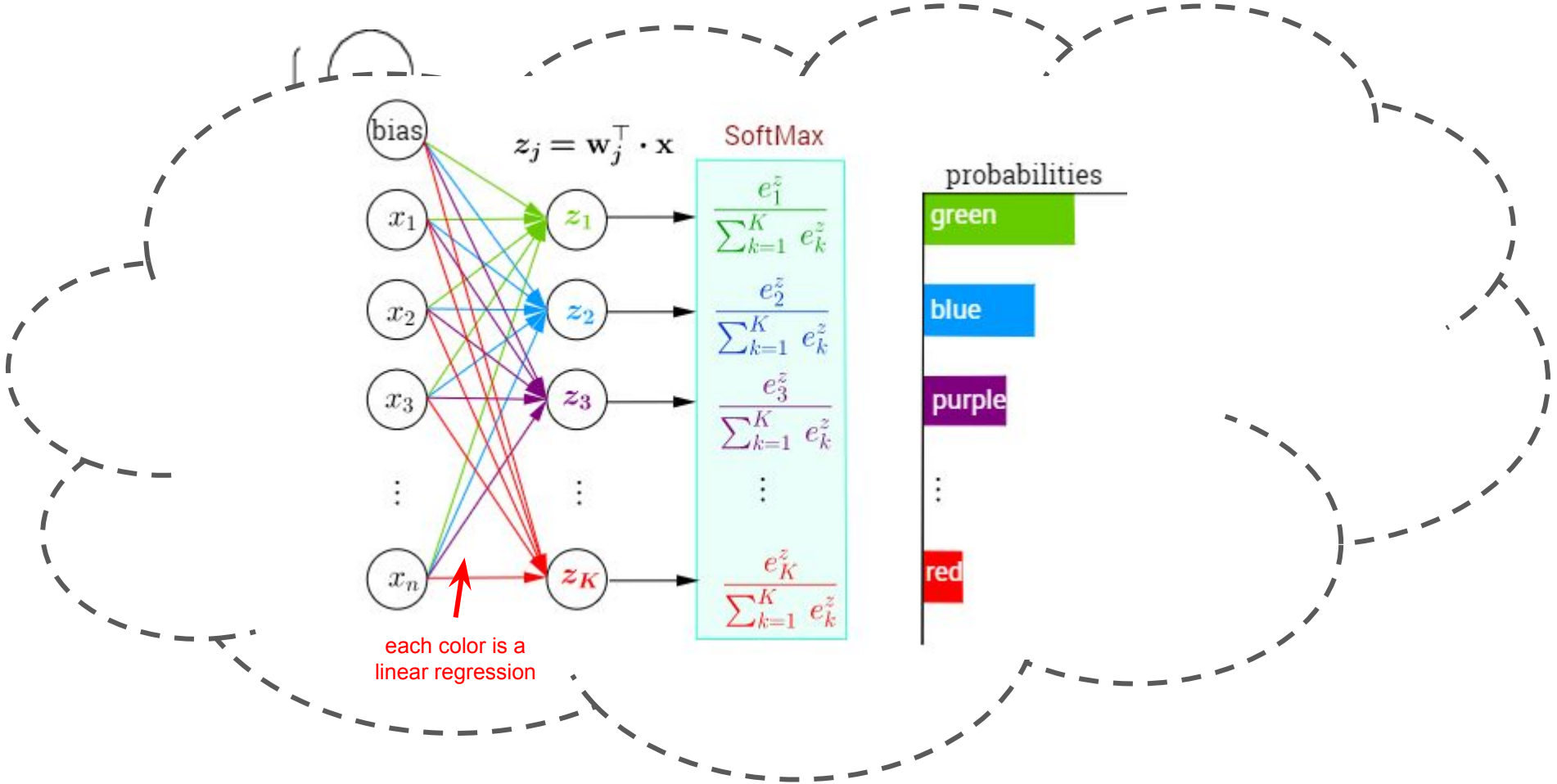


Helps the model pick
the most likely class

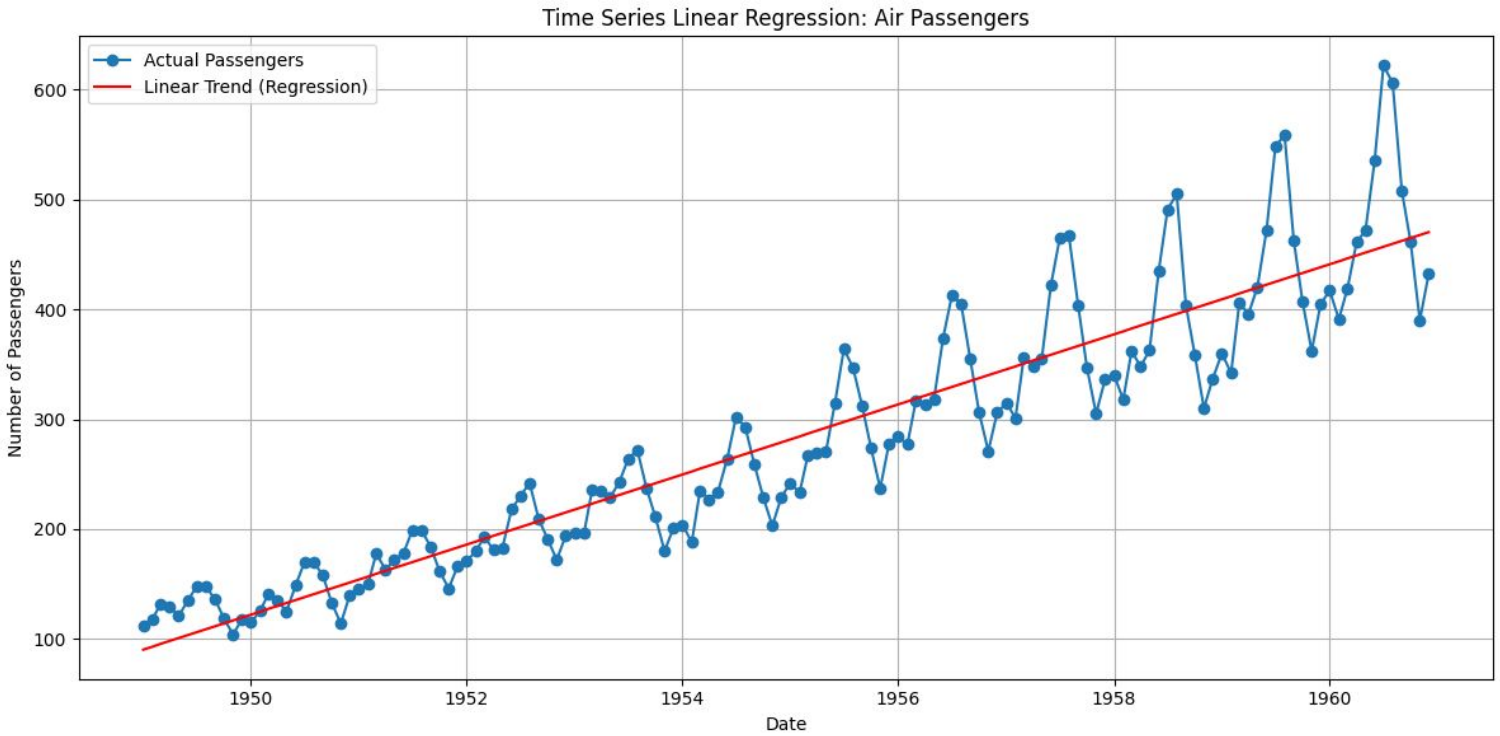
Multilayer Perceptron (MLP): Feedforward Neural Network



Multilayer Perceptron (MLP): Feedforward Neural Network

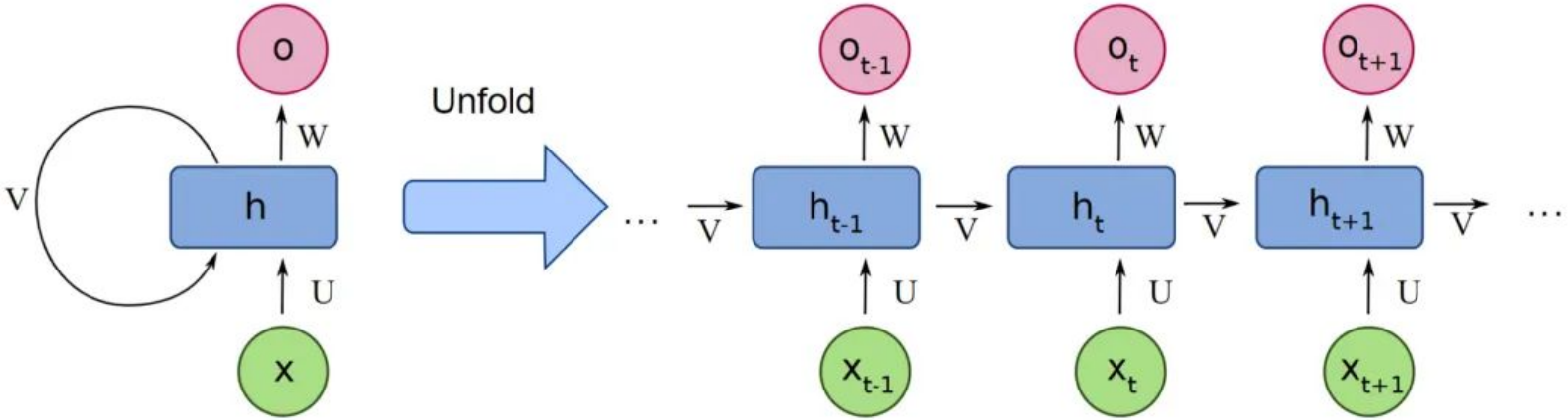


Autoregressive Regression: Sequence Analysis



- Prediction depends on the previous *n* samples—commonly referred to as an AR(*n*) (autoregressive) model.
- Objective $X_t = \varphi X_{t-1} + \varepsilon_t$

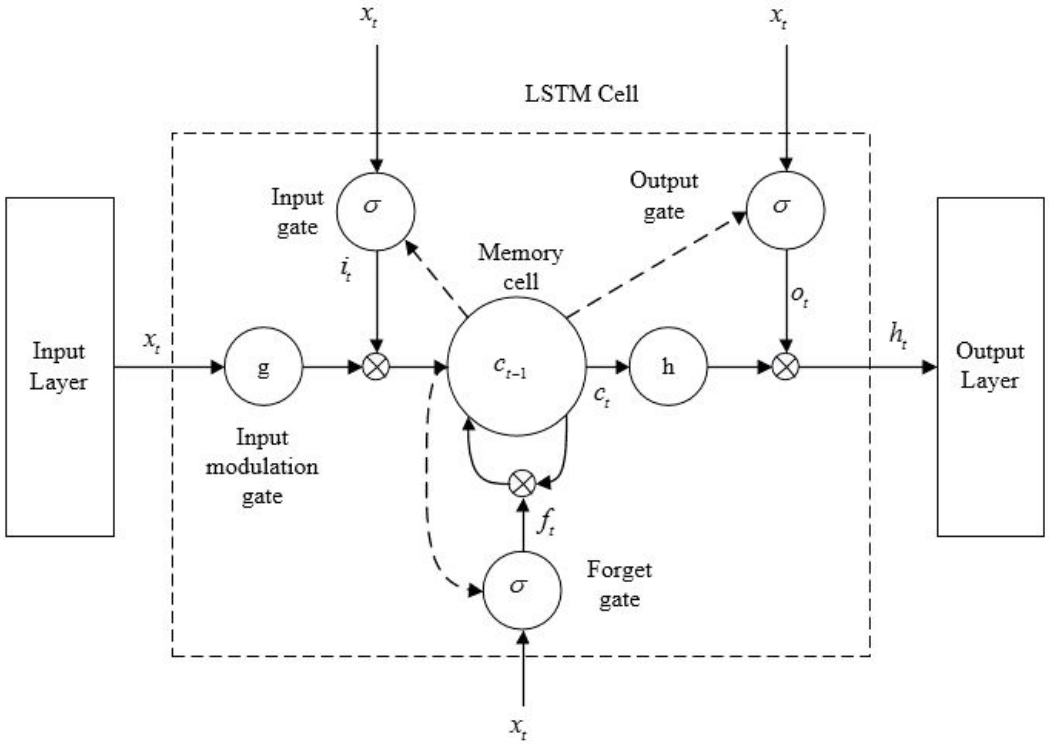
Recurrent Neural Networks (RNNs)



- Designed for **sequential data** (time series, **text**, speech)
- Capture **temporal dependencies** by using previous information
- Maintain a hidden state (**memory**) that updates at each time step
- Same network parameters are shared across all time steps
- Output depends on both **current input and past context**
- Basic RNNs suffer from:
  - **Vanishing** gradients (hard to learn long-term dependencies)
  - **Exploding** gradients (instability during training)

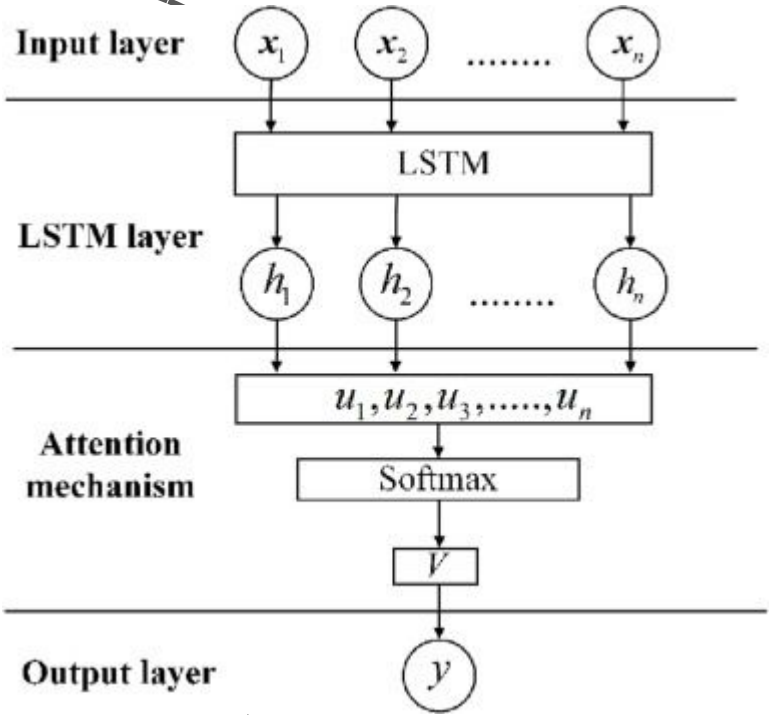
Long Short-Term Memory (LSTM)

- A type of **RNN** designed for sequential data
- Addresses the **vanishing gradient problem** in standard RNNs
- Introduces a dedicated **memory cell** to store long-term information
- Uses **gates** to control information flow:
  - **Forget gate** → decides what to remove from memory
  - **Input gate** → decides what new information to store
  - **Output gate** → controls what to output
- Enables learning of **long-term dependencies** (e.g., earlier words in a sentence)
- Still has limitations:
  - Memory is **compressed into a fixed-size vector**
  - Struggles with **very long sequences** despite improvements



Long Short-Term Memory (LSTM)

- A type of **RNN** designed for sequential dat
- Addresses th
  RNNs
- Introd
  infor
-
-
-
  improve



Attention

Attention Is All You Need

Claude Shannon vs. Noam Chomsky

Chomsky (1969):

"But it must be recognized that the notion of 'probability of a sentence' is an entirely **useless** one, under any known interpretation of this term"

Shannon (1951):

Probability gives us a tool for understanding what is likely to have been said or meant … it is useful for psychological modeling as well as engineering

Linguistics Wars (https://en.wikipedia.org/wiki/Linguistics_wars)

Claude Shannon vs. Noam Chomsky

Chomsky (1969):

"But it must be recognized that the notion of 'probability of a sentence' is an entirely **useless** one, under any known interpretation of this term"

Shannon (1951):

Probability gives us a tool for understanding what is likely to have been said or meant … it is useful for psychological modeling as well as engineering

Linguistics Wars (https://en.wikipedia.org/wiki/Linguistics_wars)

Attention Is All You Need (NIPS 2017)

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

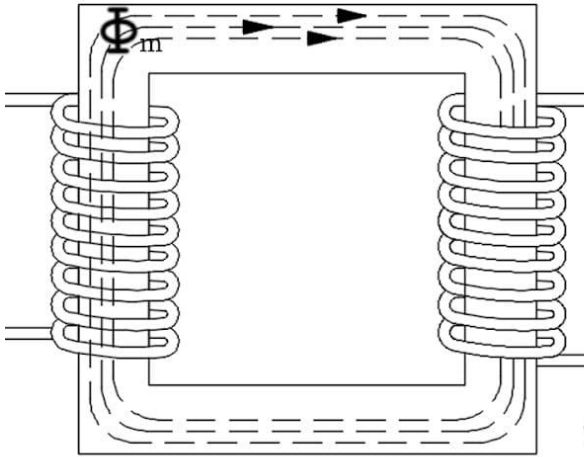
Llion Jones* Google Research llion@google.com

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

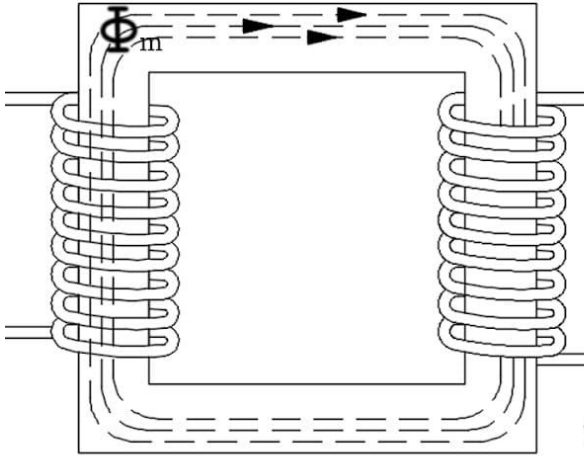
Łukasz Kaiser* Google Brain lukaszkaiser@google.com

Illia Polosukhin* ‡ illia.polosukhin@gmail.com

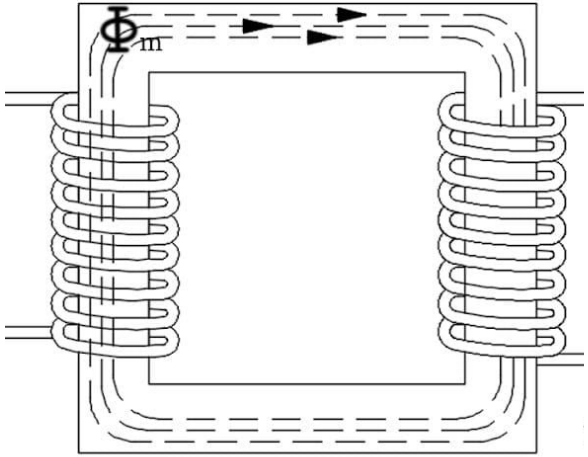
Transformers

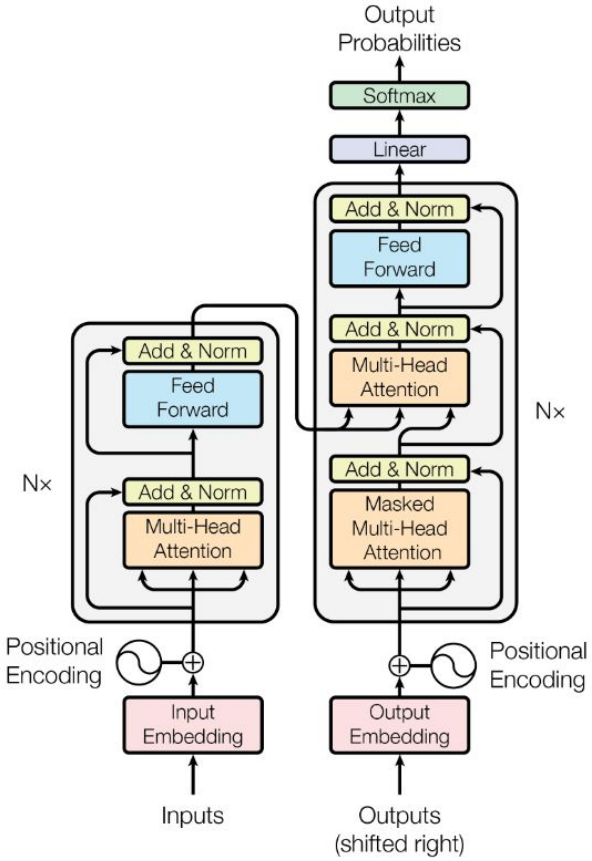


Transformers



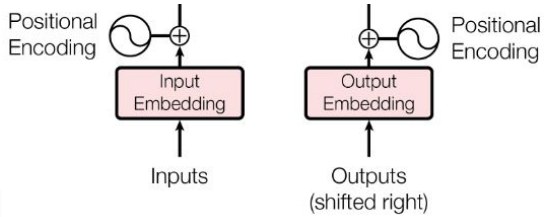
Transformers





Tokenizer

- A tokenizer converts raw text into smaller units (tokens), such as words or subwords, that a model can process numerically.



Step 1: User Types a Message

"Hello world! How are you doing today?"

Step 2: Tokenizer Splits Text into Tokens

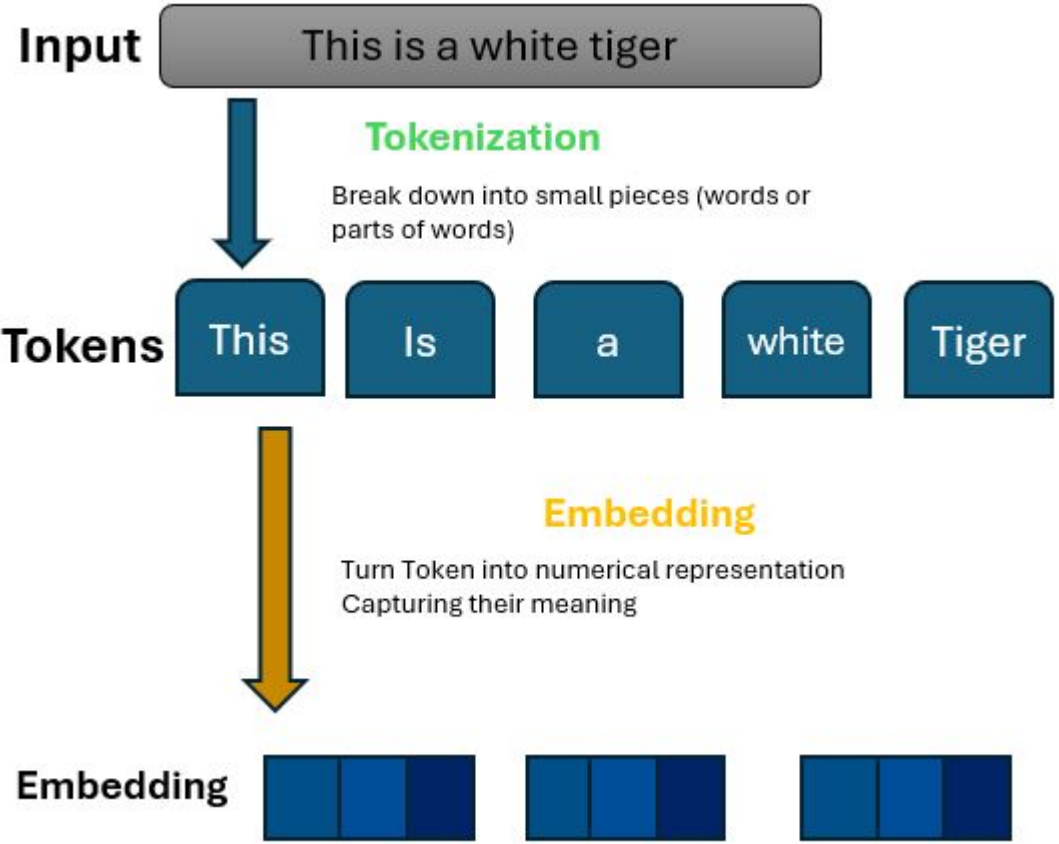
"Hello" | "world" | "!" | "How" | "are" | "you" | "doing" | "today?"

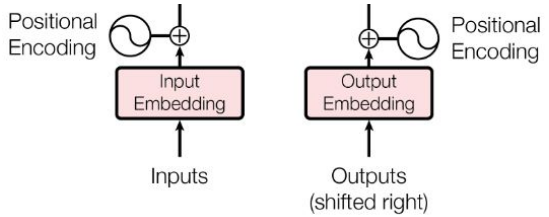
Step 3: Tokens Converted to Numeric IDs

"Hello"	"world"	"!"	"How"	"are"	"you"	"doing"	"today?"
ID: 15496	ID: 3029	ID: 0	ID: 1858	ID: 389	ID: 345	ID: 2503	ID: 9281

LLM receives: [15496, 3029, 0, 1858, 389, 345, 2503, 9281]

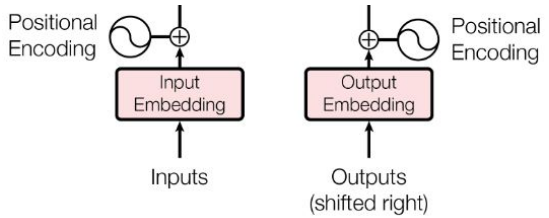
Embedding

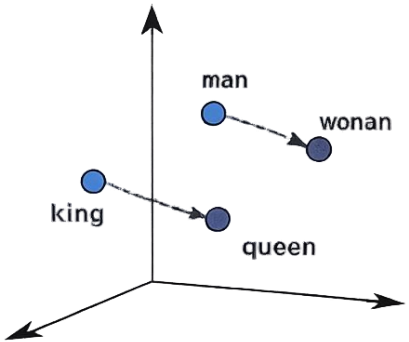




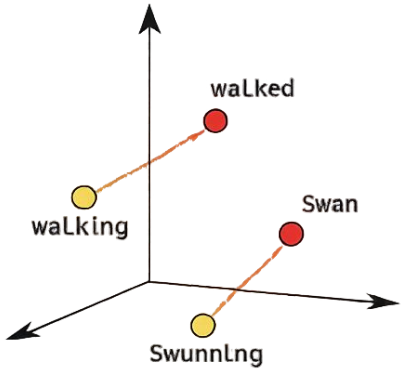
- Convert words, sentences, or data into **numeric vectors**
- Capture **semantic meaning** (similar meanings → similar vectors)
  Represent data in a **continuous vector space**
- Enable models to understand **relationships** (*e.g., king − man + woman ≈ queen*)

Embedding: Relationships

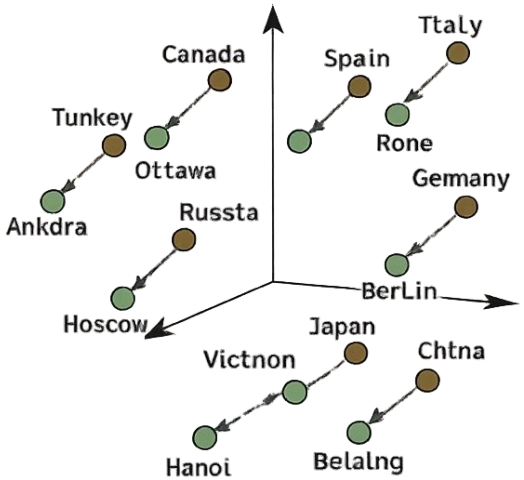




Male-Female

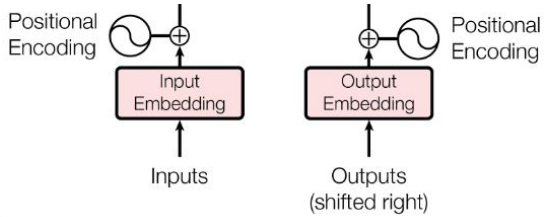


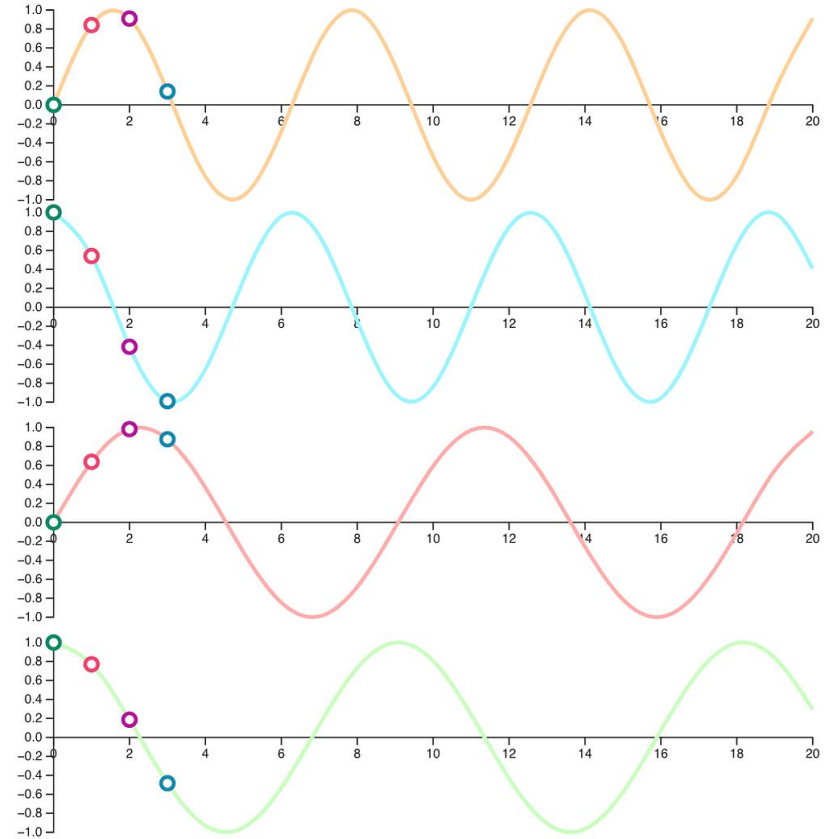
Verb Tense



Country-Capital

Positional Encoding





	p0	p1	p2	p3	
	0.000	0.841	0.909	0.141	**i=0**
	1.000	0.540	-0.416	-0.990	**i=1**
	0.000	0.638	0.983	0.875	**i=2**
	1.000	0.770	0.186	-0.484	**i=3**

Positional Encoding

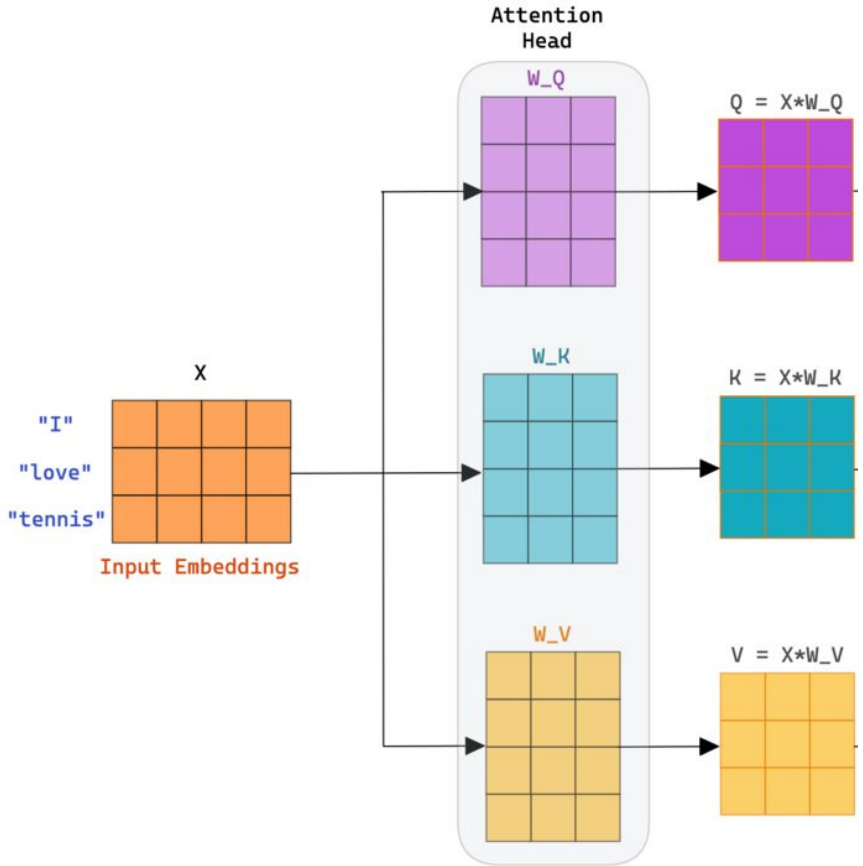
$$PE_{(pos,2i)} = sin(\frac{pos}{10000^{2i/d_{\text{model}}}})$$

$$PE_{(pos,2i+1)} = cos(\frac{pos}{10000^{2i/d_{\text{model}}}})$$

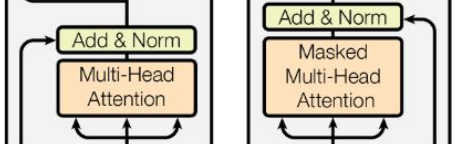
Settings: d = 50

The value of each positional encoding depends on the *position* (*pos*) and *dimension* (*d*). We calculate result for every *index* (*i*) to get the whole vector.

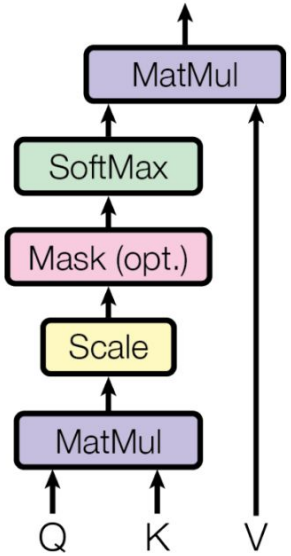
Attention Mechanism



Attention Mechanism

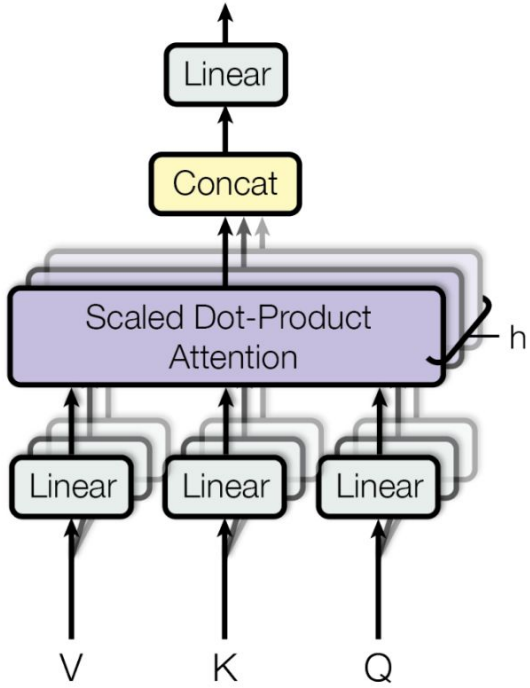


Scaled Dot-Product Attention



$$\text{Attention}(Q, K, V) = \text{softmax}(\frac{QK^T}{\sqrt{d_k}})V$$

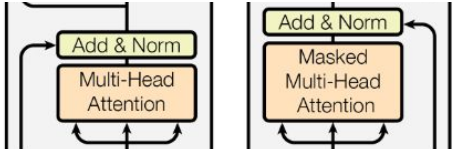
Multi-Head Attention

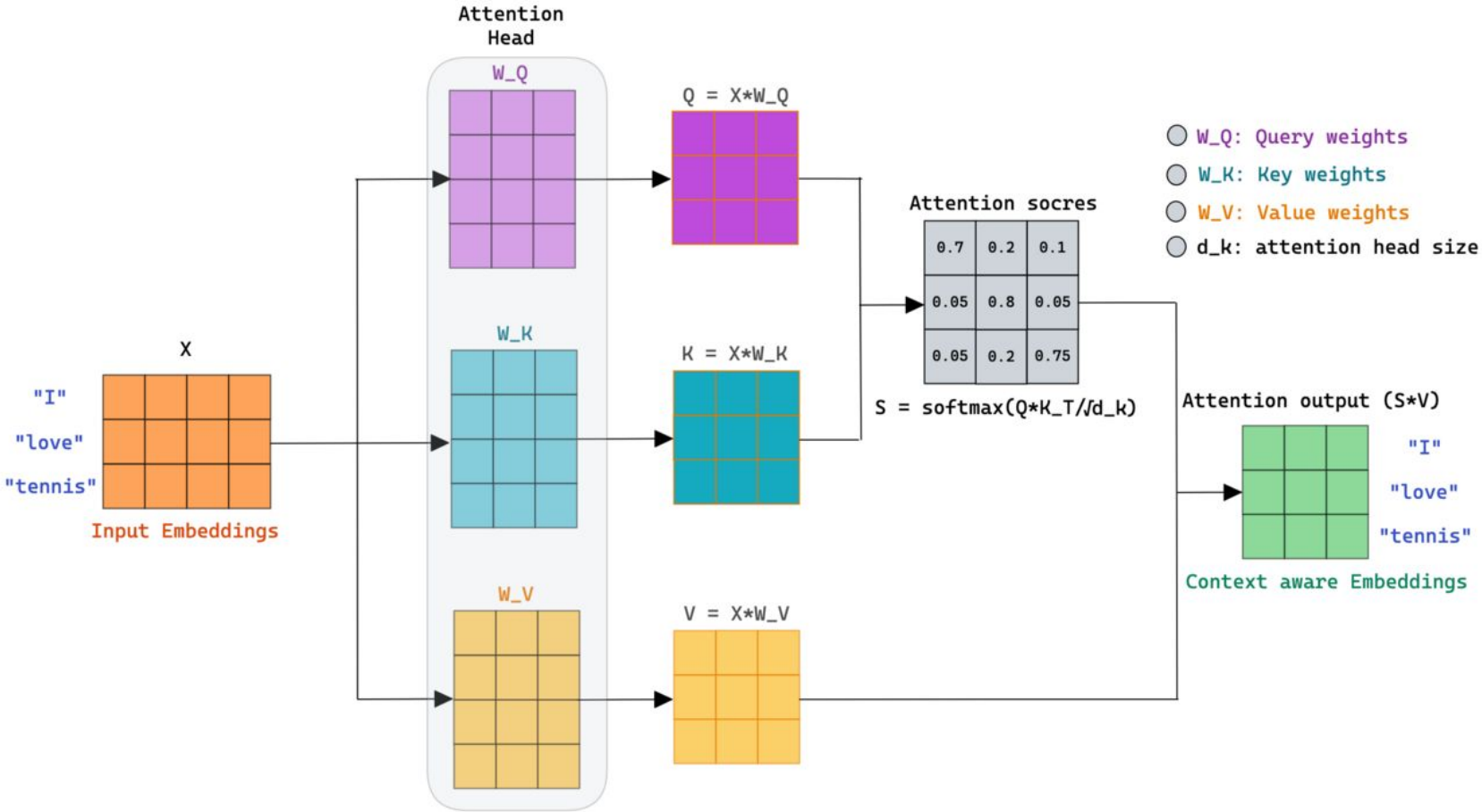


$$\text{MultiHead}(Q, K, V) = \text{Concat}(\text{head}_1, ..., \text{head}_\text{h})W^O$$

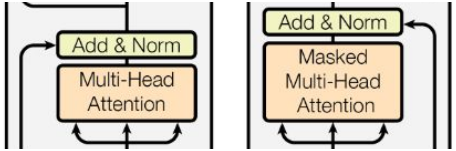
$$\textbf{where } \text{head}_\text{i} = \text{Attention}(QW_i^Q, KW_i^K, VW_i^V)$$

Attention Mechanism





Attention Mechanism



Attention Head

W_Q

Q = X*W_Q

Key

I love tennis

Attention scores

W_Q: Query weights

W_K: Key weights

W_V: Value weights

d_k: attention head size

0.7	0.2	0.1
0.05	0.8	0.05
0.05	0.2	0.75

Query I love tennis

W_K

K = X*W_K

X

"I"

"love"

"tennis"

Input Embeddings

S = softmax(Q*K_T/√d_k)

Attention output (S*V)

W_V

V = X*W_V

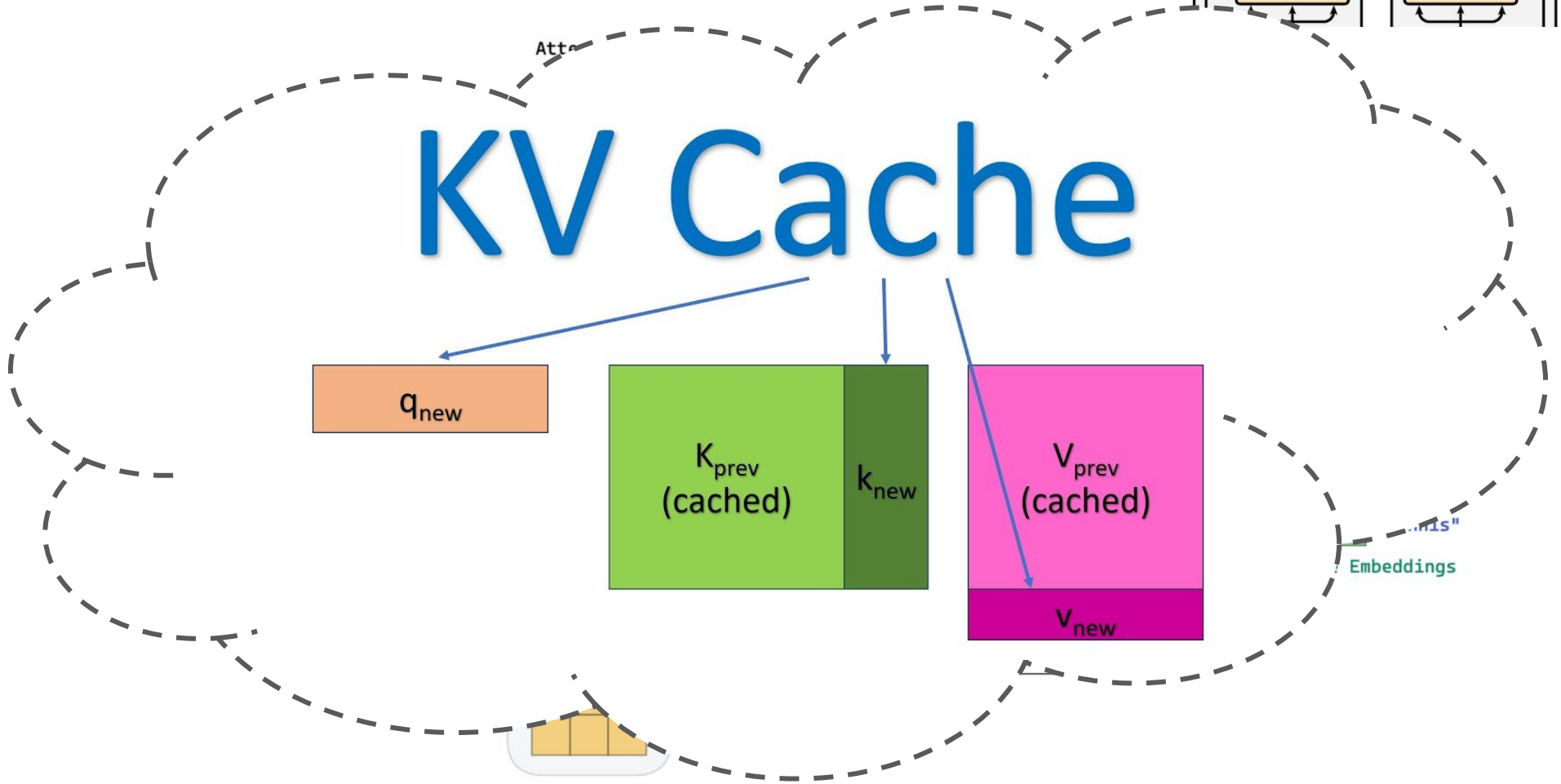
"I"

"love"

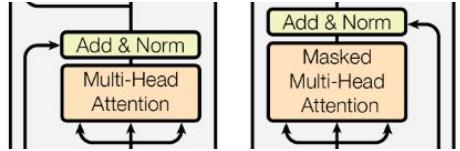
"tennis"

Context aware Embeddings

Attention Mechanism: KV Cache



Attention Mechanism: Temperature



Temperature = 0

my favorite food is pizza
my favorite food is pizza
my favorite food is pizza

Temperature = 0.3

my favorite food is pizza
my favorite food is sushi
my favorite food is pizza

Temperature = 0.7

my favorite food is tacos
my favorite food is sushi
my favorite foo

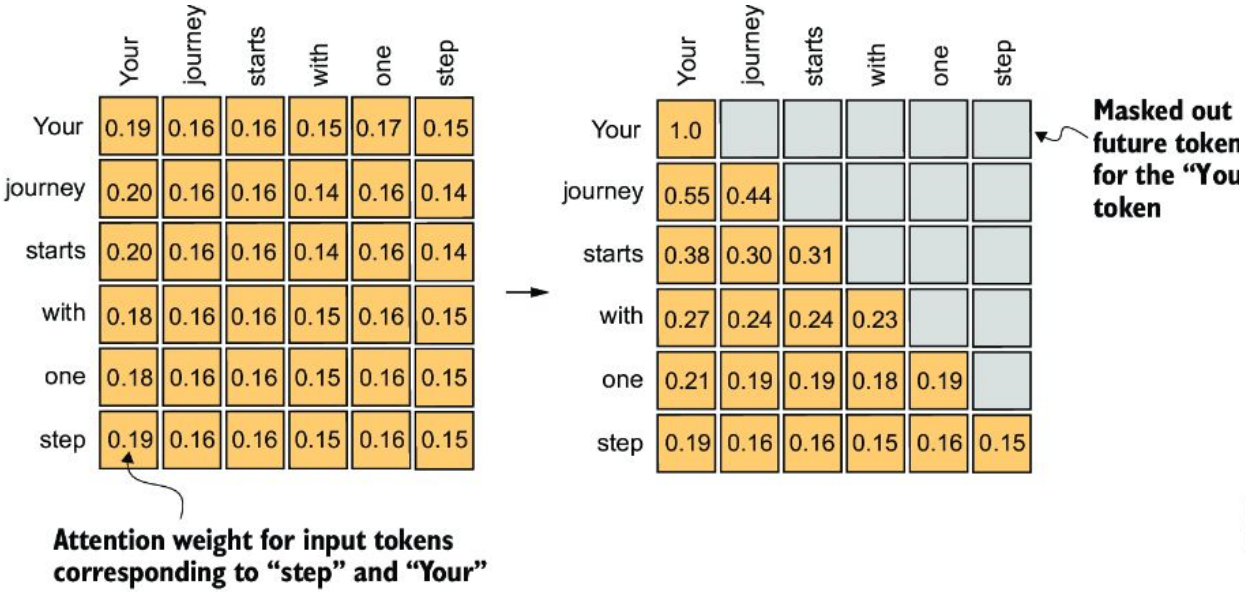
$$\text{Attention}(Q, K, V) = \text{softmax}(\frac{QK^T}{\sqrt{d_k}})V$$

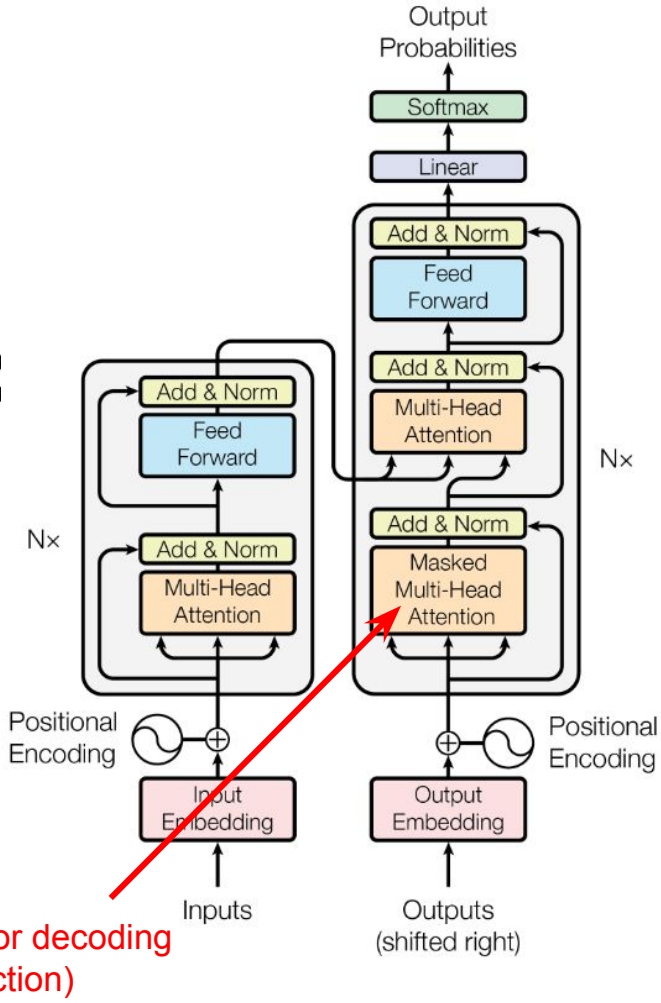
$$p_i = \text{Softmax}(z_i) = \frac{\exp(z_i)}{\sum_{j=1}^{n} \exp(z_j)}$$

$$p_i(T) = \text{Softmax}(z_i/T) = \frac{\exp(z_i/T)}{\sum_{j=1}^{n} \exp(z_j/T)}$$

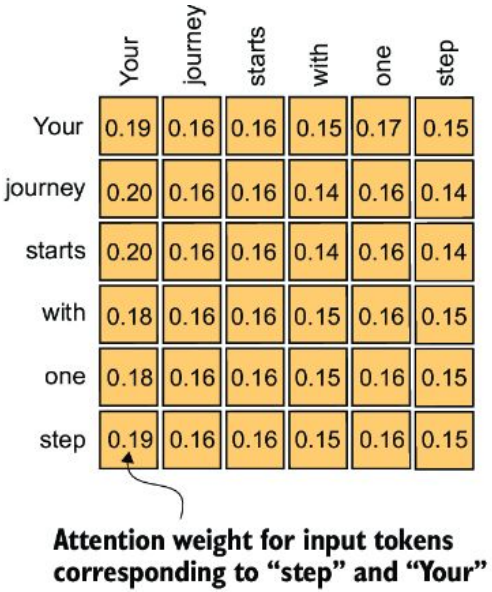
Temperature

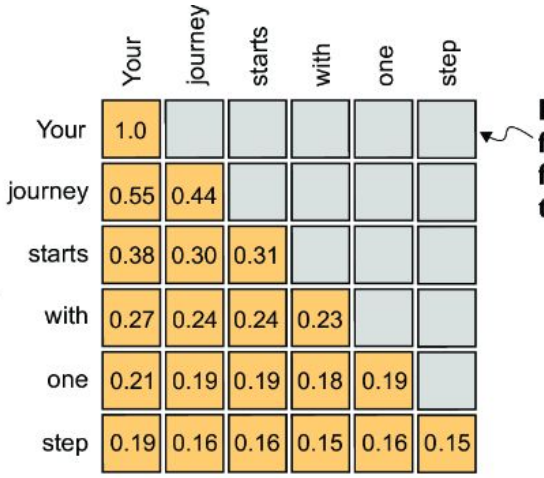
Attention Mechanism: Masked

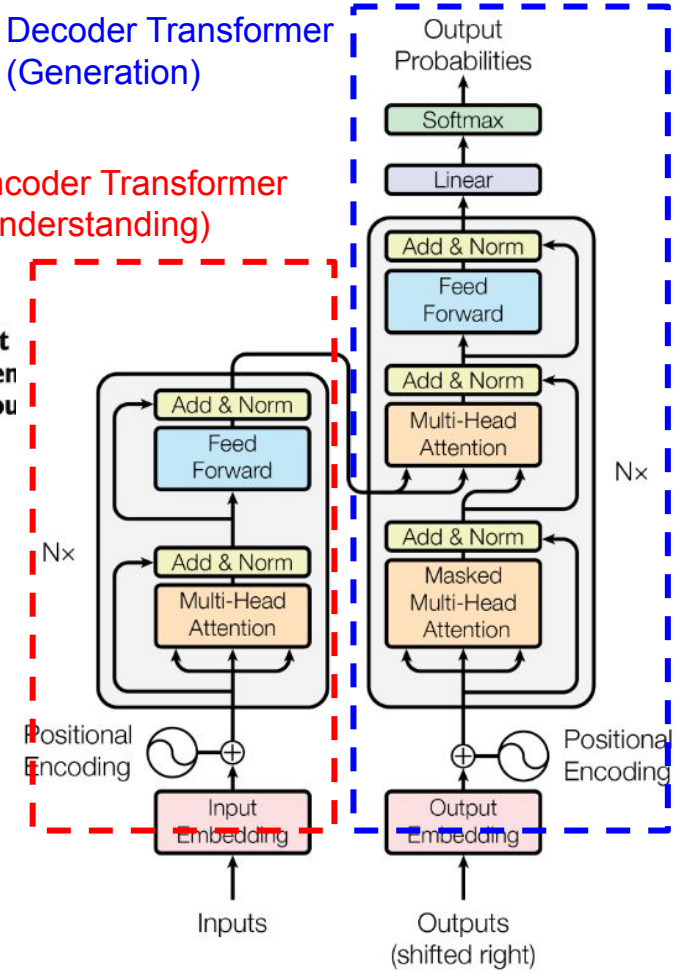




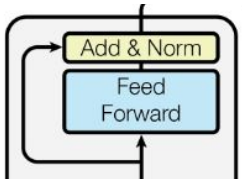
Attention Mechanism: Masked

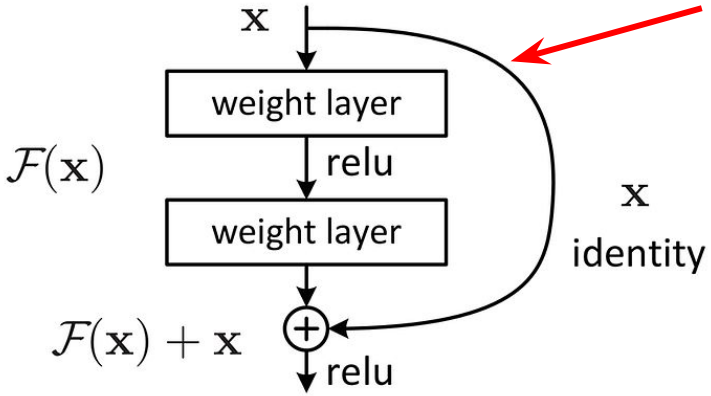






Feedforward Neural Network



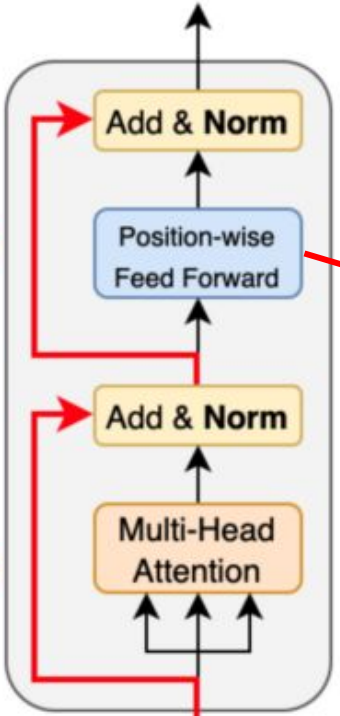


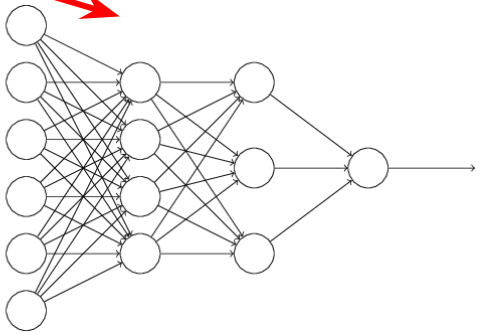
x
identity

residual connection

Encoder Block

residual connection





positional encoding
word embedding

Inputs

Model Training Pipeline

1. Pre-training (Foundation)

- Train model on **massive datasets** (often unlabeled)
- Learns **general patterns** (language, structure, semantics)
- Very **expensive** (data + compute)
- Output: a **general-purpose model**

2. Training (Task Learning / Intermediate)

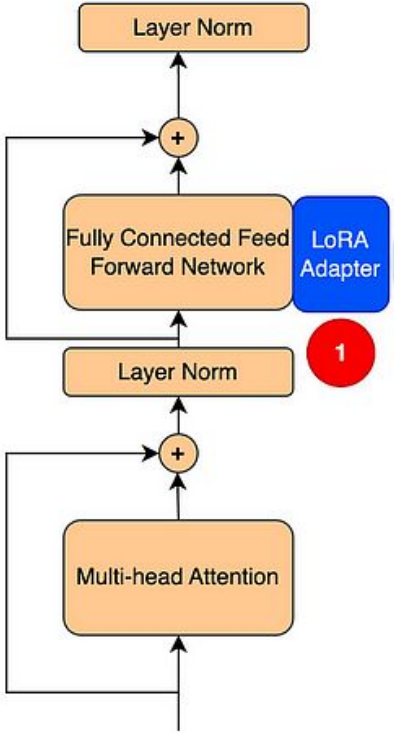
- Adapt model to a **specific problem setup**
- May include:
  - Supervised learning
  - Domain adaptation
- Uses **moderate-size dataset**
- Bridges general knowledge → task understanding

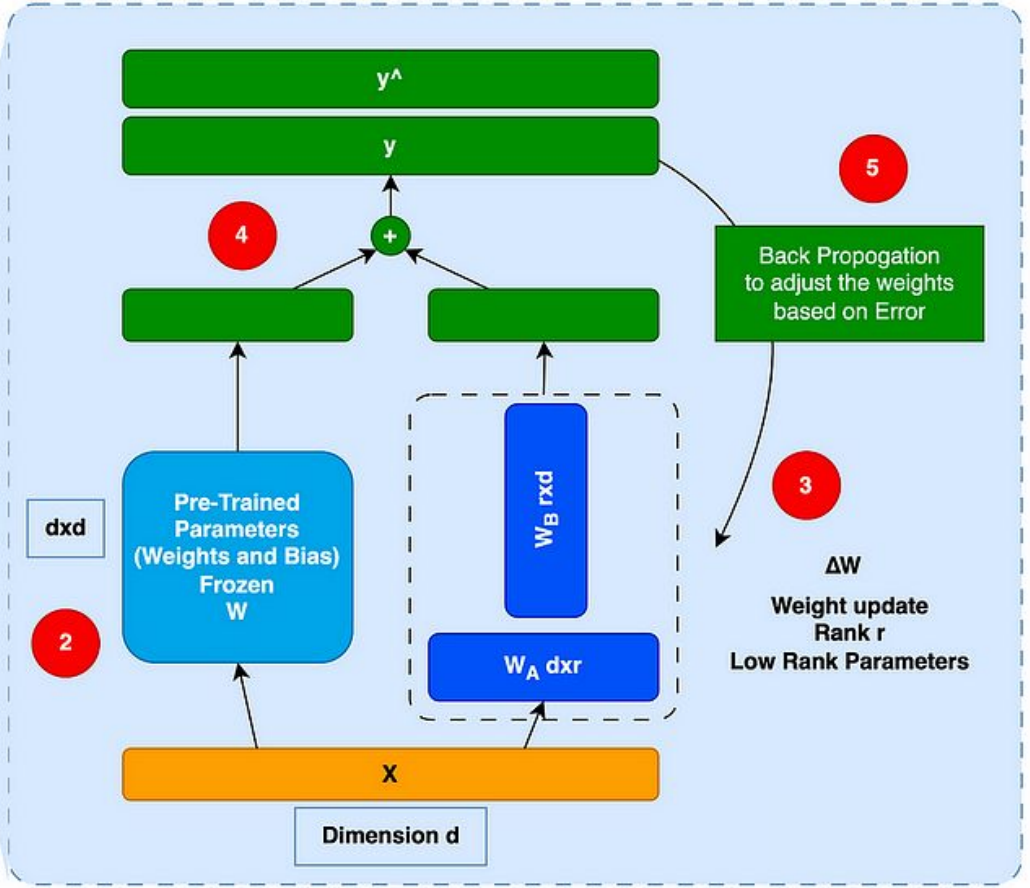
3. Fine-tuning (Specialization)

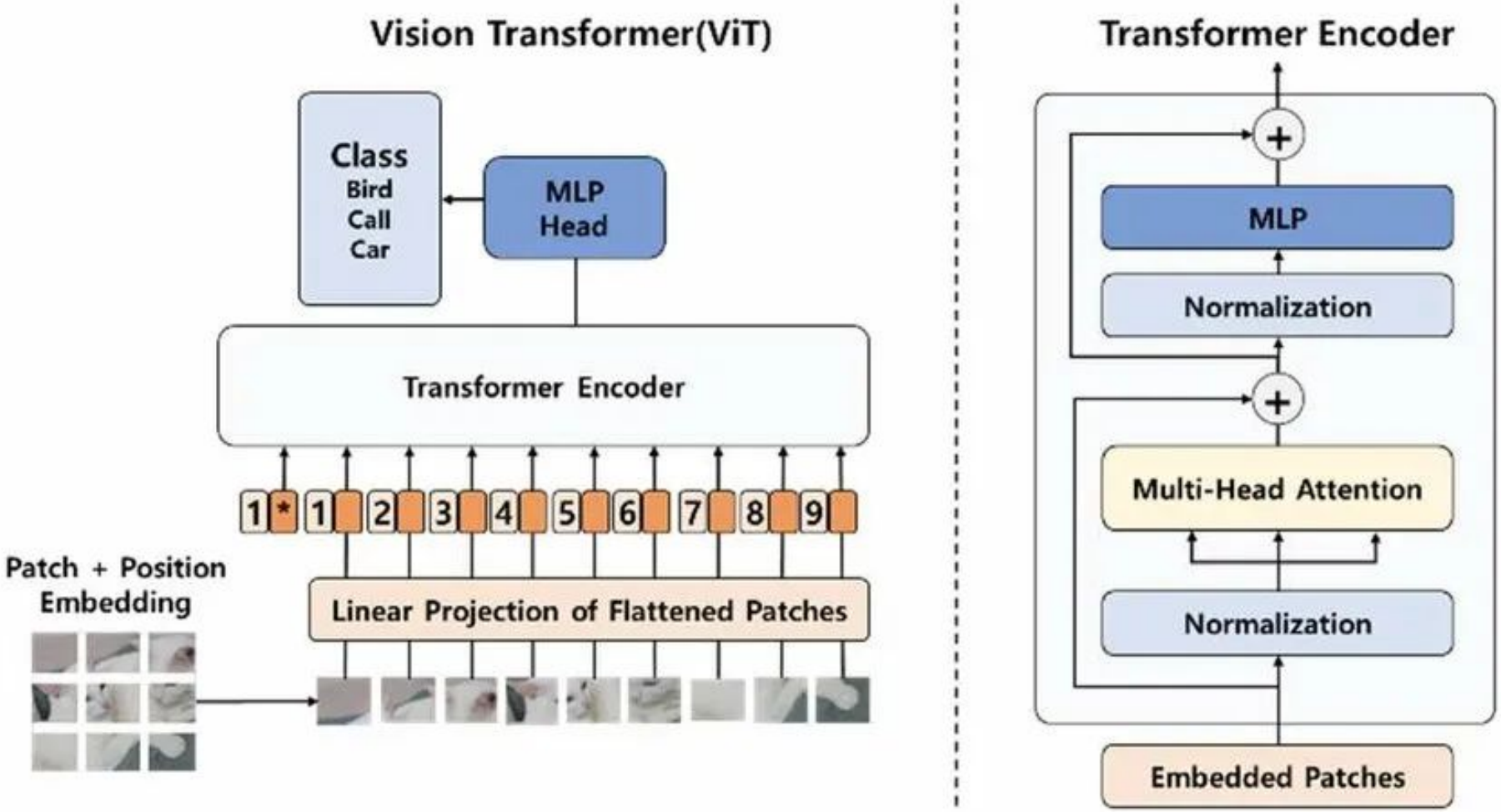
- Train on **small, high-quality dataset**
- Focus on a **specific task or domain**
- Adjusts pretrained weights (often with small learning rate)
- Makes model **accurate and aligned** for real use

Low-Rank Adaptation (LoRA)

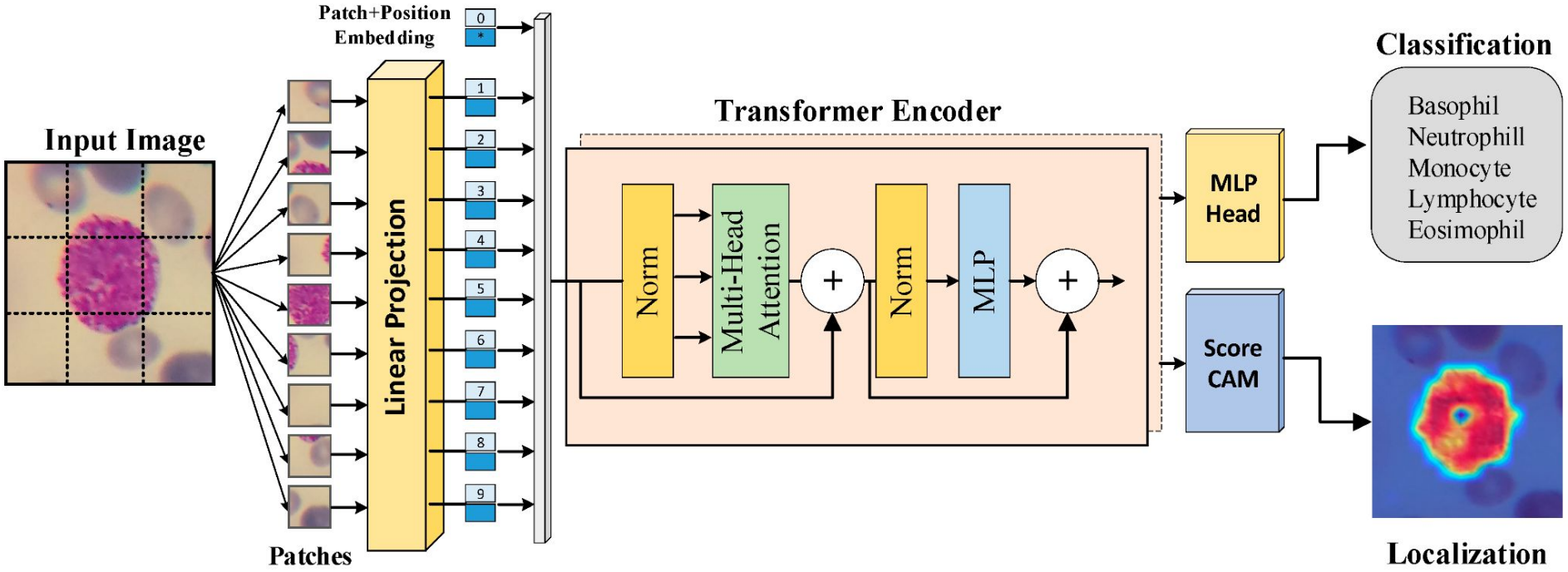
- LoRA is a technique that fine-tunes large models by learning small low-rank updates to pretrained weights instead of modifying the full model.





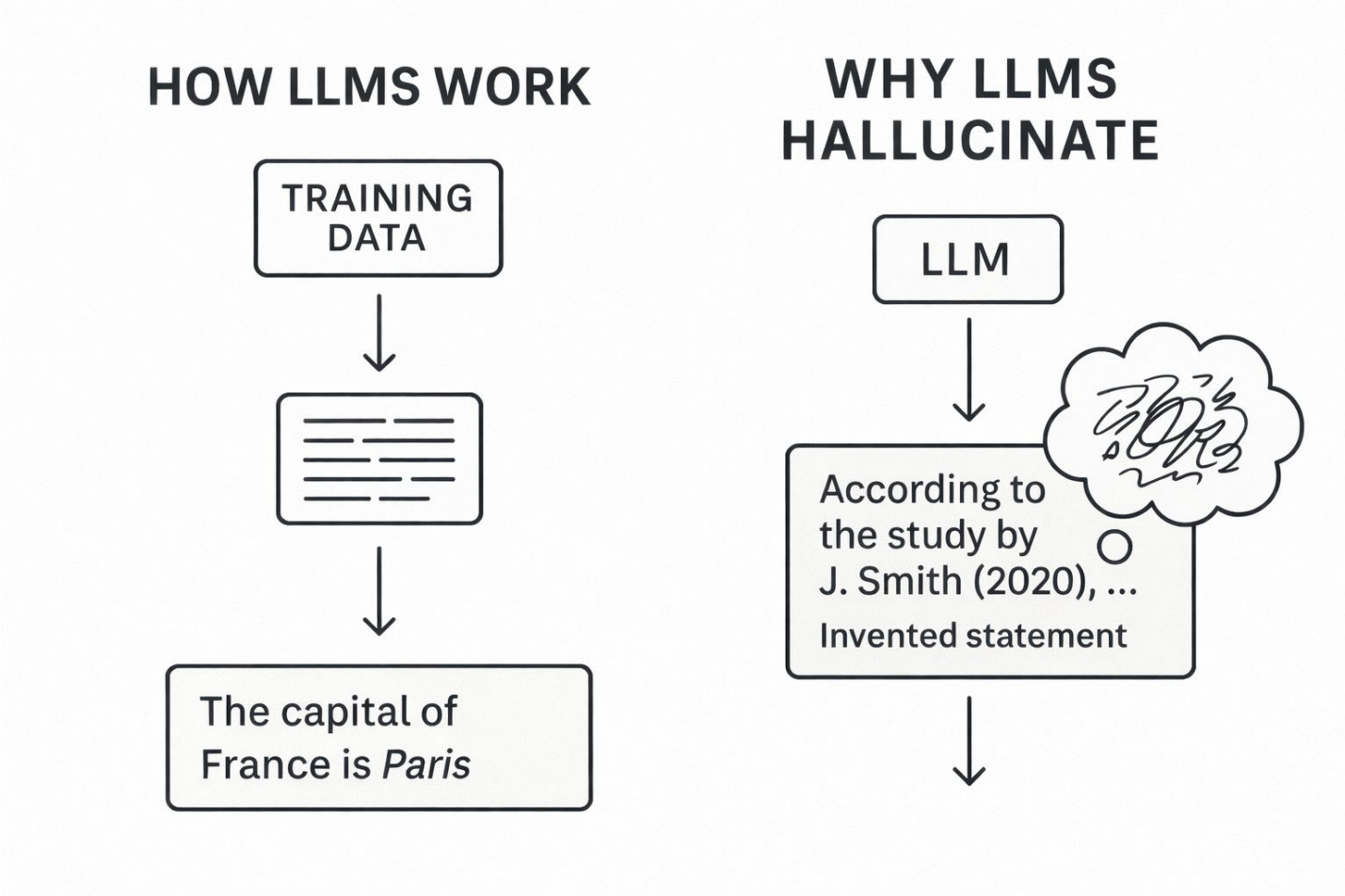

Vision Transformer(ViT)
Class
Bird
Call
Car
MLP
Head
Transformer Encoder
1 * 1 2 3 4 5 6 7 8 9
Patch + Position
Embedding
Linear Projection of Flattened Patches
Transformer Encoder
MLP
Normalization
Multi-Head Attention
Normalization
Embedded Patches

Example of ViT in Medical Application



The Era After
Large Language Models

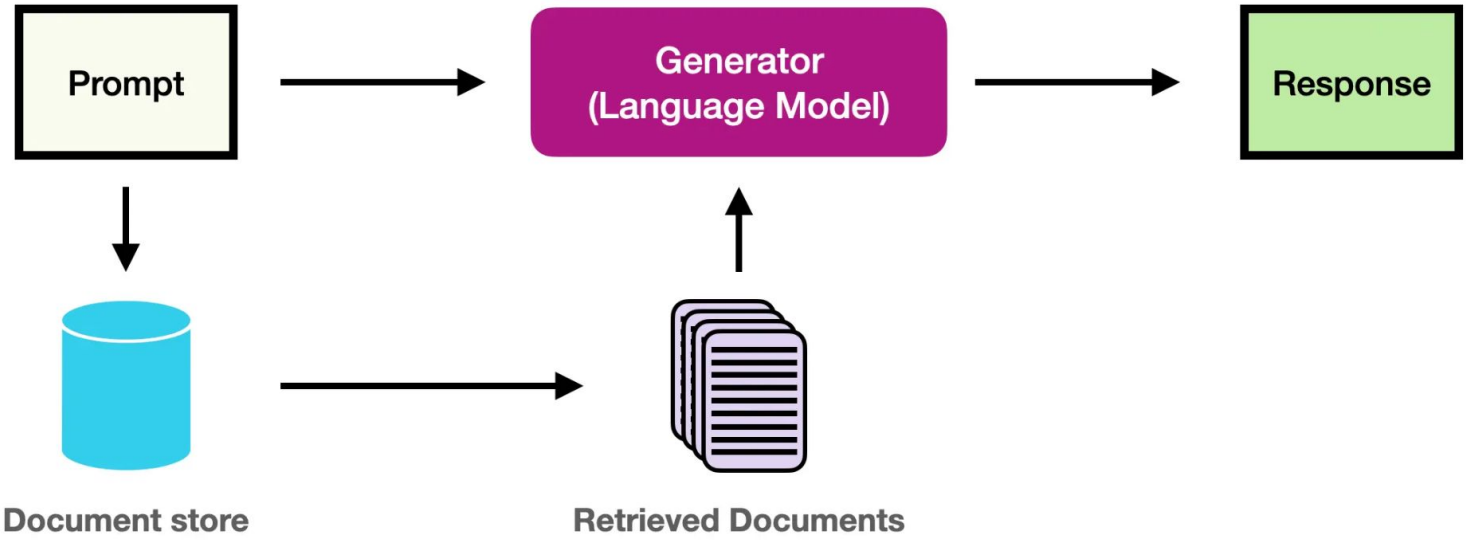
Hallucination



Hallucination

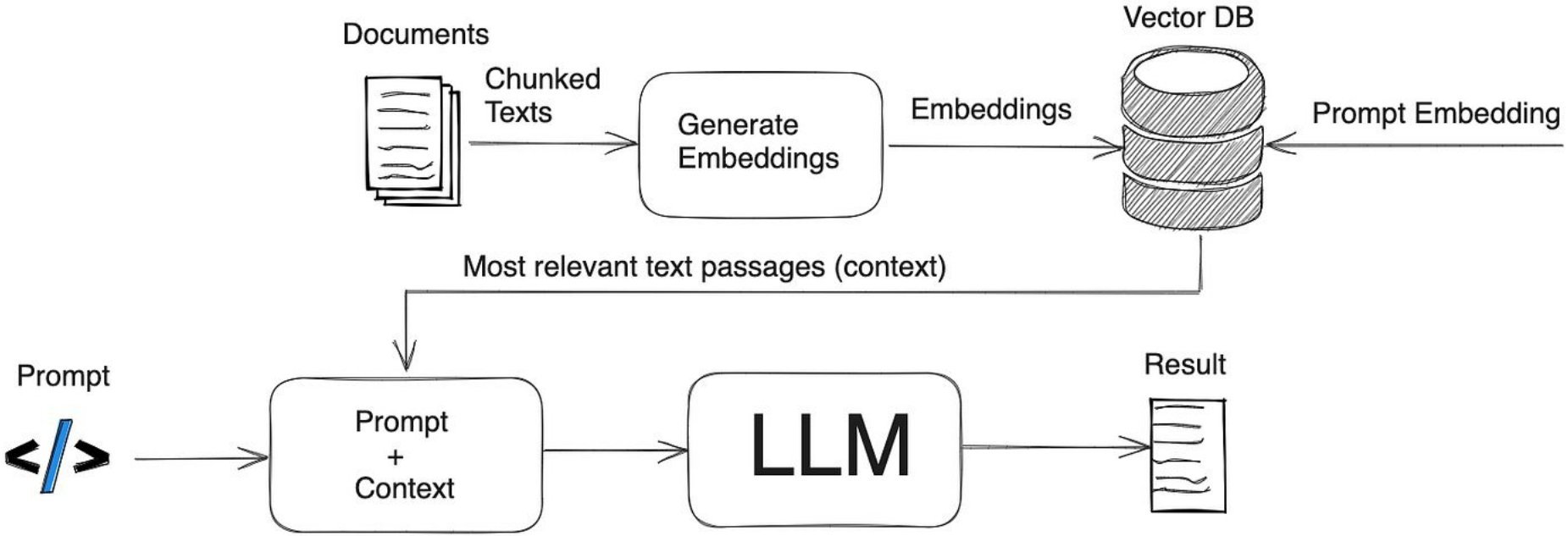
Reference: Fabric

Retrieval Augmented Generation (RAG)

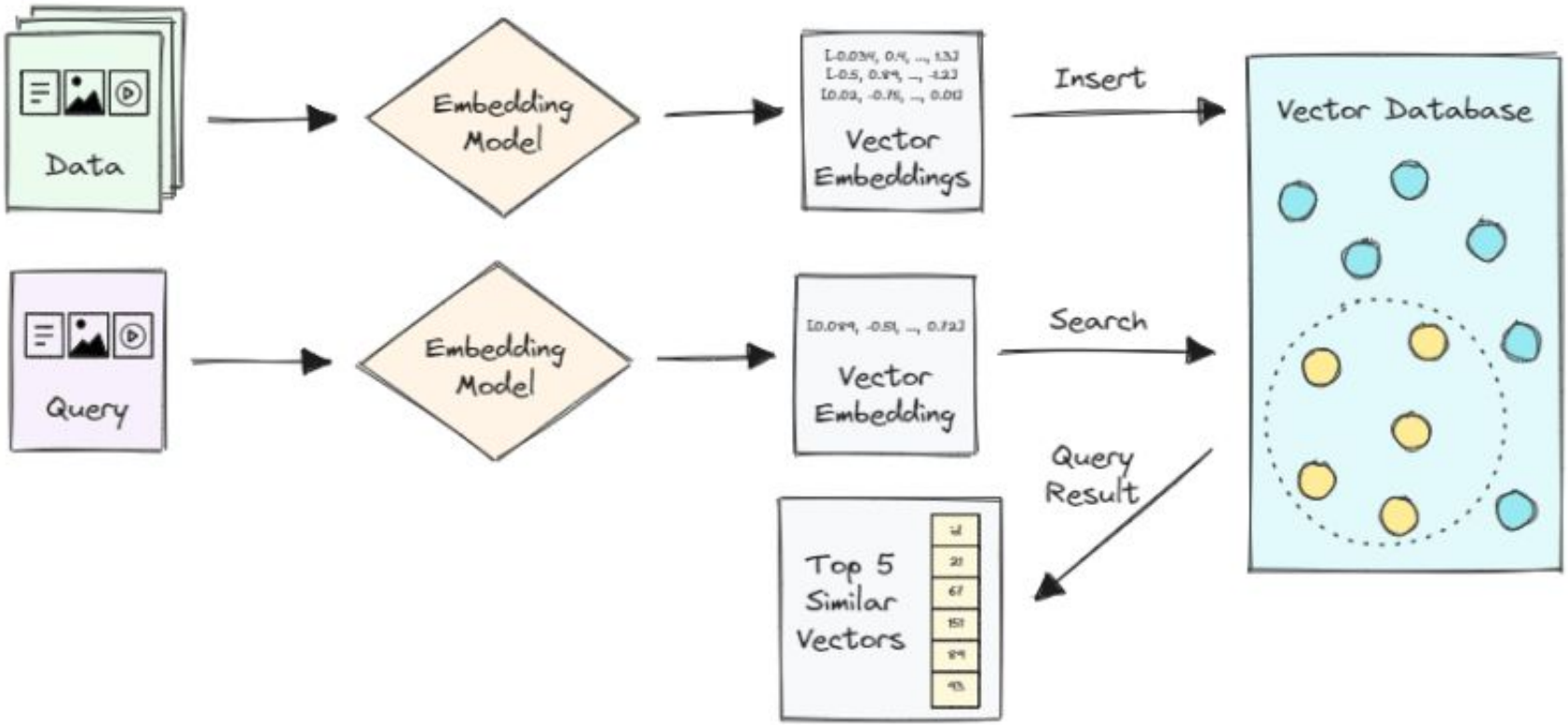


- Combines **retrieval + generation** to produce more accurate responses
- Retrieves **relevant external knowledge** (documents, database, vector search) at query time
- Reduces hallucination by grounding outputs in **real, up-to-date information**

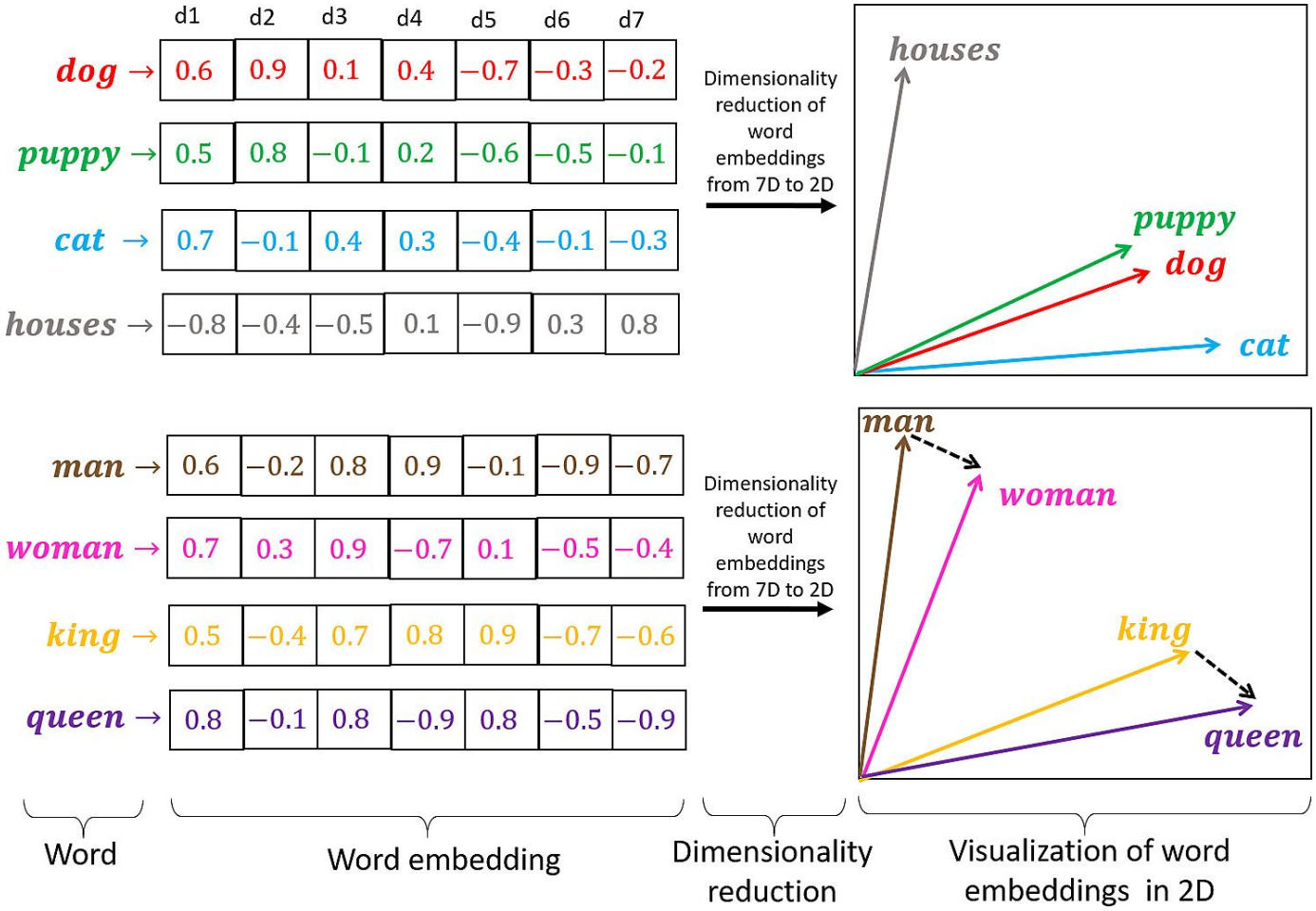
Retrieval Augmented Generation (RAG): Pipeline



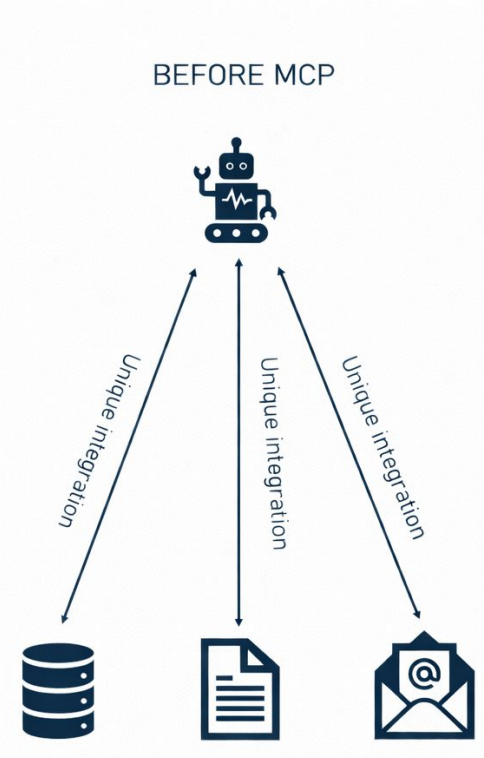
Retrieval Augmented Generation (RAG): Vector Database

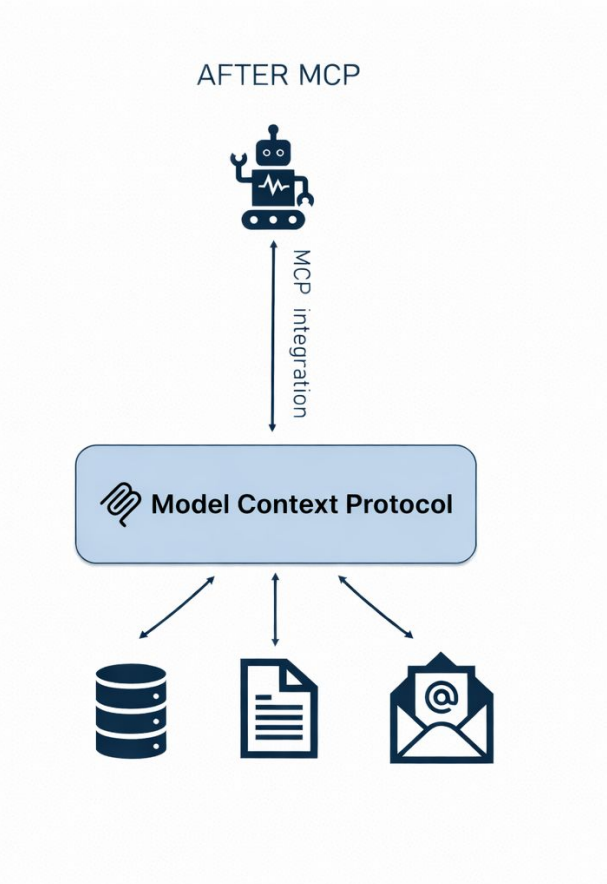


Retrieval Augmented Generation (RAG): Embedding Space



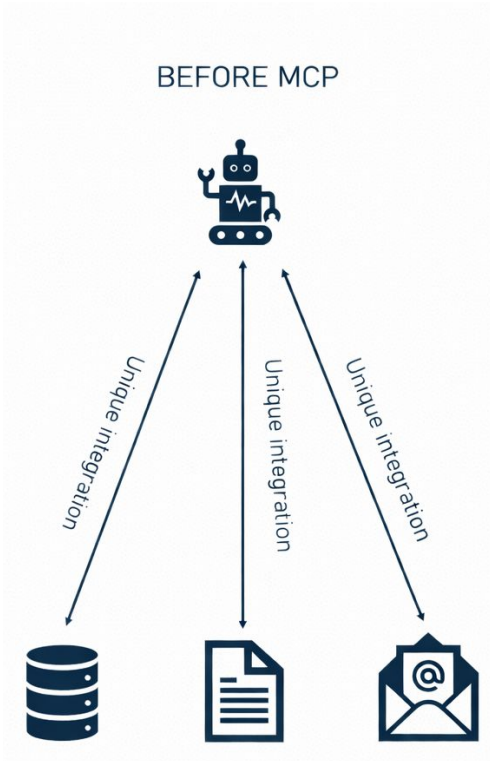
Model Context Protocol (MCP)

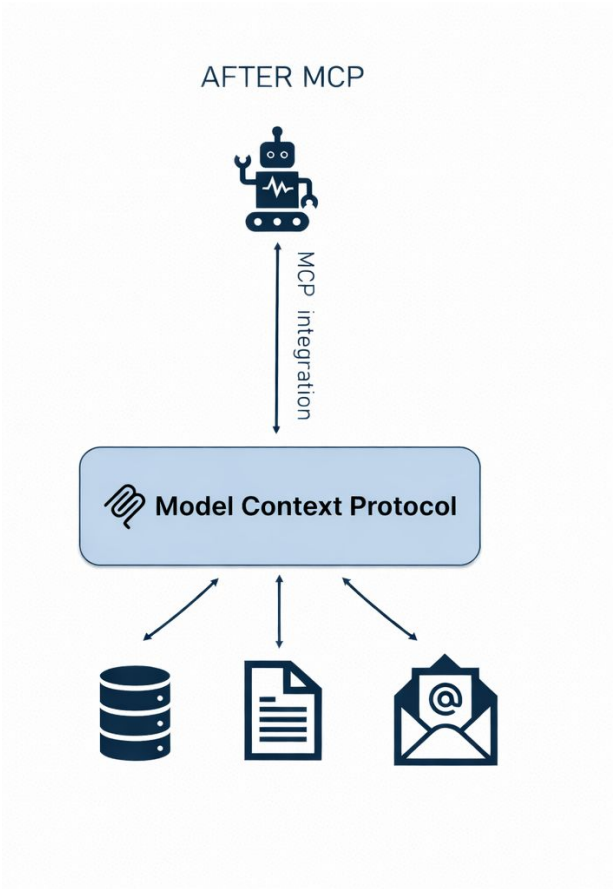




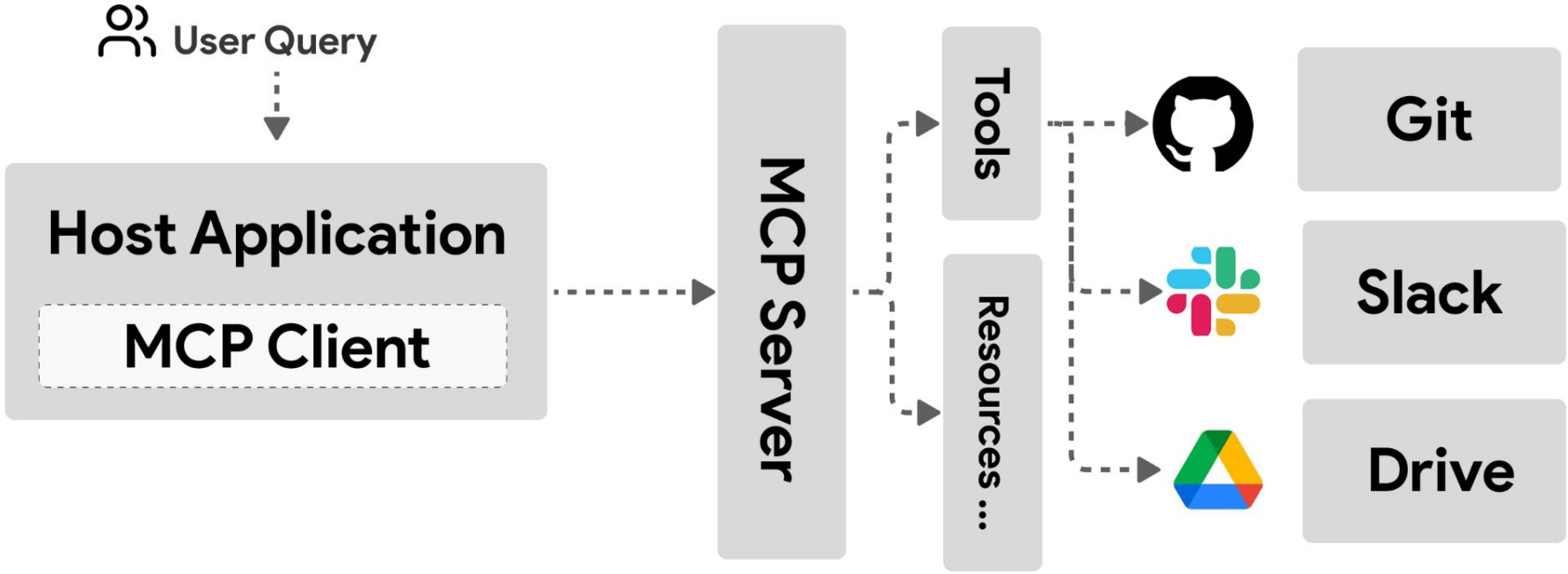
Model Context Protocol (MCP)

Structured data
(time series, tabular)

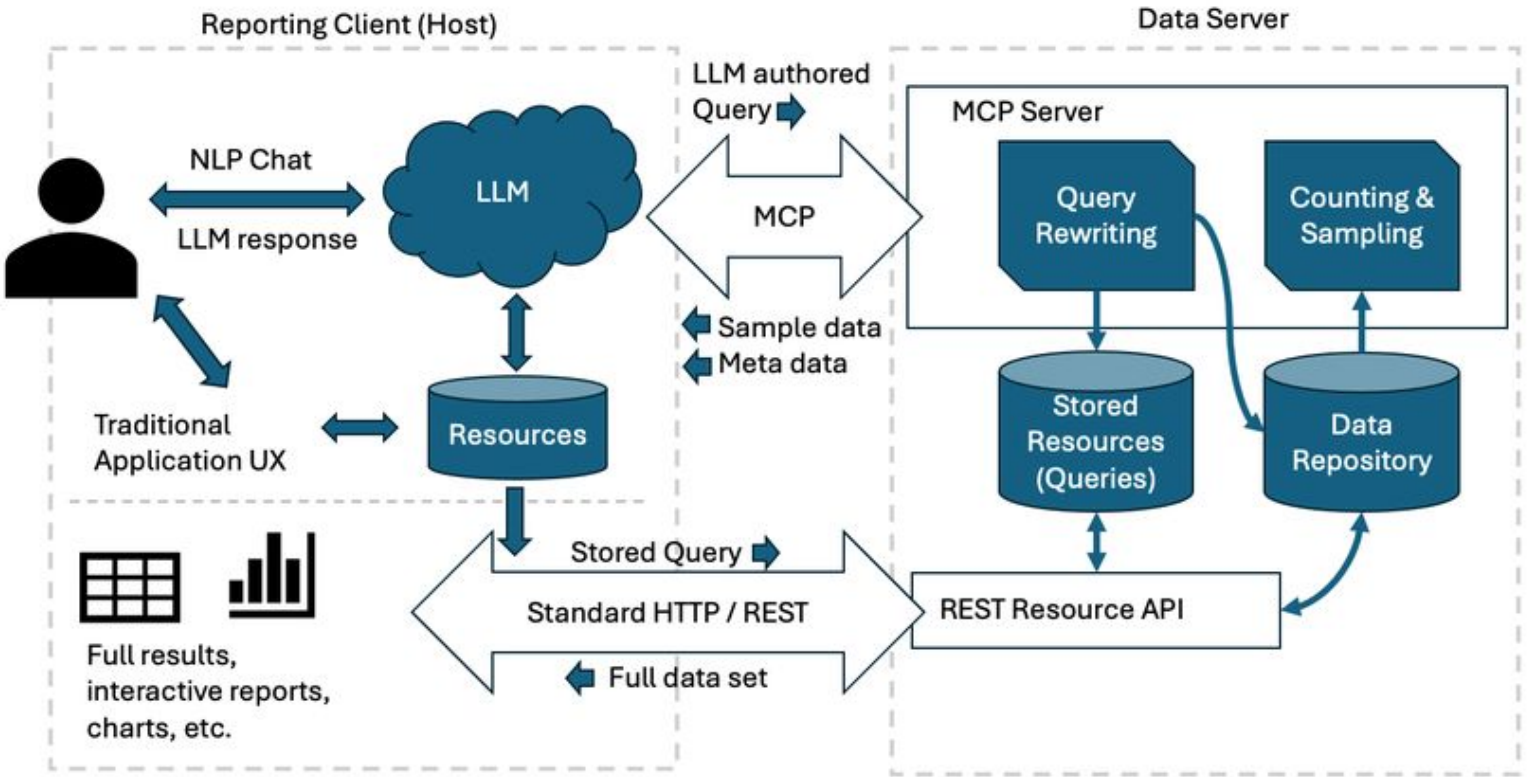




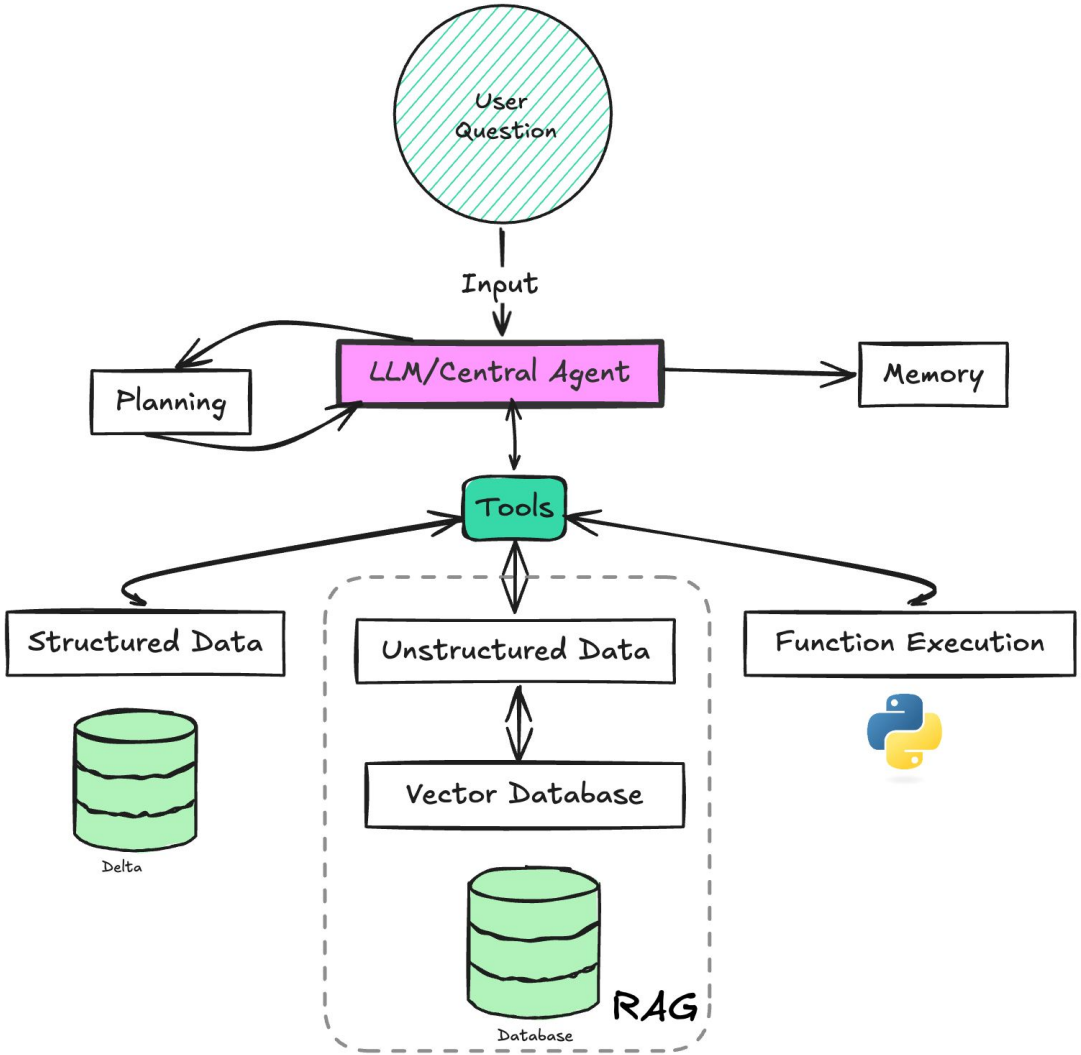
Model Context Protocol (MCP)



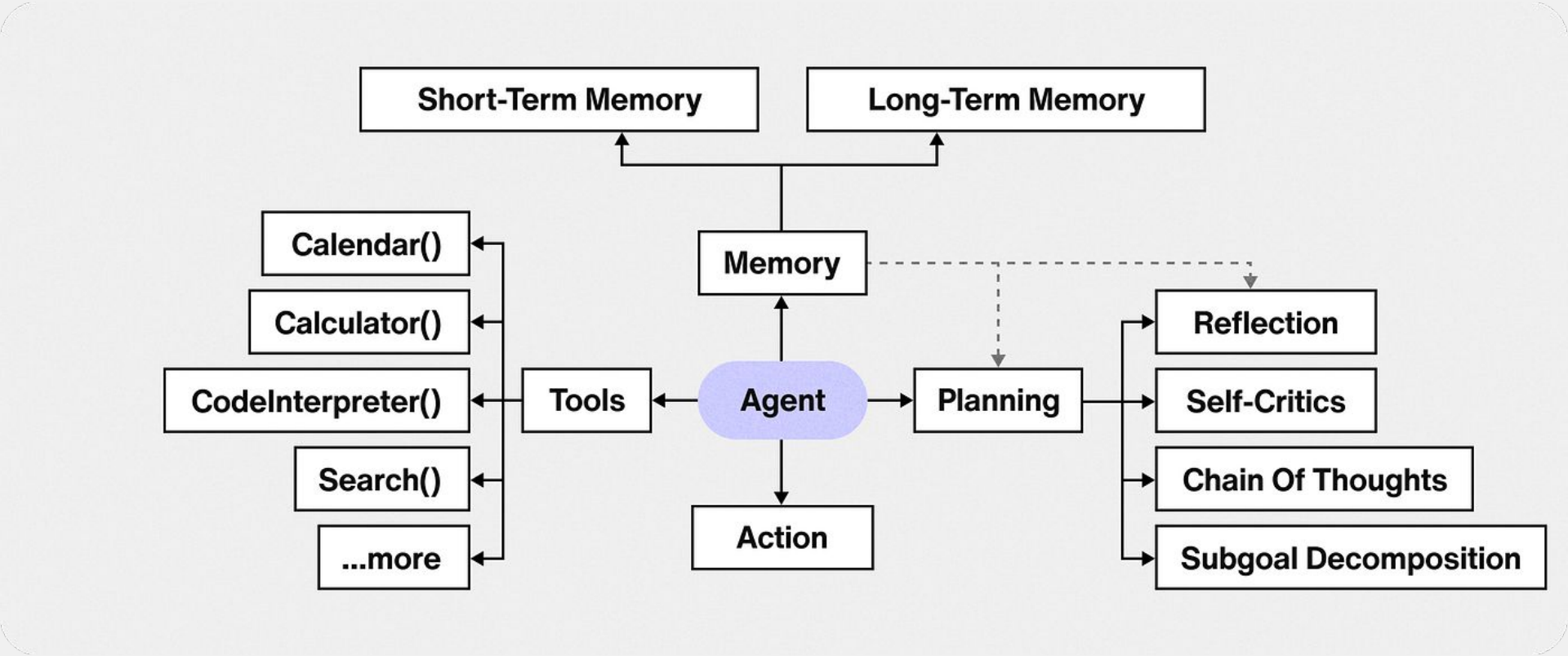
Model Context Protocol (MCP): Technical Pipeline



Agents


User Question
Input
LLM/Central Agent
Planning
Memory
Tools
Structured Data
Unstructured Data
Function Execution
Delta
Vector Database
Database
RAG

Agents


Short-Term Memory
Long-Term Memory
Memory
Calendar()
Calculator()
CodeInterpreter()
Search()
...more
Tools
Agent
Action
Planning
Reflection
Self-Critics
Chain Of Thoughts
Subgoal Decomposition

Modern LLM Architectures (https://sebastianraschka.com/llm-architecture-gallery/)

Conclusion

Conclusion: When (and When Not) to Use Generative AI in Business

- **Is this actually an AI problem?**
  Or could it be solved with simpler approaches like ETL, rules, or basic analytics?

Conclusion: When (and When Not) to Use Generative AI in Business

- **Is this actually an AI problem?**
  Or could it be solved with simpler approaches like ETL, rules, or basic analytics?
- **Do we have the right data?**
  What data exists, and where does it come from?

Conclusion: When (and When Not) to Use Generative AI in Business

- **Is this actually an AI problem?**
  Or could it be solved with simpler approaches like ETL, rules, or basic analytics?
- **Do we have the right data?**
  What data exists, and where does it come from?
- **What type of data are we dealing with? What is the data modality?**
  Structured (tables, time series) or unstructured (text, images, audio)?

Conclusion: When (and When Not) to Use Generative AI in Business

- **Is this actually an AI problem?**
  Or could it be solved with simpler approaches like ETL, rules, or basic analytics?
- **Do we have the right data?**
  What data exists, and where does it come from?
- **What type of data are we dealing with? What is the data modality?**
  Structured (tables, time series) or unstructured (text, images, audio)?
- **What is the actual problem to solve?**
  Classification, prediction, generation, summarization, etc.

Conclusion: When (and When Not) to Use Generative AI in Business

- **Is this actually an AI problem?**
  Or could it be solved with simpler approaches like ETL, rules, or basic analytics?
- **Do we have the right data?**
  What data exists, and where does it come from?
- **What type of data are we dealing with? What is the data modality?**
  Structured (tables, time series) or unstructured (text, images, audio)?
- **What is the actual problem to solve?**
  Classification, prediction, generation, summarization, etc.
- **Is our data suitable for this task?**
  Enough volume, quality, and relevance?

Conclusion: When (and When Not) to Use Generative AI in Business

- **Is this actually an AI problem?**
  Or could it be solved with simpler approaches like ETL, rules, or basic analytics?
- **Do we have the right data?**
  What data exists, and where does it come from?
- **What type of data are we dealing with? What is the data modality?**
  Structured (tables, time series) or unstructured (text, images, audio)?
- **What is the actual problem to solve?**
  Classification, prediction, generation, summarization, etc.
- **Is our data suitable for this task?**
  Enough volume, quality, and relevance?
- **Do we need deterministic results?**
  Or can we tolerate some variability and uncertainty?

Conclusion: When (and When Not) to Use Generative AI in Business

- **Is this actually an AI problem?**
  Or could it be solved with simpler approaches like ETL, rules, or basic analytics?
- **Do we have the right data?**
  What data exists, and where does it come from?
- **What type of data are we dealing with? What is the data modality?**
  Structured (tables, time series) or unstructured (text, images, audio)?
- **What is the actual problem to solve?**
  Classification, prediction, generation, summarization, etc.
- **Is our data suitable for this task?**
  Enough volume, quality, and relevance?
- **Do we need deterministic results?**
  Or can we tolerate some variability and uncertainty?
- **How much uncertainty is acceptable?**
  Is approximation okay, or do we need precise answers?

Conclusion: When (and When Not) to Use Generative AI in Business

- **Is this actually an AI problem?**
  Or could it be solved with simpler approaches like ETL, rules, or basic analytics?
- **Do we have the right data?**
  What data exists, and where does it come from?
- **What type of data are we dealing with? What is the data modality?**
  Structured (tables, time series) or unstructured (text, images, audio)?
- **What is the actual problem to solve?**
  Classification, prediction, generation, summarization, etc.
- **Is our data suitable for this task?**
  Enough volume, quality, and relevance?
- **Do we need deterministic results?**
  Or can we tolerate some variability and uncertainty?
- **How much uncertainty is acceptable?**
  Is approximation okay, or do we need precise answers?
- **Is a generative model the right fit?**
  Or would a traditional ML or rule-based system work better?

Conclusion: When (and When Not) to Use Generative AI in Business

- **Is this actually an AI problem?**
  Or could it be solved with simpler approaches like ETL, rules, or basic analytics?
- **Do we have the right data?**
  What data exists, and where does it come from?
- **What type of data are we dealing with? What is the data modality?**
  Structured (tables, time series) or unstructured (text, images, audio)?
- **What is the actual problem to solve?**
  Classification, prediction, generation, summarization, etc.
- **Is our data suitable for this task?**
  Enough volume, quality, and relevance?
- **Do we need deterministic results?**
  Or can we tolerate some variability and uncertainty?
- **How much uncertainty is acceptable?**
  Is approximation okay, or do we need precise answers?
- **Is a generative model the right fit?**
  Or would a traditional ML or rule-based system work better?
- **Non-functional requirements**
  Privacy, security, interpretability, compliance, etc.

UNIVERSITY OF ALBERTA

NATIONAL BANK

Thank You!

Q&A

Personal Website

Learning By Machine