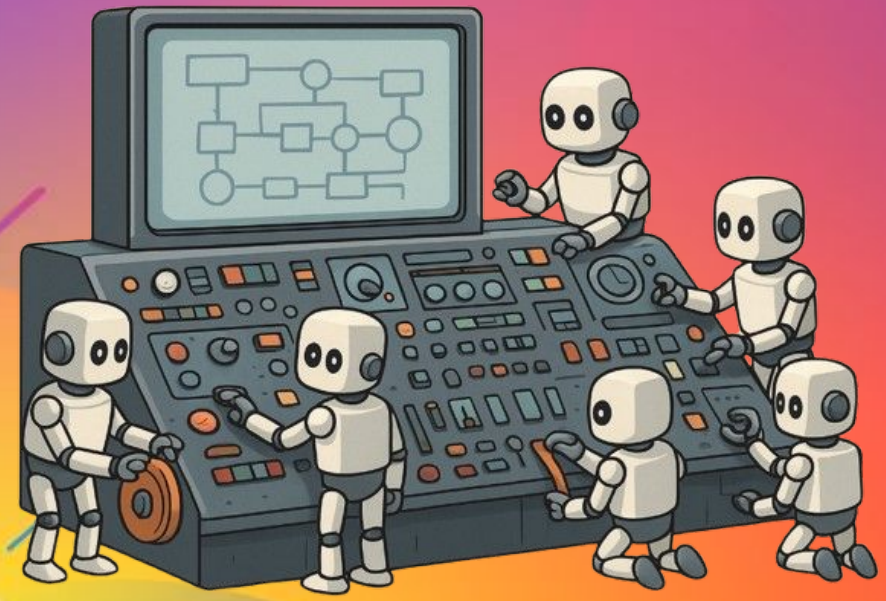


AI Agents using Small Language Models

MICAI 2025

INAOE

Hi, welcome to the tutorial.
We're happy to have you here!

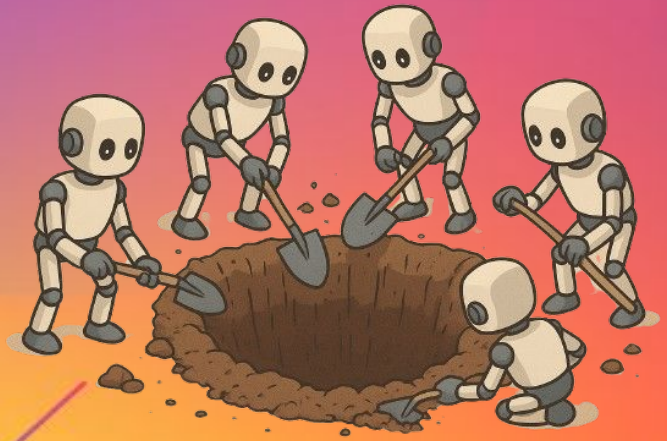
Jeanfed Ramírez-Lima
INAOE, Ford, jeanfed.ramirez@gmail.com

Luis Joaquín Arellano
INAOE, arellano.luis.3c@gmail.com

Hugo Jair Escalante
INAOE, hugo.jair@gmail.com

2025 MICAI

INAOE



Content

- Machine learning definition
- Deep learning history
- Neural architectures

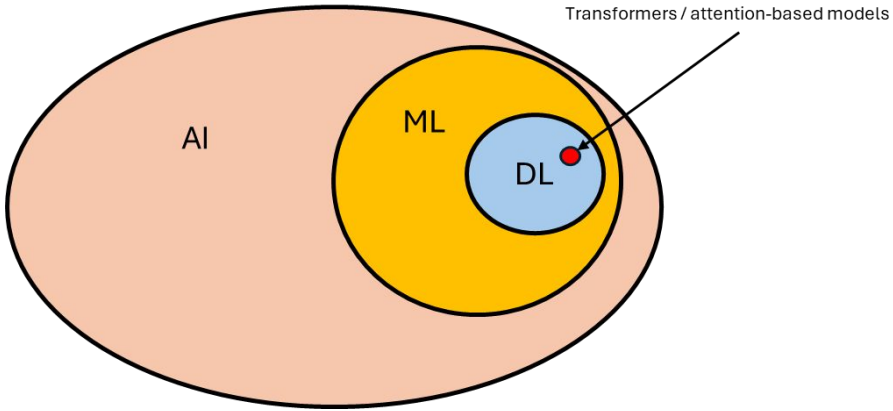
MICCAI 2025

INAOE

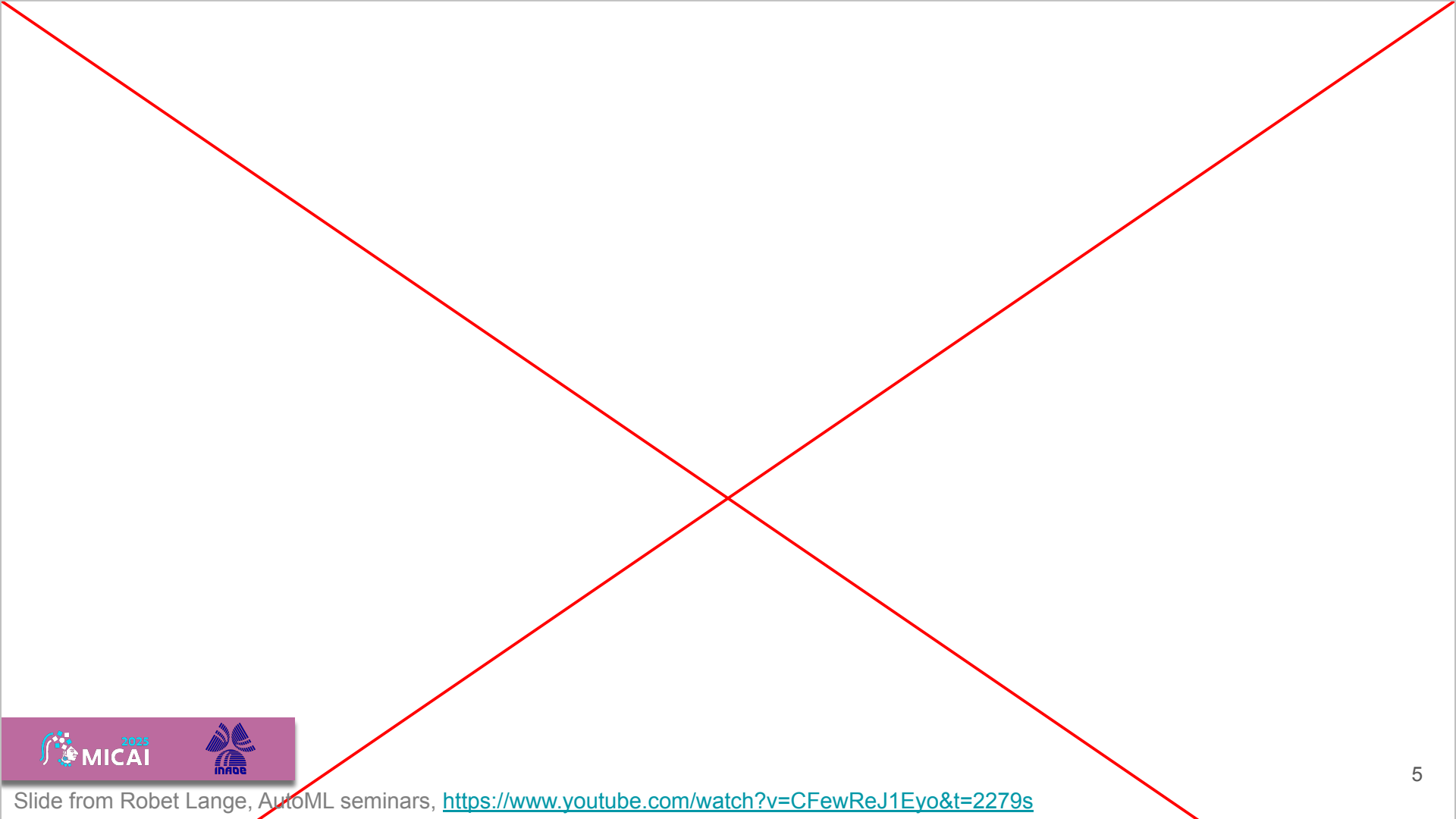
Deep Learning Intro

Machine Learning and Artificial Intelligence

ML is a subfield of AI focusing on the development of computer programs that automatically adapt their behavior from their interaction with data. They can "*learn*" without being programmed explicitly



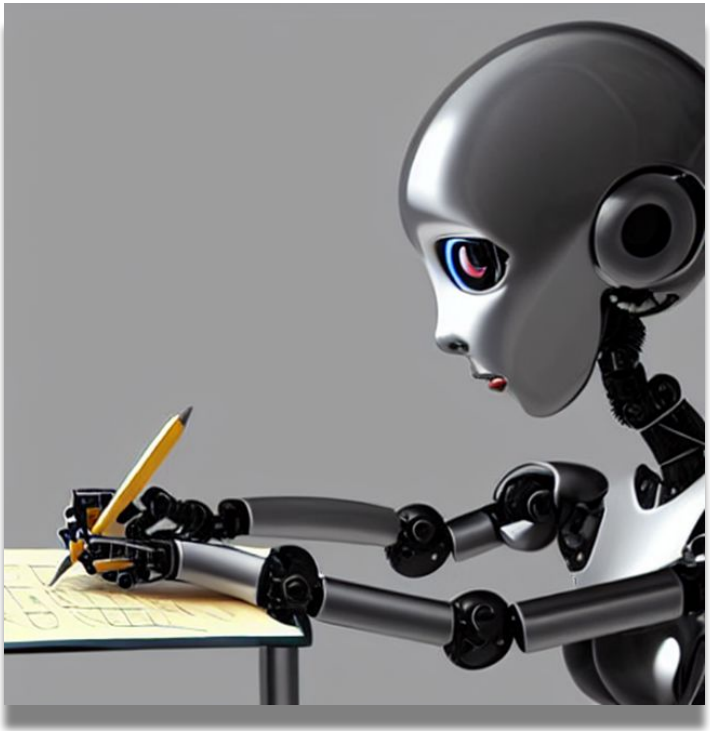
T. M. Mitchell. **The Discipline of Machine Learning.** Tech. Report. CMU-ML-06-108, 2006



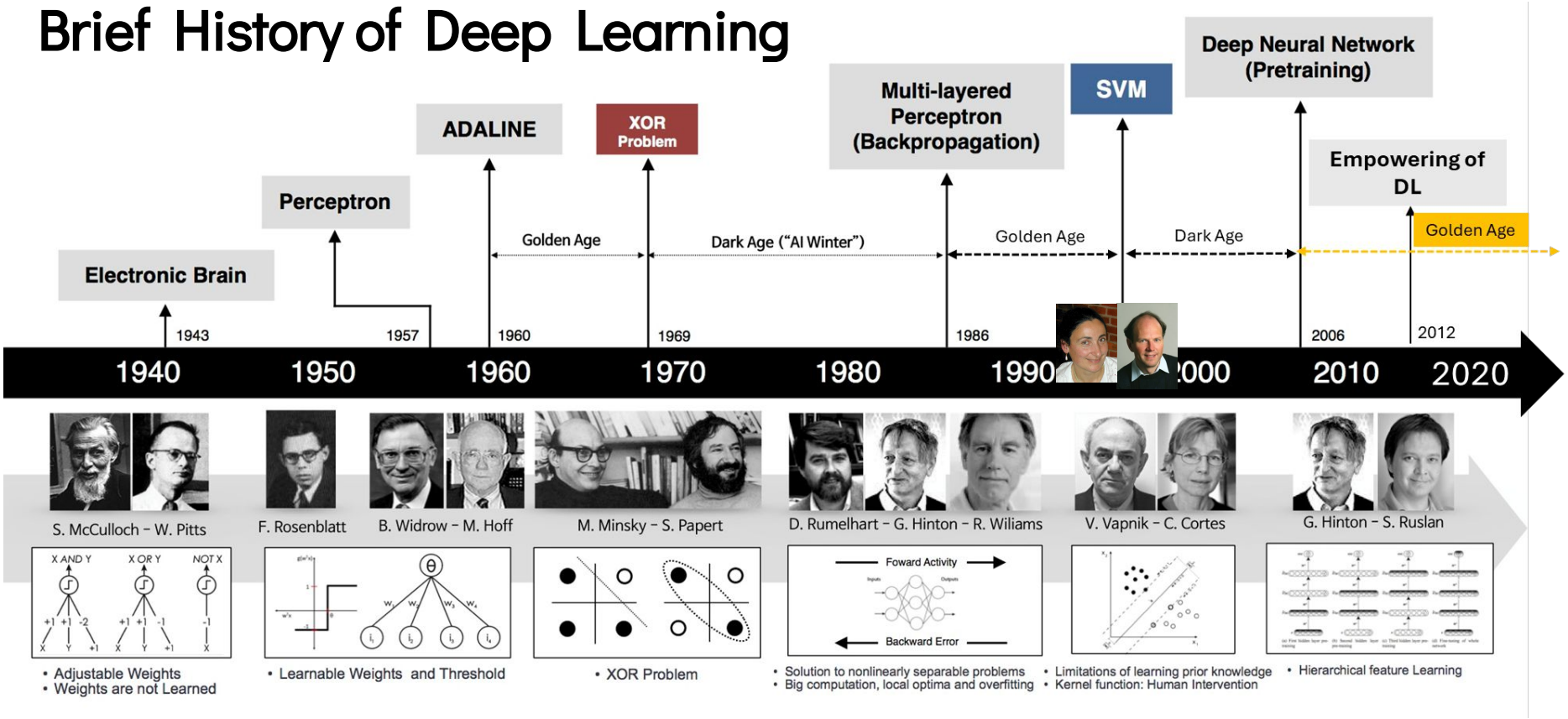
Slide from Robet Lange, AutoML seminars, https://www.youtube.com/watch?v=CFewReJ1Eyo&t=2279s

Deep Learning

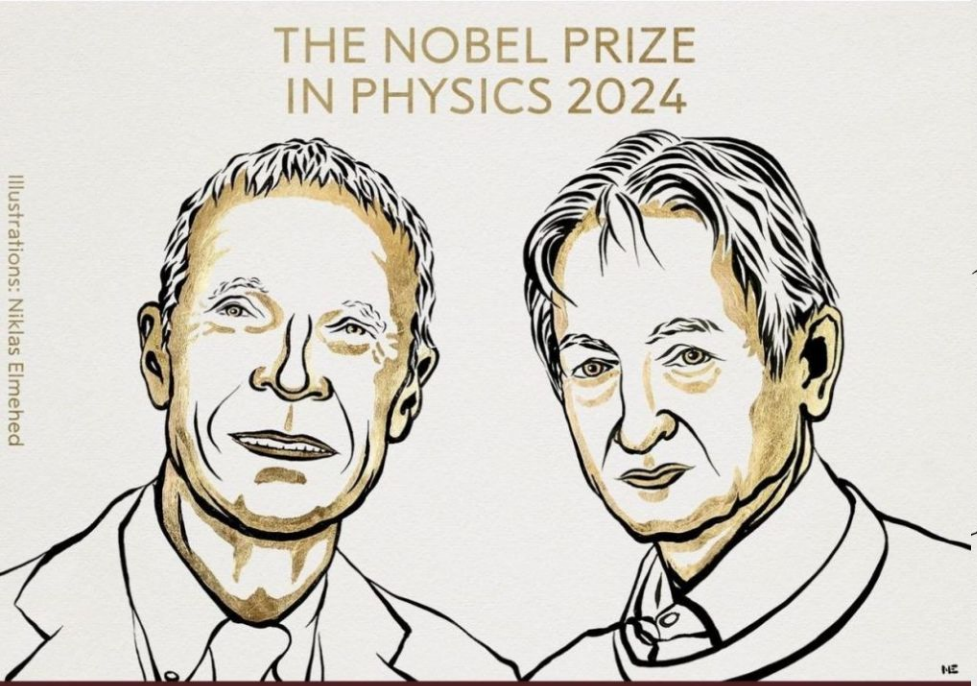
- Methodology of ML approaching problems through models organized in layers of parameters
- Layers capture descriptive & discriminative information from raw data directly
- Can be used for supervised / unsupervised learning, reinforcement learning, feature extraction, probabilistic modeling, etc.
- Some DL models are: *MLPs, DNNs, CNNs, Transformers, GANs, DBNs, AEs, etc.*

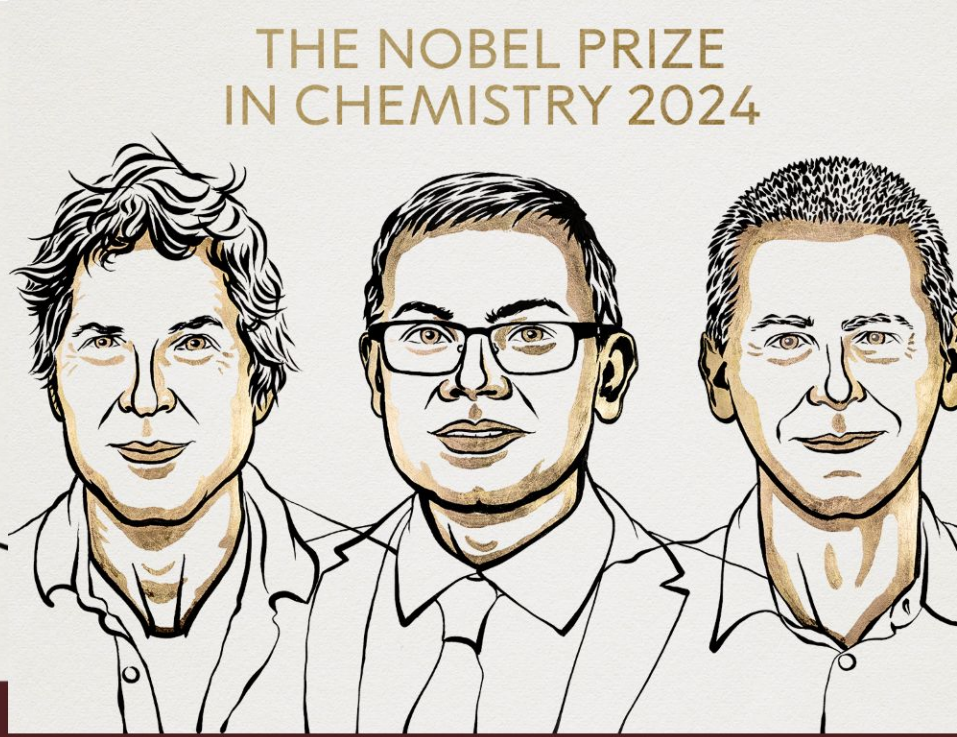


Brief History of Deep Learning



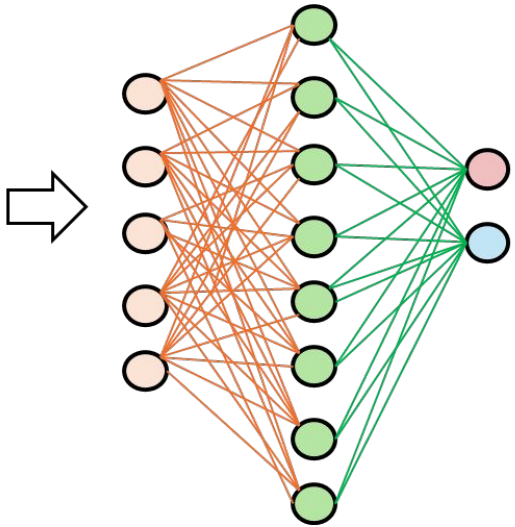
http://beamandrew.github.io/deeplearning/2017/02/23/deep_learning_101_part1.html
http://beamlab.org/deeplearning/2017/02/23/deep_learning_101_part1.html





Recent history (2024)

Multi-layer Perceptron



Outputs

Variables - U

5x8 double

	1	2	3	4	5	6	7	8	9
1	-0.7203	1.1714	-1.5029	3.3133	0.0338	-0.3660	-0.2886	-1.0836	
2	-0.5102	0.2660	0.3880	0.8725	-0.5068	-1.6706	1.5959	1.0228	
3	-1.6597	-0.5933	-0.4178	1.8309	1.0331	0.1132	-0.3820	1.0903	
4	0.3005	0.3909	-0.2304	-2.3196	3.2656	-0.0441	0.4254	0.0461	
5	0.2737	-0.6014	0.0314	0.5855	0.3692	2.1029	1.5813	-0.9019	
6									
7									
8									
9									
10									
11									

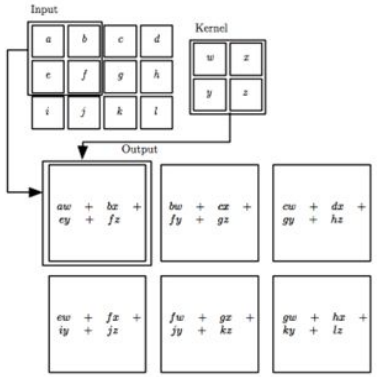
Variables - W

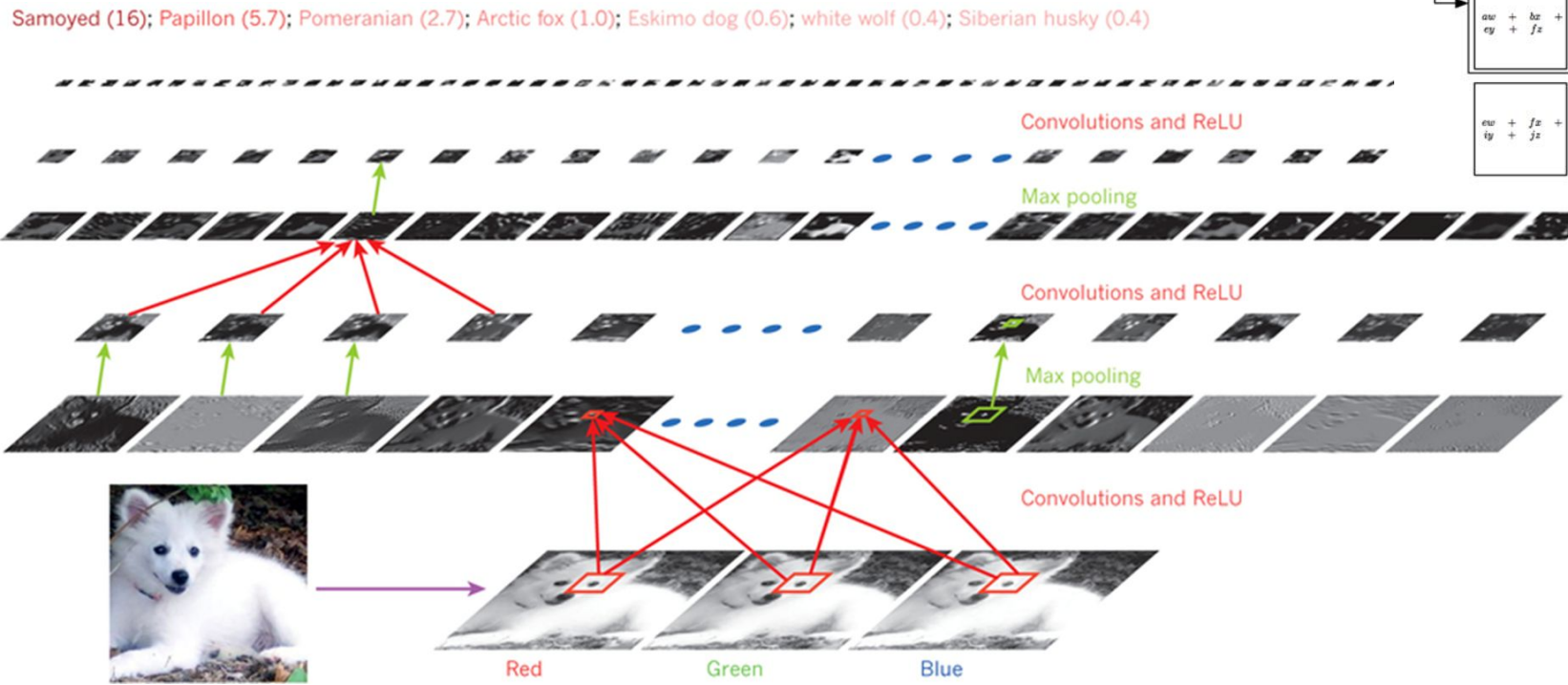
8x2 double

	1	2	3
1	0.5884	0.1427	
2	0.9775	0.6085	
3	0.2137	0.6286	
4	0.6291	0.0238	
5	0.7810	0.7787	
6	0.3571	0.4912	
7	0.9407	0.4116	
8	0.9616	0.7775	
9			
10			
11			
12			
13			

$$f(\mathbf{x}) = s(\mathbf{w}\phi(\mathbf{x}) + b)$$

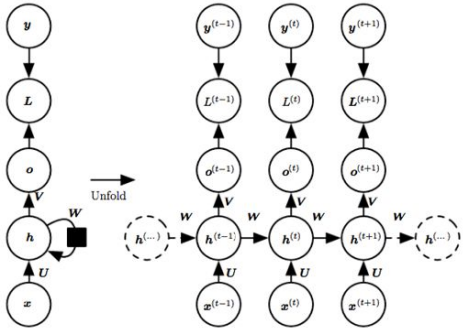
Convolutional Neural Networks

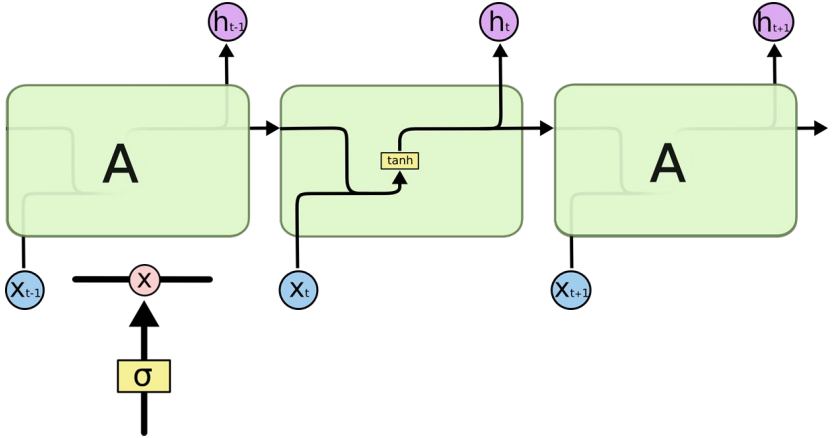


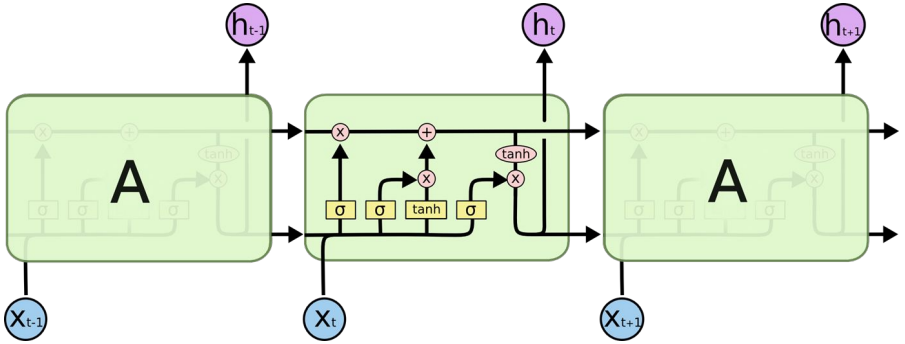


Sequential Models

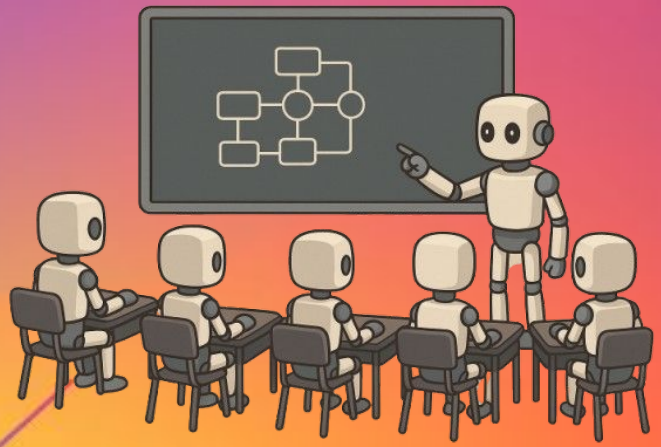
RNNs, LSTMs, and GRUs







http://colah.github.io/posts/2015-08-Understanding-LSTMs/



Content

- Attention mechanism
- Transformer architecture
- Encoders and decoders

Attention and Transformers

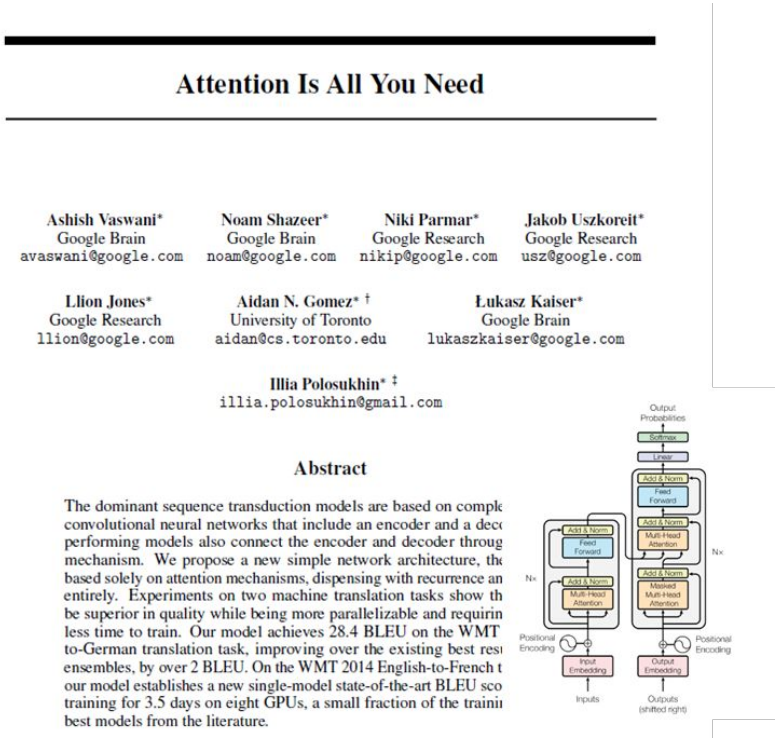
MICAI 2025

INAOE

Machine Learning and Artificial Intelligence

…. If modern artificial intelligence has a founding document, a sacred text, it is Google's 2017 research paper "Attention Is All You Need."…

Rob Toews, Forbes, 2023



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

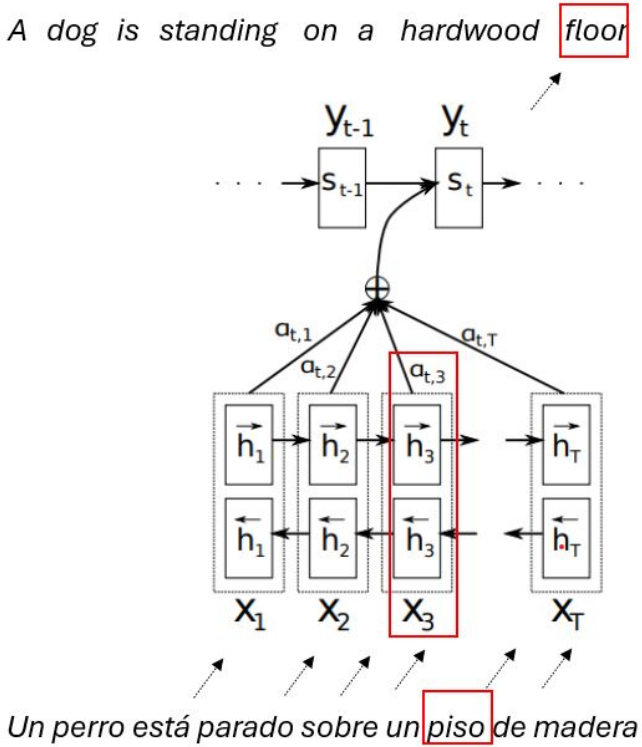
The dominant sequence transduction models are based on comple convolutional neural networks that include an encoder and a dec performing models also connect the encoder and decoder throug mechanism. We propose a new simple network architecture, th based solely on attention mechanisms, dispensing with recurrence an entirely. Experiments on two machine translation tasks show th be superior in quality while being more parallelizable and requirin less time to train. Our model achieves 28.4 BLEU on the WMT to-German translation task, improving over the existing best resu ensembles, by over 2 BLEU. On the WMT 2014 English-to-French t our model establishes a new single-model state-of-the-art BLEU sco training for 3.5 days on eight GPUs, a small fraction of the trainin best models from the literature.

https://www.forbes.com/sites/robtoews/2023/09/03/transformers-revolutionized-ai-what-will-replace-them/

Attention Mechanisms

- Neural machine translation:
  - A weight for the decoder considering, inputs, hidden states and a (learnable) ***context vector*** derived from hidden states

- The context aligns input and output sequences, indicating what input words are more likely to be related to each output word



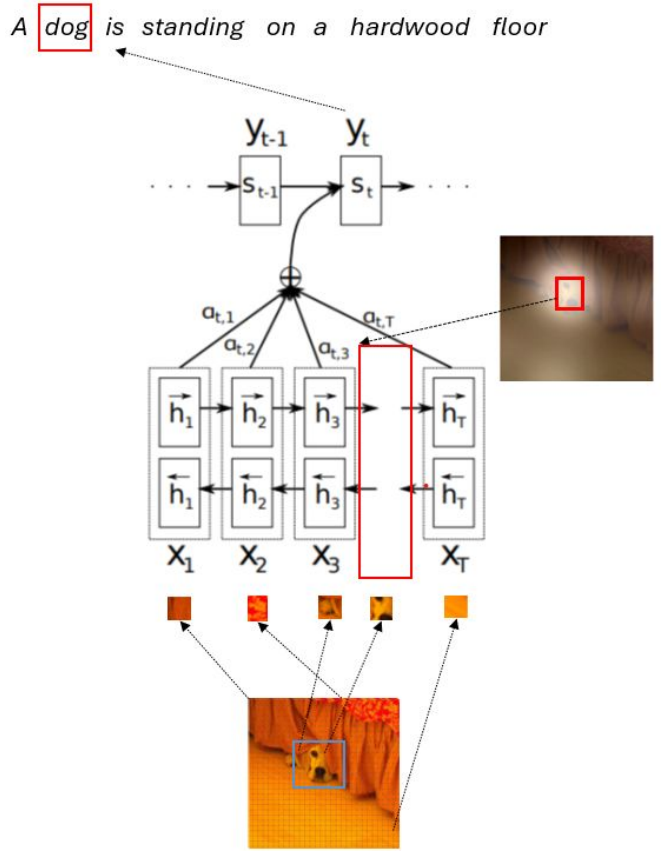
$$c_i = \sum_{j=1}^{T_x} \alpha_{ij} h_j. \qquad p(y_i|y_1, \ldots, y_{i-1}, \mathbf{x}) = g(y_{i-1}, s_i, c_i),$$

K. Xu, et. al. Proc. of 32nd ICML, PMLR 37:2048-2057, 2015.

Attention Mechanisms

- Neural machine translation:
  - A weight for the decoder considering, inputs, hidden states and a (learnable) ***context vector*** derived from hidden states

- The context aligns input and output sequences, indicating what input words are more likely to be related to each output word



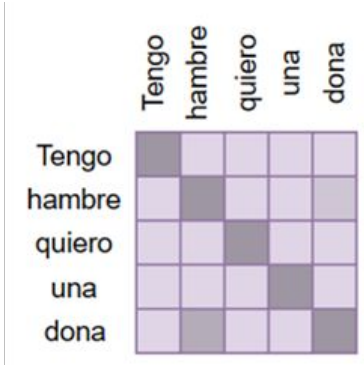
$$c_i = \sum_{j=1}^{T_x} \alpha_{ij} h_j. \qquad p(y_i|y_1, \ldots, y_{i-1}, \mathbf{x}) = g(y_{i-1}, s_i, c_i),$$

K. Xu, et. al. Proc. of 32nd ICML, PMLR 37:2048-2057, 2015.

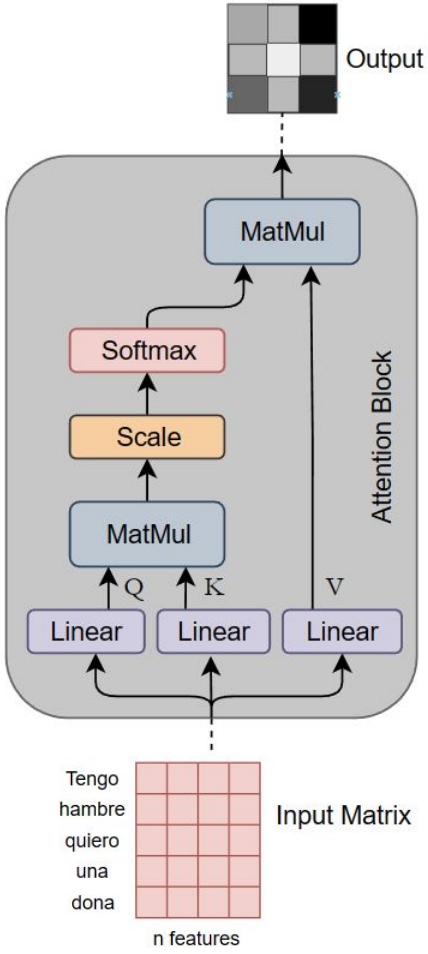
Transformers

Self attention: how relevant are the *words* in a sentence to each other?

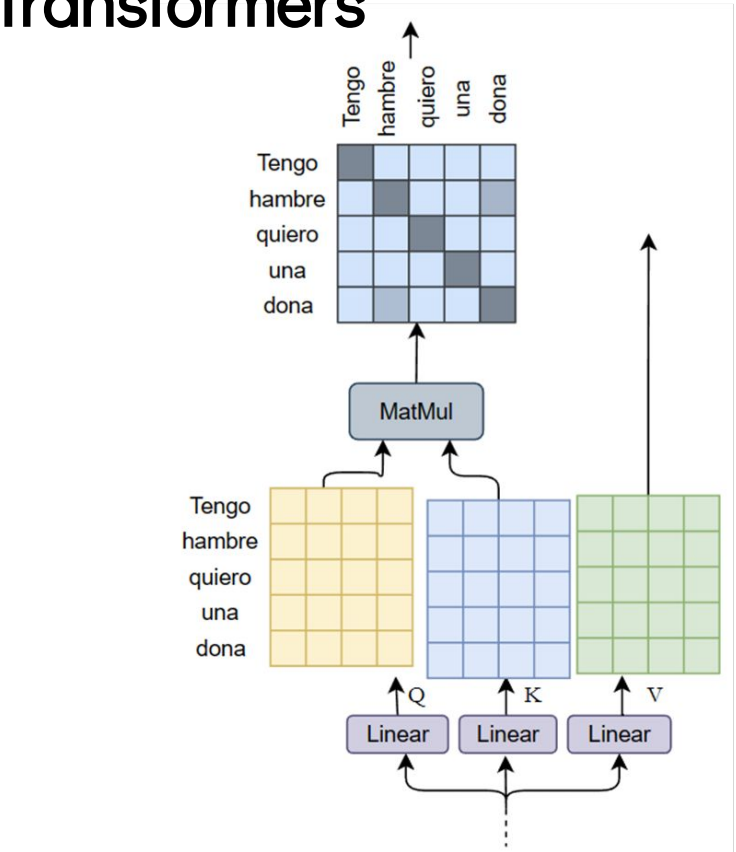
“Tengo hambre, quiero una dona”

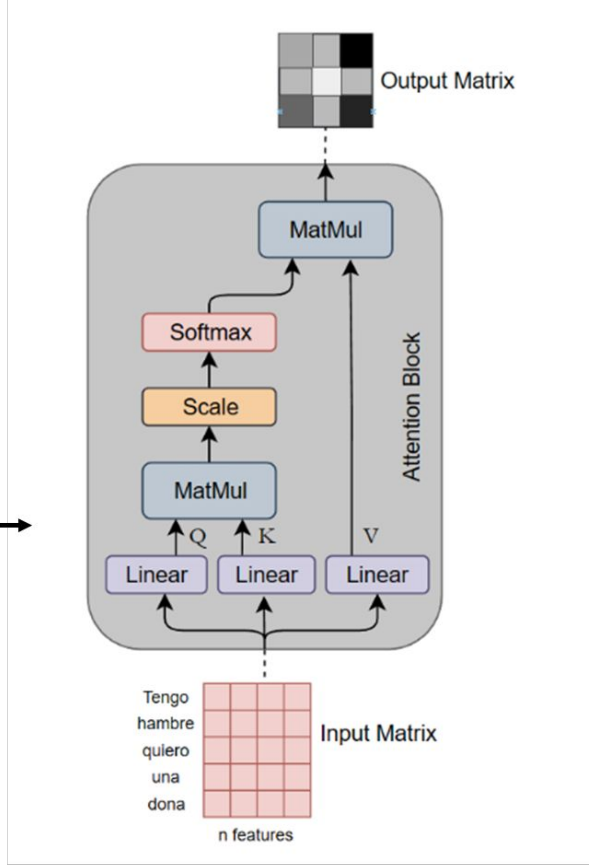


$$\text{Attention}(Q, K, V) = \text{softmax}(\frac{QK^T}{\sqrt{d_k}})V$$

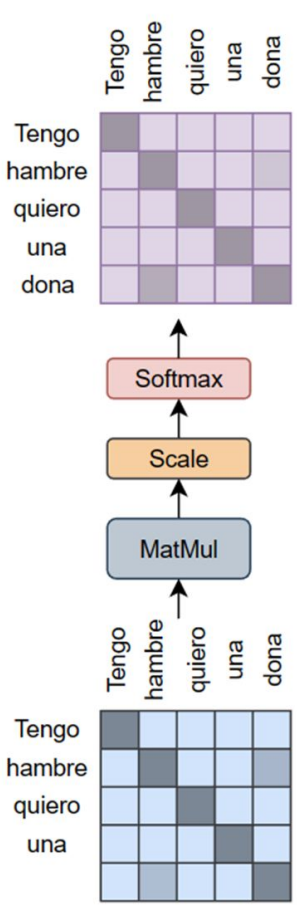


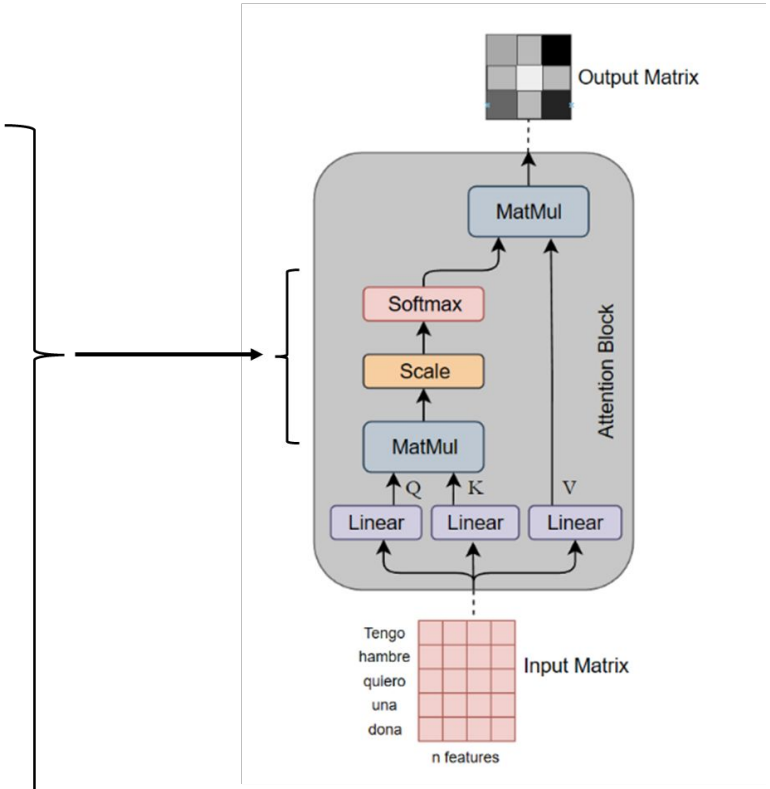
Transformers



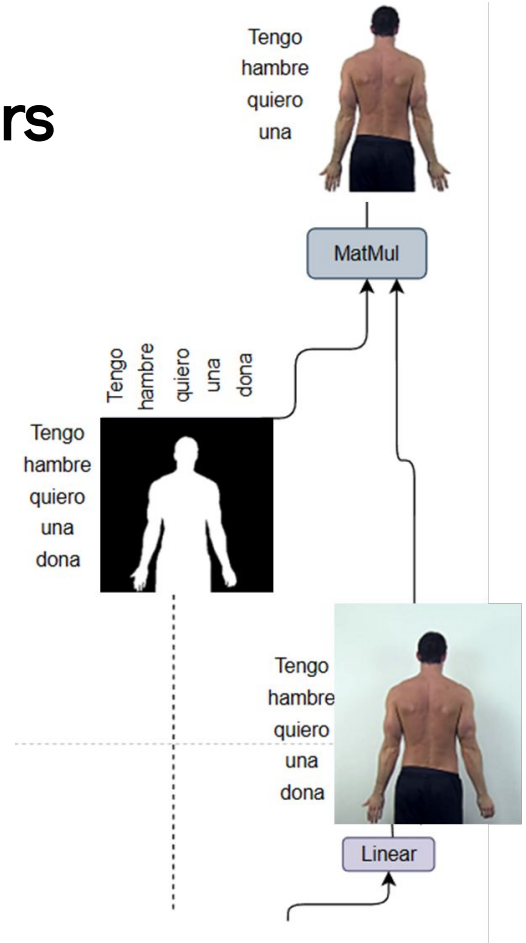


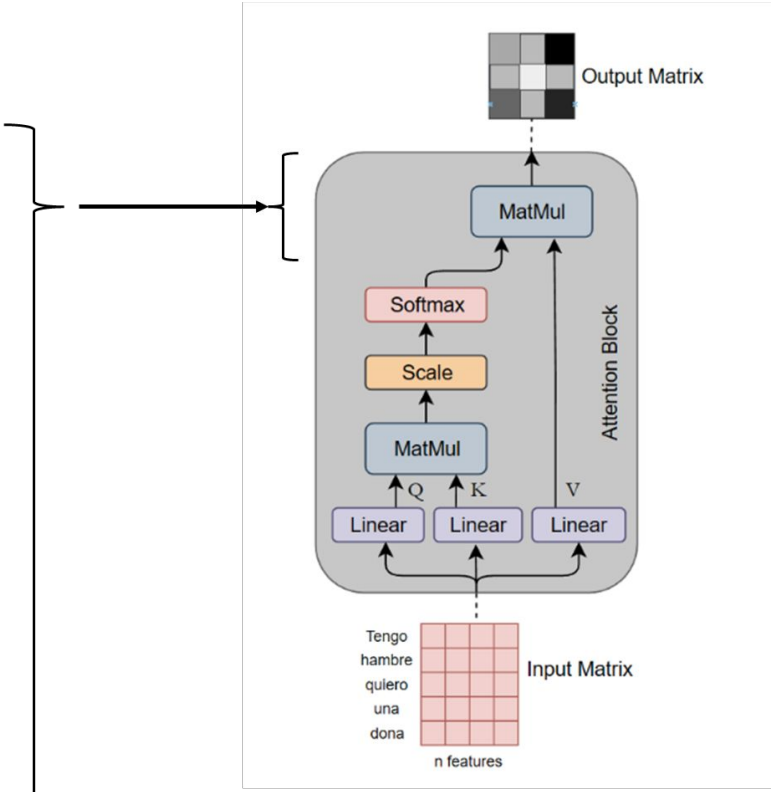
Transformers





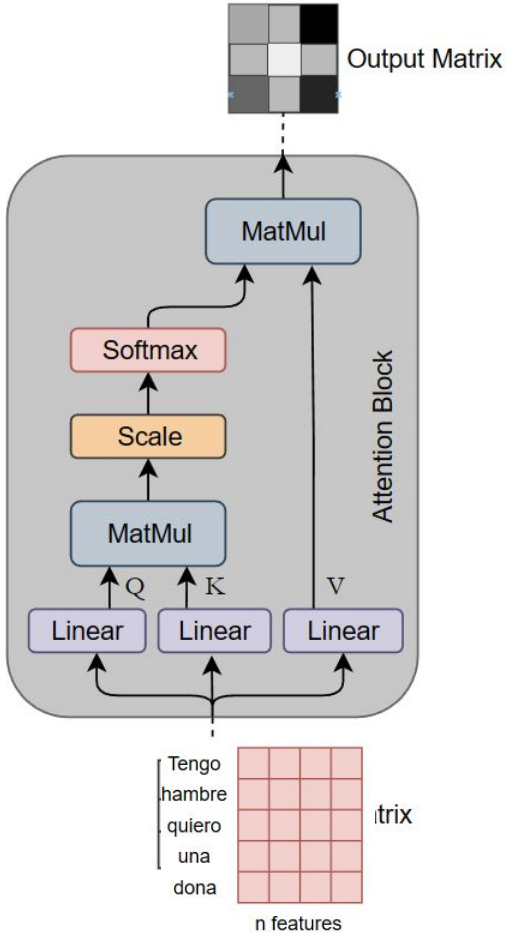
Transformers



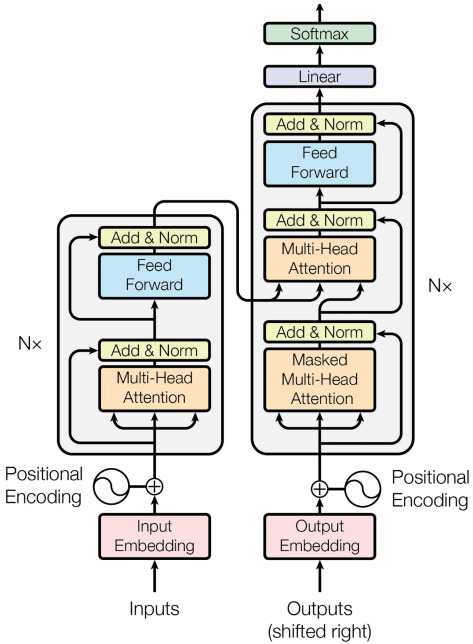


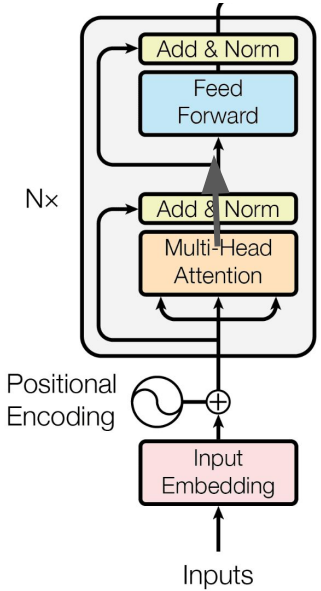
Practice - The Attention Mechanism

GitHub



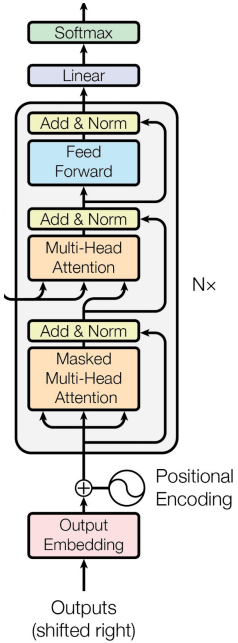
Architectures





Encoder

- *Positional encoding*
- *Usual output: embeddings*
- *MHA, AN+FF*

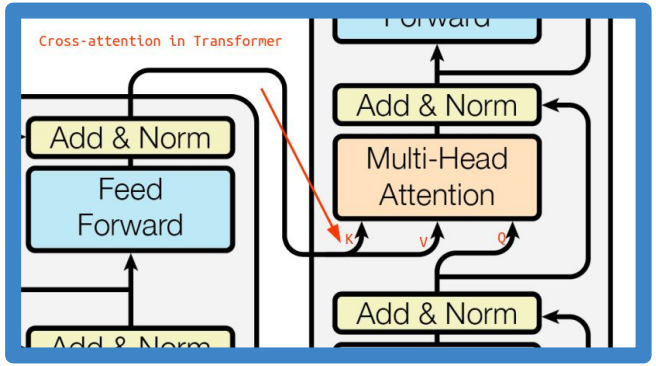


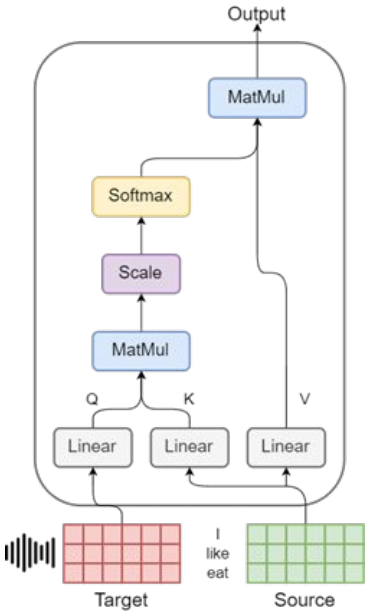
Decoder

- *Autoregressive*
- *Usual output: token probability*
- *MHA, AN+FF, PE*

Transformers

Cross attention: how relevant are the *words* in a sentence to words from another sentence (*K* and *V* from a sentence, *Q* from the other one)



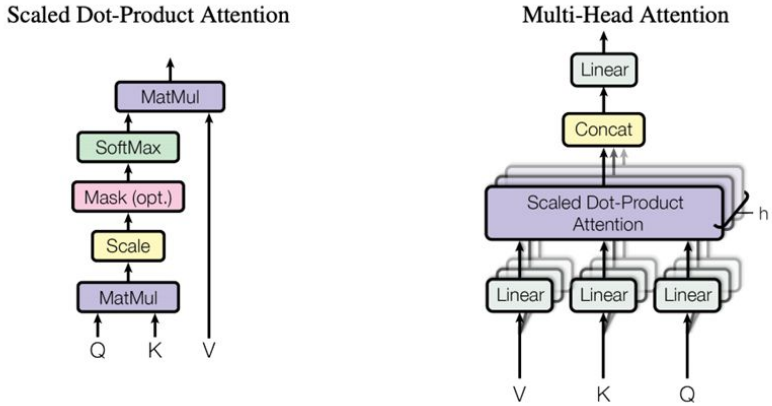


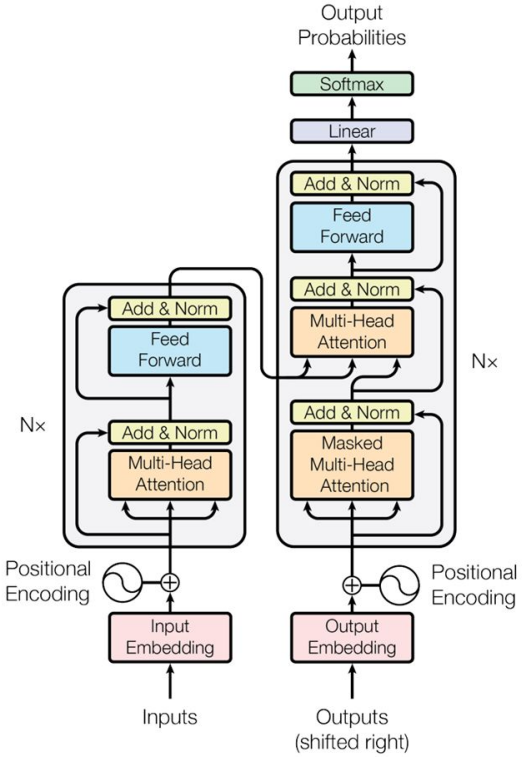
Ashish Vaswani et al. Attention is all you need. NeurIPS 2017

http://nlp.seas.harvard.edu/2018/04/03/attention.html
https://www.youtube.com/watch?v=8wUVJRm8DX0

Transformers

Multi-head attention: parallel attention layers





Ashish Vaswani et al. Attention is all you need. NeurIPS 2017

http://nlp.seas.harvard.edu/2018/04/03/attention.html

https://www.youtube.com/watch?v=8wUVJRm8DX0

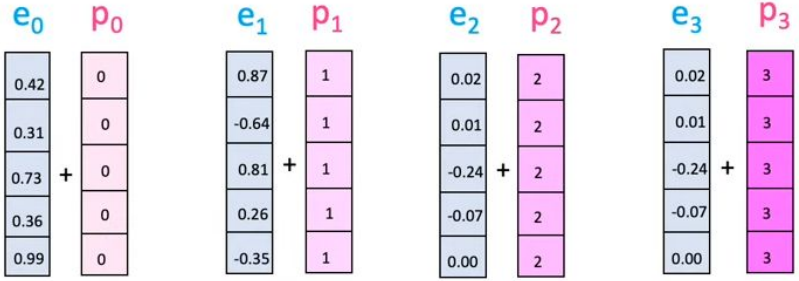
https://tinyurl.com/3vnz3kmu

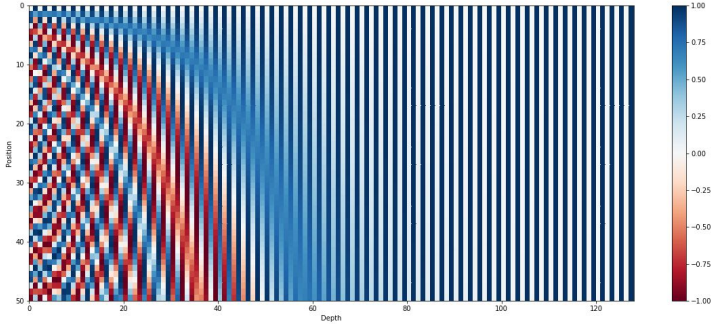
Transformers

Tokenization: a way to split sequential data into primitives (tokens)

Positioning encoding: The only explicit mechanism to account for sequential information

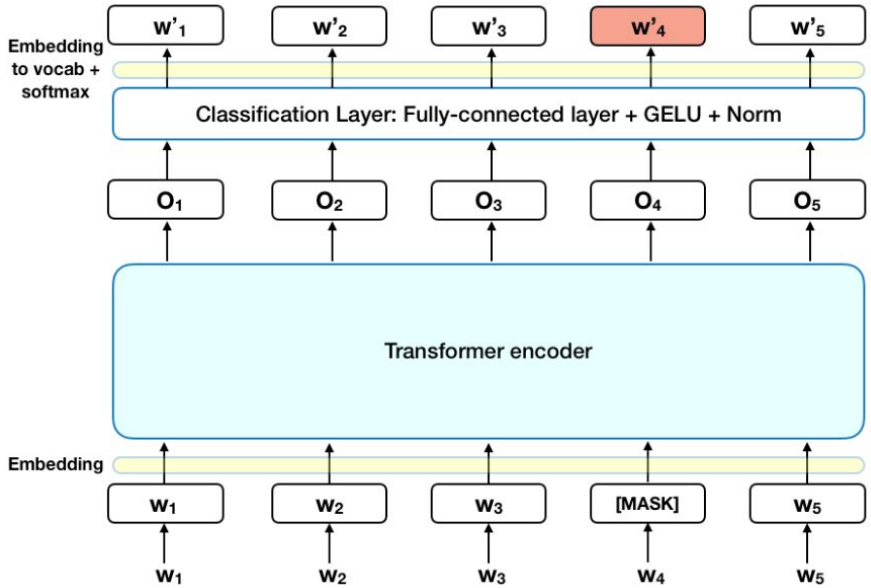
ORIGINAL TOKENIZATION	WORDPIECES
Leicestershire	Leicester
	##shire
beat	beat
Somerset	Somerset
by	by
an	an
innings	innings





Popular transformers: BERT

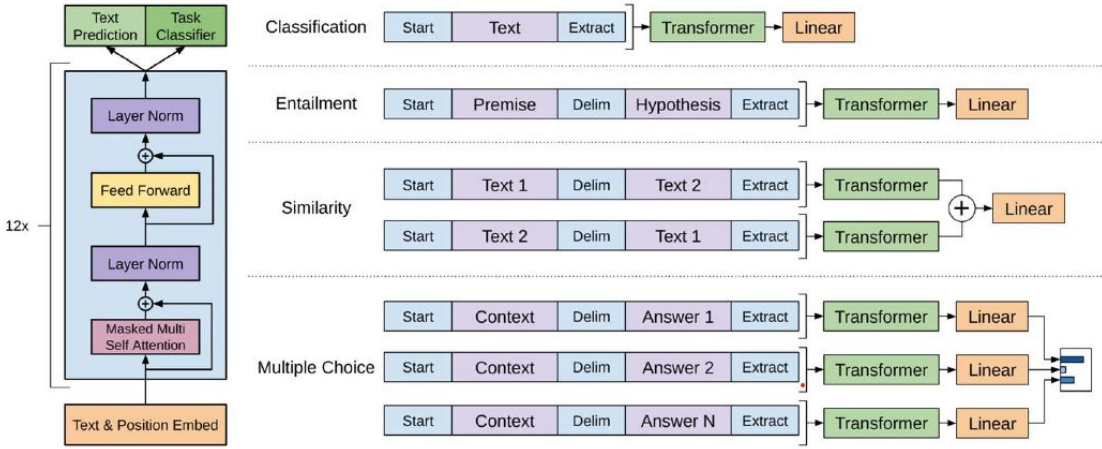
- Bidirectional Encoder Representations from Transformers
  - Large scale pretraining
  - Pretext tasks: MLM & NSP
  - Bidirectional encoder model
  - Special tokens



Popular Transformers: GPT

- Generative pretraining transformer
  - Large scale pretraining
  - Autoregressive decoder language model (left to right)
  - Fine tuning for solving downstream tasks for the task at hand

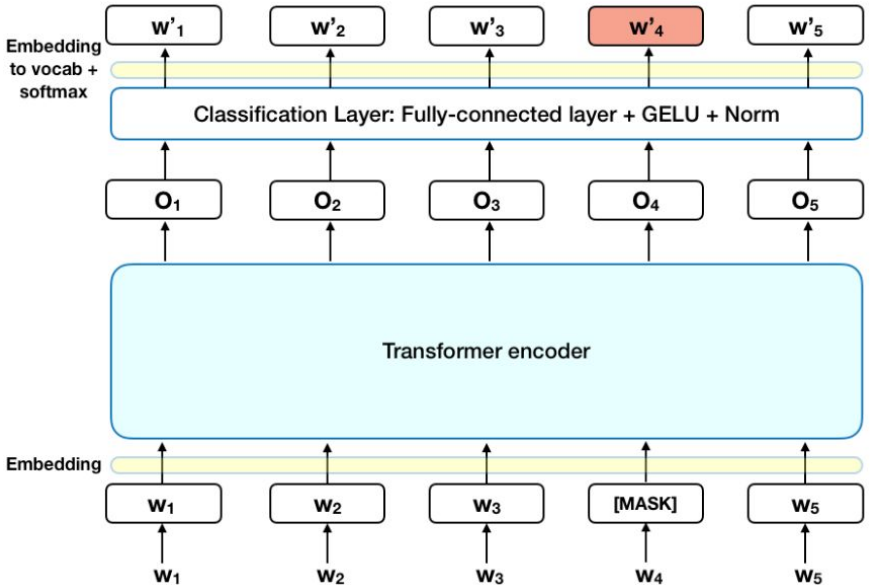
$$L_1(\mathcal{U}) = \sum \log P(u_i | u_{i-k}, \ldots, u_{i-1}; \Theta)$$

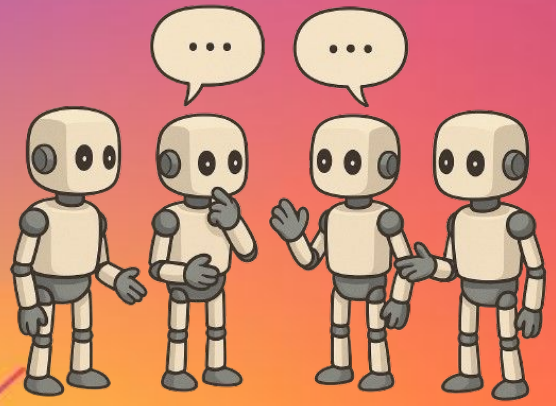


A. Radford et al. Improving Language Understanding by Generative Pre-Training. 2018 https://cdn.openai.com/research-covers/language-unsupervised/language_understanding_paper.pdf

Practice - Using BERT to encode a sentence

GitHub





Content

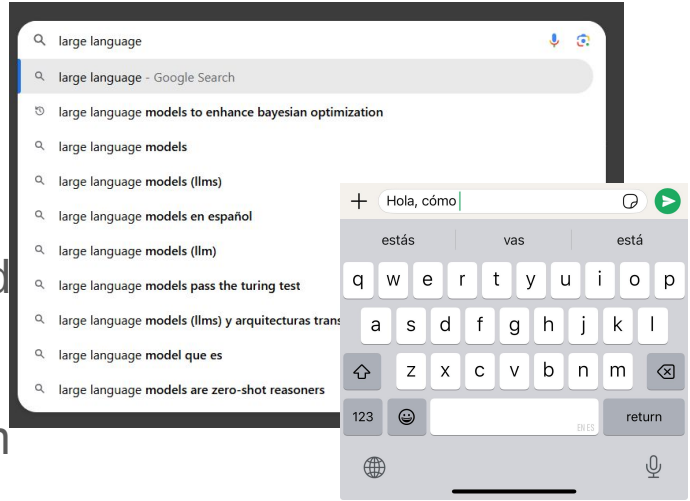
- Language model definition
- Architecture visualization
- Quantization
- Prompting techniques

MICAI 2025

INAOE

Language Models

Language Models



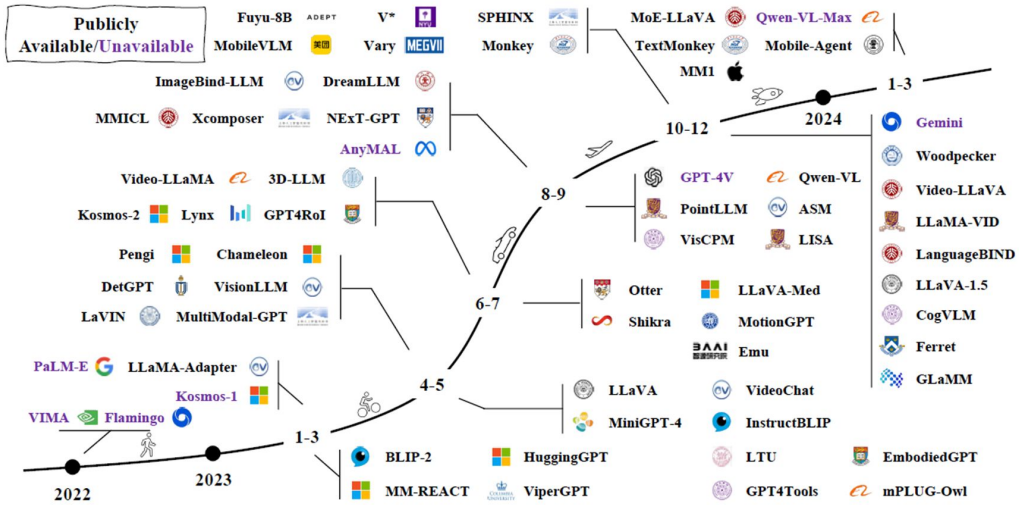
- A language model learns to predict the next word in a sentence
  - It aims to "*understand*" and generate language according to patterns learned from a large corpus of documents

- Maximize: $L_1(\mathcal{U}) = \sum \log P(u_i | u_{i-k}, \ldots, u_{i-1}; \Theta)$

- LLMs are LMs trained on massive amounts of data

Large Language Models

- How many LLMs are out there?
- Are LLMs restricted to text?
- Which one is better?
- What are the criteria used to evaluate LLMs?



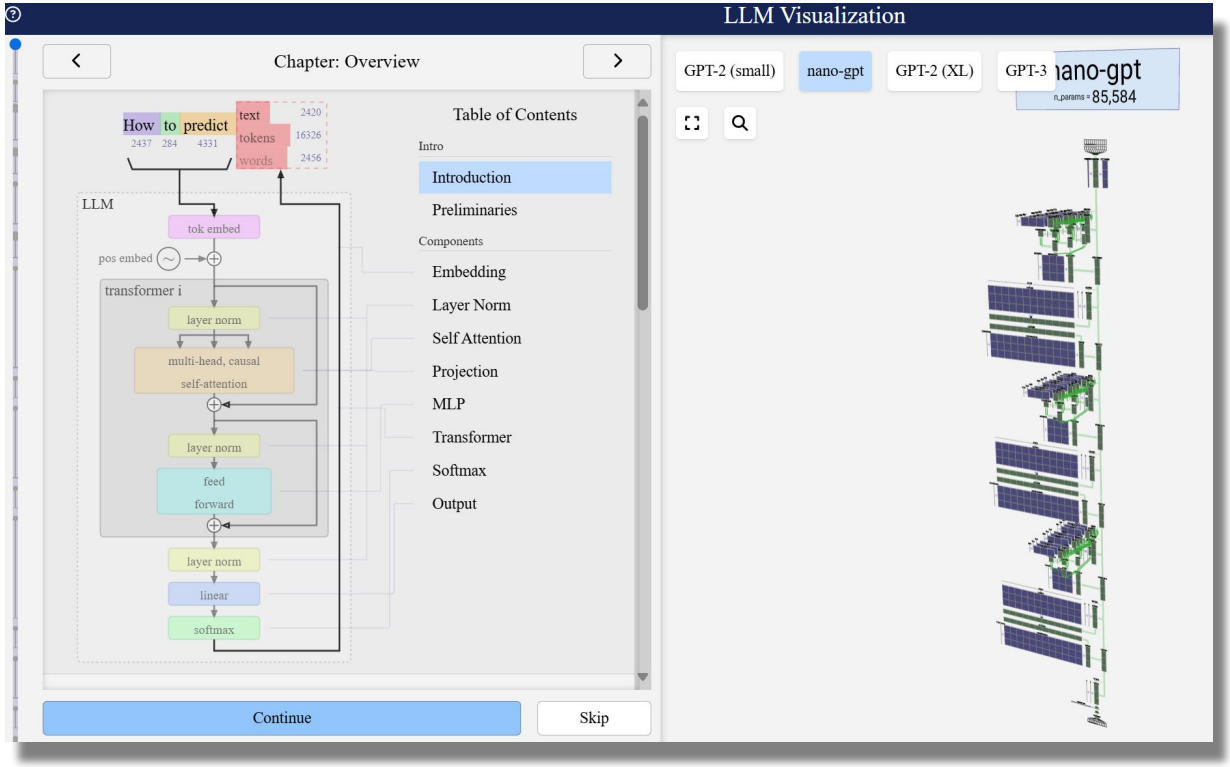
Text

View

Rank (UB)	Model	Score	Votes
1	gemini-2.5-pro	1451	54,087
1	claude-opus-4-1-20250805-thi…	1447	21,306
1	claude-sonnet-4-5-20250929-t…	1445	6,287
1	gpt-4.5-preview-2025-02-27	1441	14,644
2	chatgpt-4o-latest-20250326	1440	40,013
2	o3-2025-04-16	1440	51,293
2	claude-sonnet-4-5-20250929	1438	6,144
2	gpt-5-high	1437	23,580
2	claude-opus-4-1-20250805	1437	33,298
3	qwen3-max-preview	1434	18,078

https://en.wikipedia.org/wiki/List_of_large_language_models
https://www.vellum.ai/llm-leaderboard?utm_source=google&utm_medium=organic
https://huggingface.co/spaces/lmarena-ai/lmarena-leaderboard
S. Yin et al. A Survey on Multimodal Large Language Models. TPAMI, 2025,
https://github.com/BradyFU/Awesome-Multimodal-Large-Language-Models

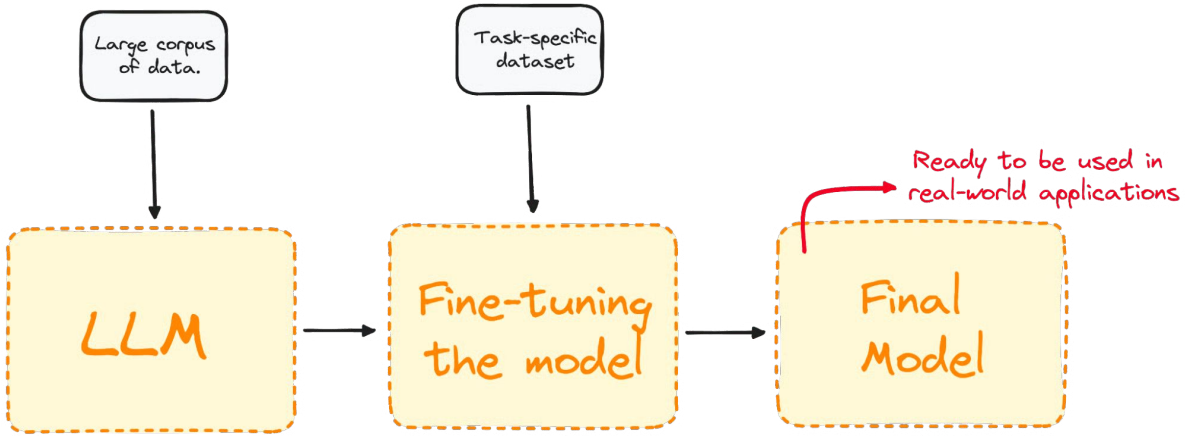
Visualization of LLMs



Brendan Bycroft: https://bbycroft.net/llm

LLM Fine tuning

- Given a (generic) trained LLM, adjust the model for a specific domain by running additional learning rounds over a task-specific dataset

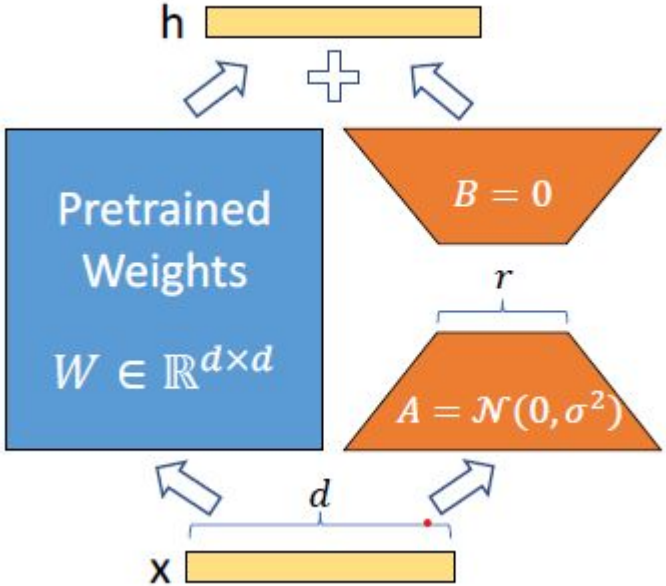


https://www.datacamp.com/es/tutorial/fine-tuning-large-language-models

Fine tuning with LoRA

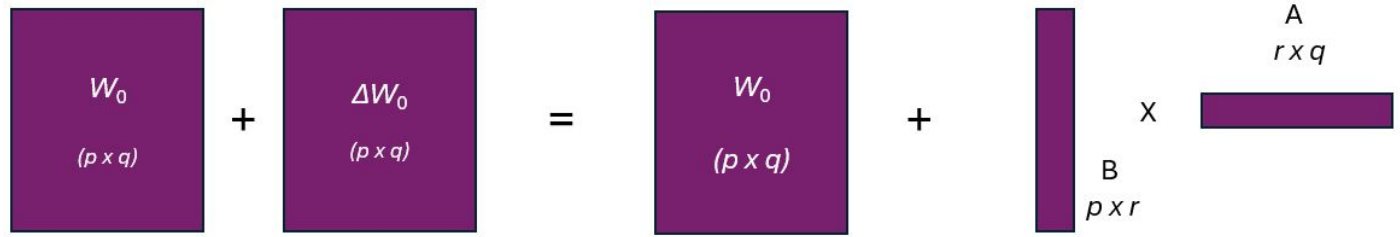
- Efficient fine tuning by freezing the initial LLM and learning (small) decomposition matrix

- These matrices learn the changes in weights as tuning occurs



Fine tuning with LoRA

$$W_0 + \Delta W = W_0 + BA$$

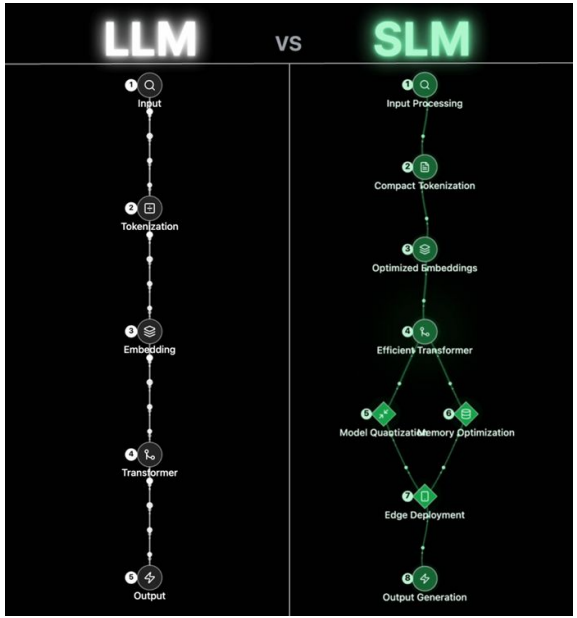


E. Hu et al. LORA: oRA: Low-Rank Adaptation of Large Language Models. https://arxiv.org/abs/2106.09685

Small Language Models

"... Small Language Models (SLMs) are lightweight versions of traditional language models designed to operate efficiently on resource-constrained environments such as smartphones, embedded systems, or low-power computers… "

Dimension	LLMs	SLMs
Parameters	Billions	Millions– few B
Speed	Slower	Faster
Cost	High	Low
Deployment	Cloud	Edge / Local
Examples	GPT-5, Claude 3 DeepSeek-V3.2 Llama 4, Gemini 2.5	Phi-3.5-Mini-3.8B SmolLM2-1.7B DeepSeeek-R1-1.5B



Credit: Manthan Patel

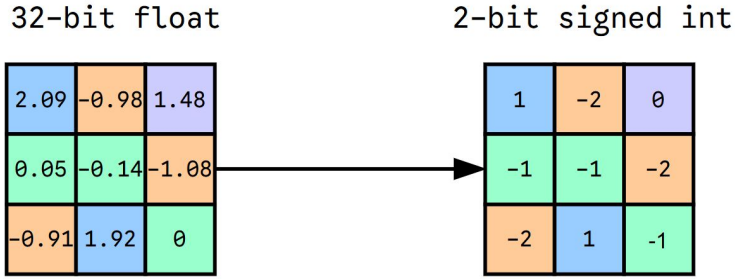
https://www.linkedin.com/posts/leadgenmanthan_llms-are-overkill-for-80-of-business-tasks-activity-7352921698897166336-H5QK/
https://huggingface.co/blog/jjokah/small-language-model

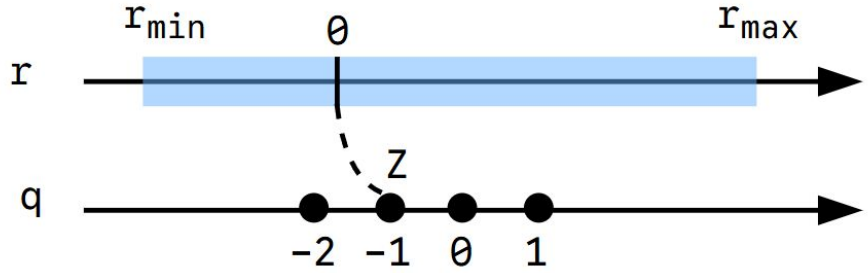
Model Quantization

How to go from a Large Language Model to a ***Small Language Model***?

- Asymmetric Linear Quantization

Represent real values with a finite number of states





Must approximate from the encoded states to the original value…

$$S = \frac{(r_{max} - r_{min})}{(q_{max} - q_{min})}$$

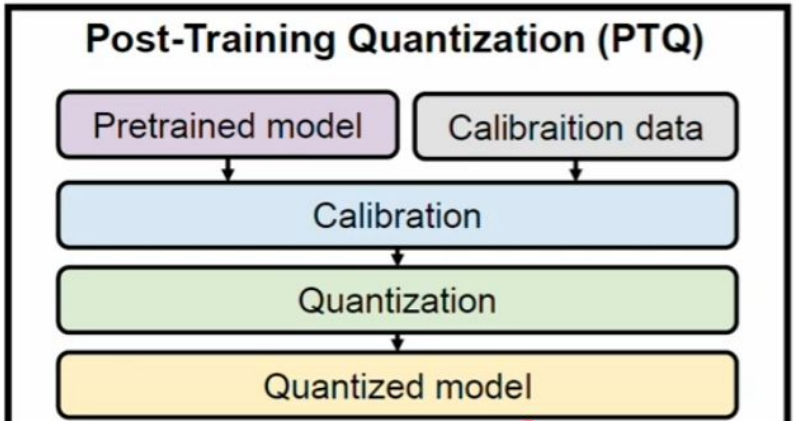
$$r = (q - Z) \times S$$

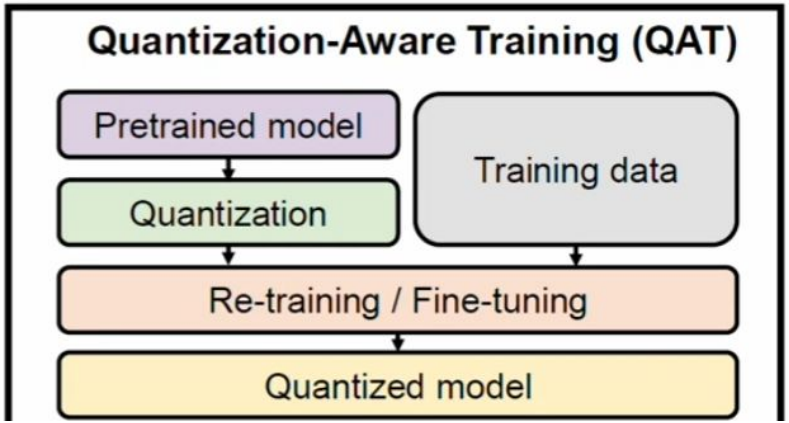
Courtesy of part of slides: MIT 6.5940 TinyML and Efficient Deep Learning Computing by Prof. H. Song

Model Quantization

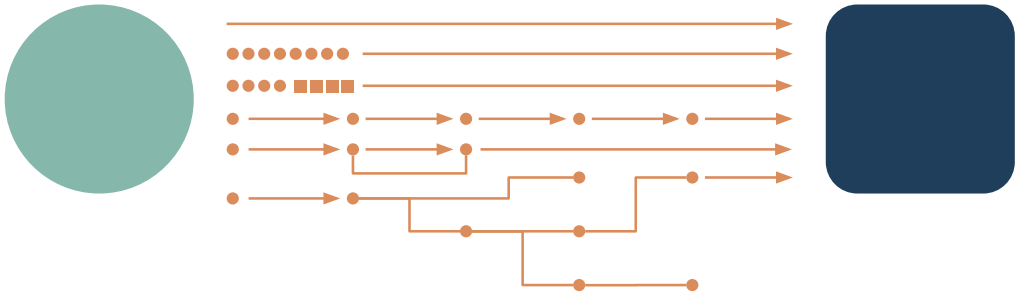
There are two common quantization techniques

- Two types of DNN quantization: QAT and PTQ
  - Post-Training Quantization (**PTQ**) directly converts parameters and activations to reduced-precision
  - Quantization-Aware Training (**QAT**) exploits re-training to compensate quantization error





Prompting Techniques



- Zero-Shot
- Few-Shot
- Chain-of-Thought Prompting
- Self-Consistency
- Generated Knowledge
- ToT
- RAG

Zero-Shot Prompting

PROMPT:

Classify the text into neutral, negative or positive.

Text: I think the vacation is okay.
Sentiment:

OUTPUT:

Zero-Shot Prompting

PROMPT:

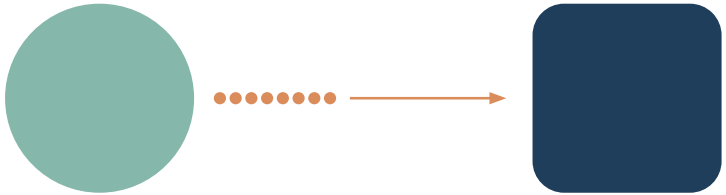
Classify the text into neutral, negative or positive.

Text: I think the vacation is okay.
Sentiment:

OUTPUT:

neutral

Few-Shot Prompting



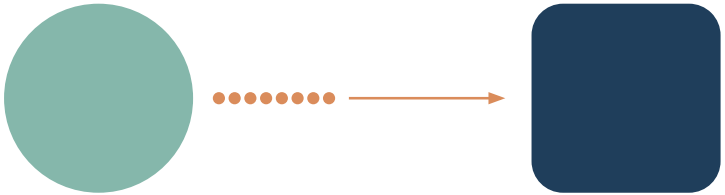
PROMPT:

A "whatpu" is a small, furry animal native to Tanzania. An example of a sentence that uses the word whatpu is:
We were traveling in Africa and we saw these very cute whatpus.

To do a "farduddle" means to jump up and down really fast. An example of a sentence that uses the word farduddle is:

OUTPUT:

Few-Shot Prompting



PROMPT:

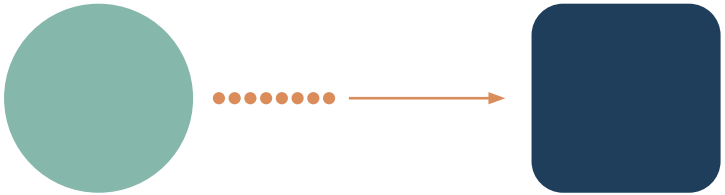
A "whatpu" is a small, furry animal native to Tanzania. An example of a sentence that uses the word whatpu is:
We were traveling in Africa and we saw these very cute whatpus.

To do a "farduddle" means to jump up and down really fast. An example of a sentence that uses the word farduddle is:

OUTPUT:

When we won the game, we all started to farduddle in celebration.

Few-Shot Prompting (2)



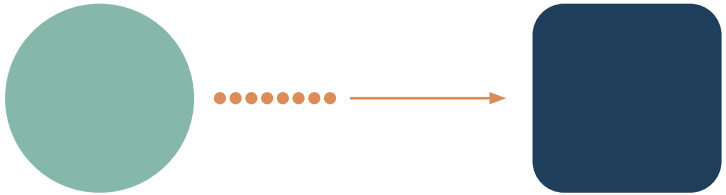
PROMPT:

This is awesome! // Negative
This is bad! // Positive
Wow that movie was rad! // Positive
What a horrible show! //

OUTPUT:

Negative

Few-Shot Prompting (Limitations)



PROMPT:

The odd numbers in this group add up to an even number: 15, 32, 5, 13, 82, 7, 1.
A:

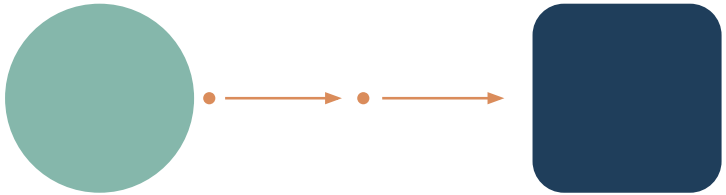
OUTPUT:

Yes, the odd numbers in this group add up to 107, which is an even number.

OUTPUT:

False

Chain-of-Thought Prompting



PROMPT:

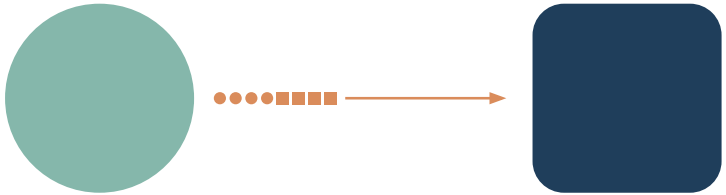
The odd numbers in this group add up to an even number: 4, 8, 9, 15, 12, 2, 1.
A: Adding all the odd numbers (9, 15, 1) gives 25. The answer is False.

The odd numbers in this group add up to an even number: 15, 32, 5, 13, 82, 7, 1.
A:

OUTPUT:

Adding all the odd numbers (15, 5, 13, 7, 1) gives 41. The answer is False.

Self-Consistency Prompting



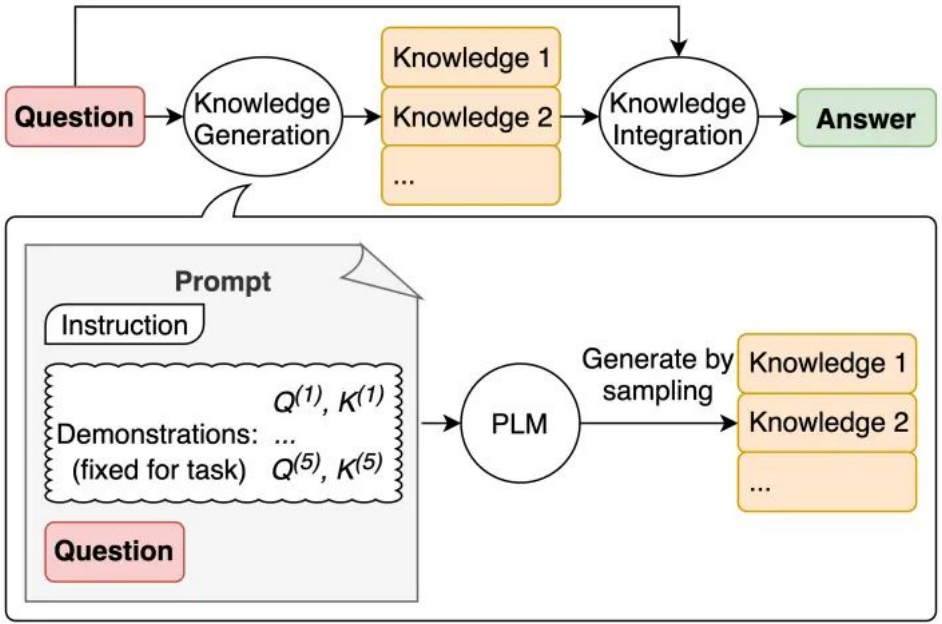
PROMPT:

Q: Michael had 58 golf balls. On tuesday, he lost 23 golf balls. On wednesday, he lost 2 more. How many golf balls did he have at the end of wednesday?
A: Michael initially had 58 balls. He lost 23 on Tuesday, so after that he has 58 - 23 = 35 balls. On Wednesday he lost 2 more so now he has 35 - 2 = 33 balls. The answer is 33.
Q: Olivia has $23. She bought five bagels for $3 each. How much money does she have left?
A: She bought 5 bagels for $3 each. This means she spent $15. She has $8 left.
Q: When I was 6 my sister was half my age. Now I'm 70 how old is my sister?
A:

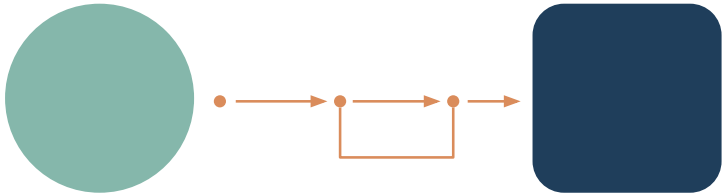
OUTPUT:

When I was 6 my sister was half my age, so she was 3. Now I am 70, so she is 70 - 3 = 67. The answer is 67.

Generated Knowledge Prompting



Generated Knowledge Prompting

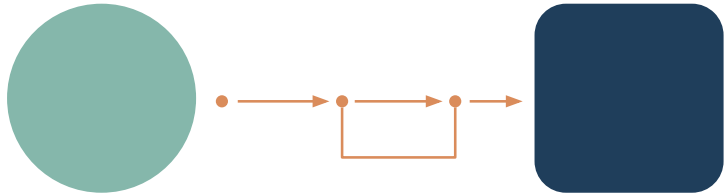


PROMPT:

Part of golf is trying to get a higher point total than others. Yes or No?

OUTPUT:

Generated Knowledge Prompting



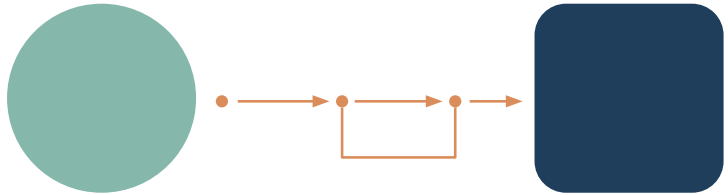
PROMPT:

Input: A rock is the same size as a pebble.
Knowledge: A pebble is a clast of rock with a particle size of 4 to 64 millimetres based on the Udden-Wentworth scale of sedimentology. Pebbles are generally considered larger than granules (2 to 4 millimetres diameter) and smaller than cobbles (64 to 256 millimetres diameter).
Input: Part of golf is trying to get a higher point total than others.
Knowledge:

OUTPUT:

The objective of golf is to play a set of holes in the least number of strokes. A round of golf typically consists of 18 holes. Each hole is played once in the round on a standard golf course. Each stroke is counted as one point, and the total number of strokes is used to determine the winner of the game.

Generated Knowledge Prompting



PROMPT:

Part of golf is trying to get a higher point total than others. Yes or No?

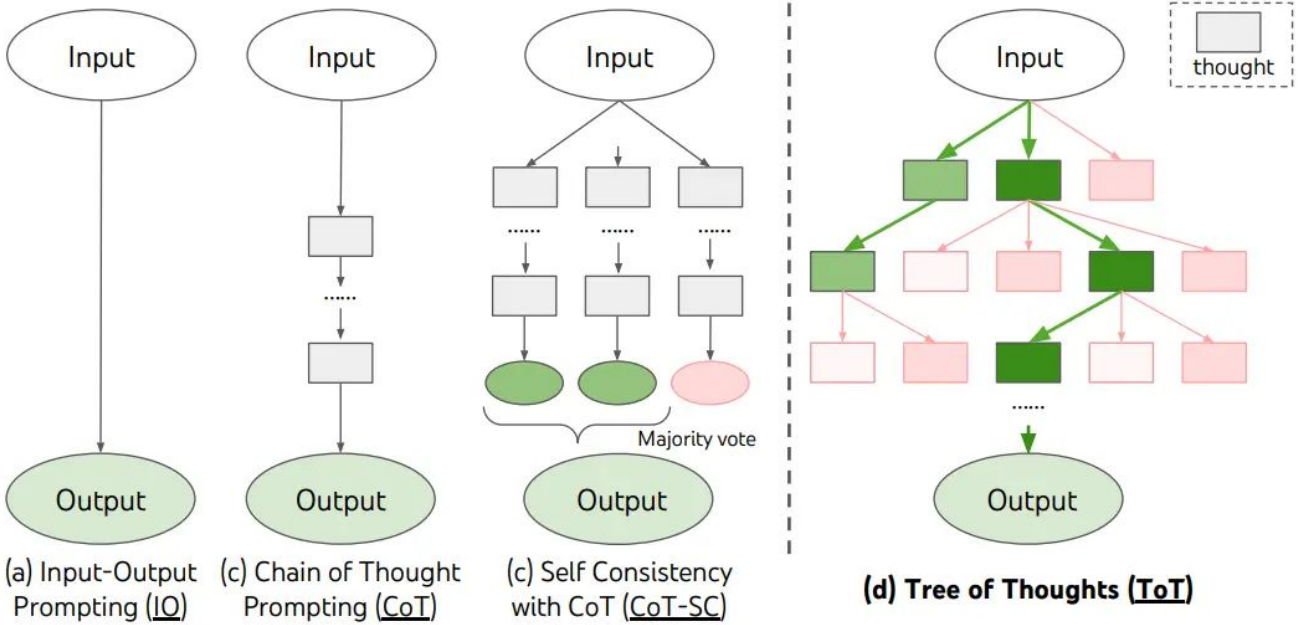
Knowledge: The objective of golf is to play a set of holes in the least number of strokes. A round of golf typically consists of 18 holes. Each hole is played once in the round on a standard golf course. Each stroke is counted as one point, and the total number of strokes is used to determine the winner of the game.

Explain and Answer:

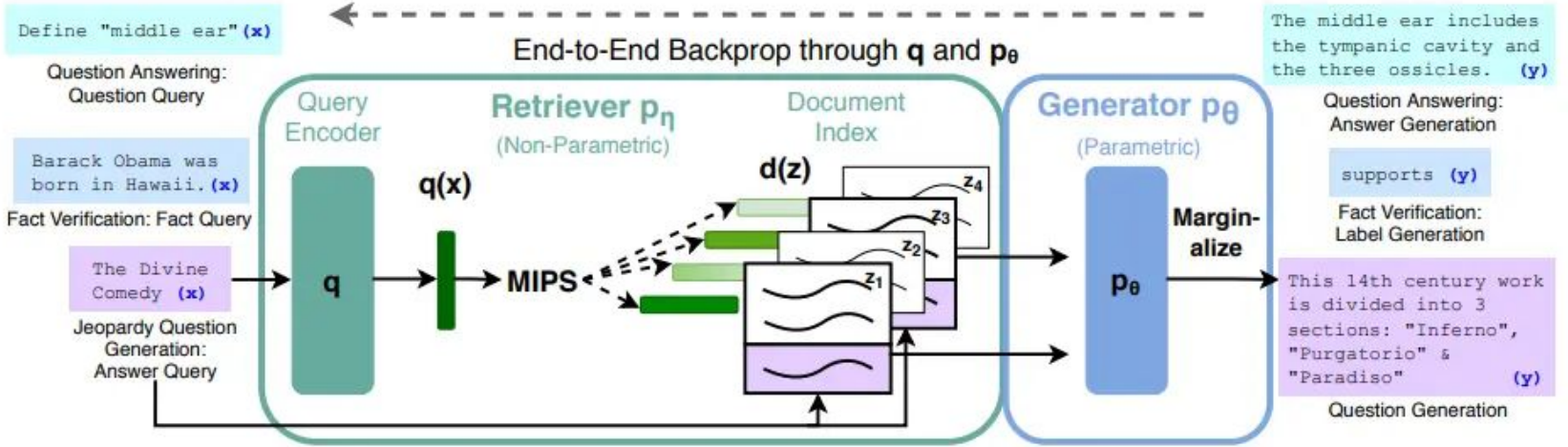
OUTPUT:

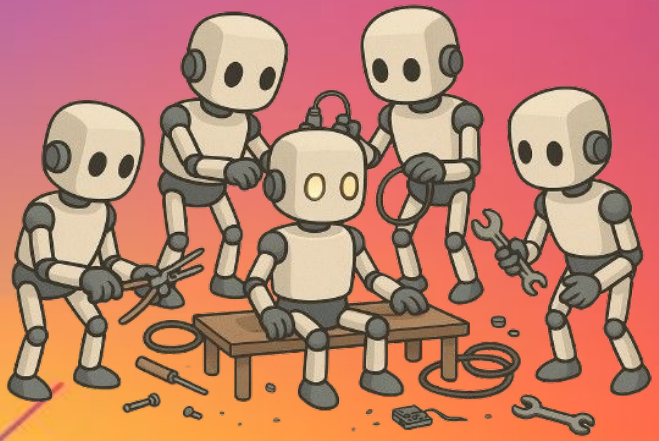
No, the objective of golf is not to get a higher point total than others. Rather, the objective is to play a set of holes in the least number of strokes. The total number of strokes is used to determine the winner of the game, not the total number of points.

Tree of Thoughts (ToT)



Retrieval Augmented Generation (RAG)





Content

- From LLMs to AI Agents
- Anatomy of an AI Agent
- Function calling & MCPs
- Building an AI Agent

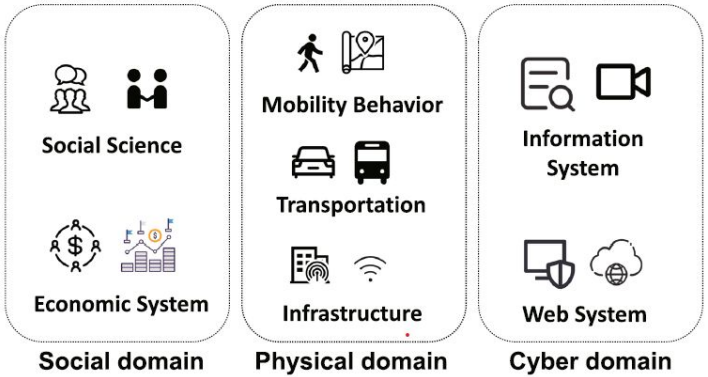
MICAI 2025

INAOE

AI Agents

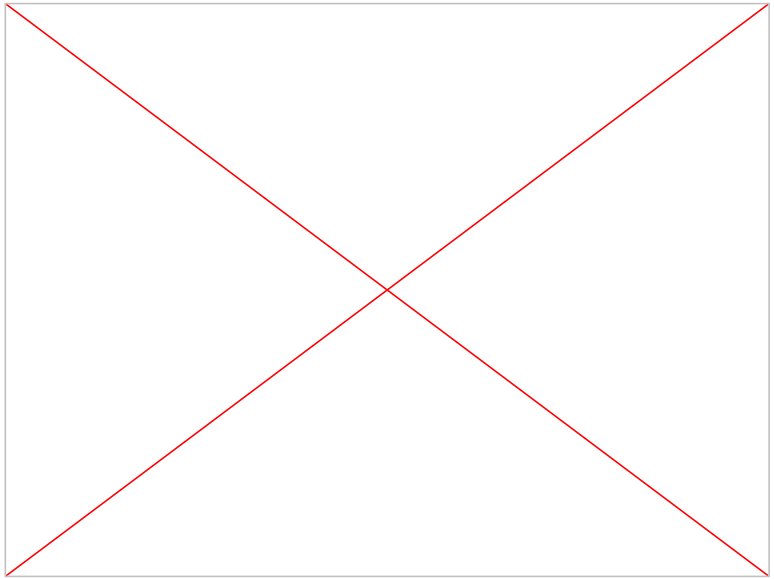
What can one do with “Agents”?

- Simulation of individual behavior
- Simulation of collective behavior



Gao, C., Lan, X., Li, N. et al. Large language models empowered agent-based modeling and simulation: a survey and perspectives. Humanit Soc Sci Commun 11, 1259 (2024). https://doi.org/10.1057/s41599-024-03611-3

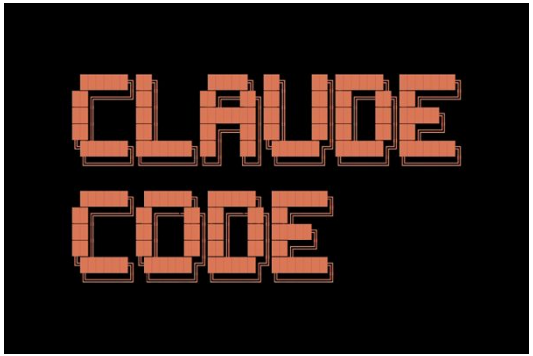
https://www.youtube.com/watch?v=ZdoU9vI2yCg



Joon Sung Park et al. Generative Agents: Interactive Simulacra of Human Behavior. arXiv:2304.03442v2

What can one do with “AI Agents”?

Trending AI Agent applications…



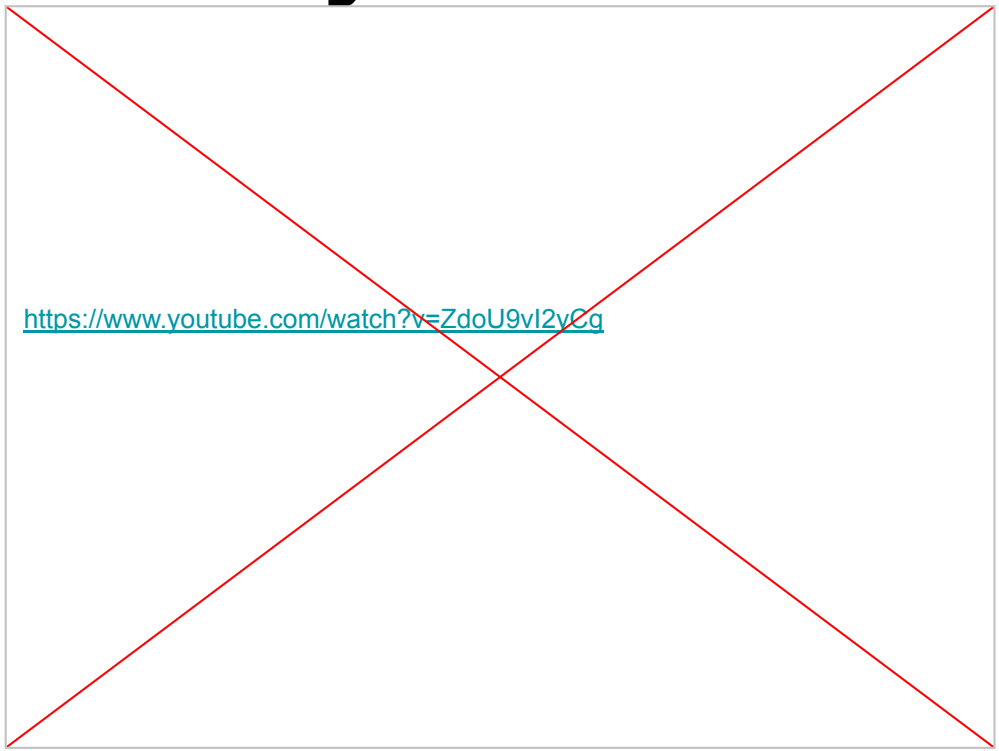
Coding

Custom Agentic Workflows

Note Taking

Joon Sung Park et al. Generative Agents: Interactive Simulacra of Human Behavior. arXiv:2304.03442v2

What can one do with “AI Agents”?



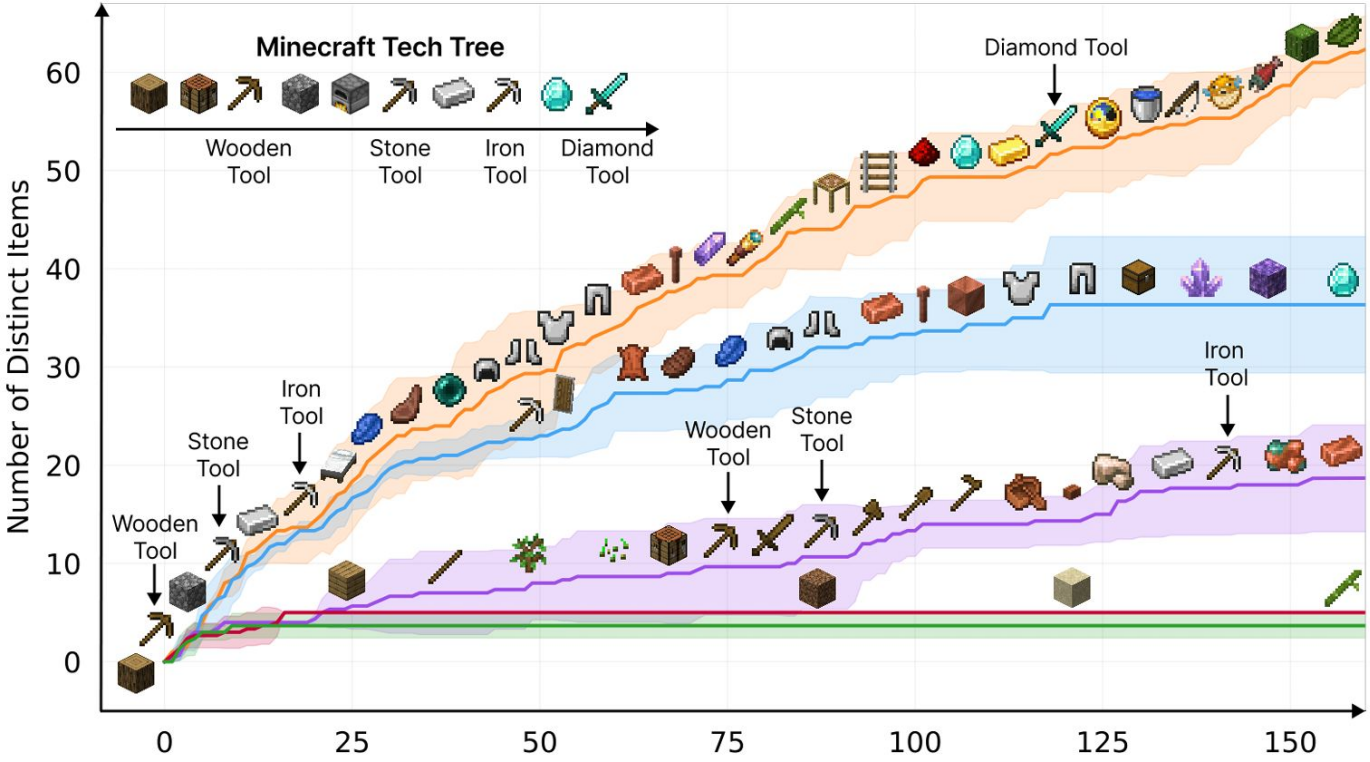
https://www.youtube.com/watch?v=ZdoU9vI2vCg

Joon Sung Park et al. Generative Agents: Interactive Simulacra of Human Behavior. arXiv:2304.03442v2

What can one do with “AI Agents”?

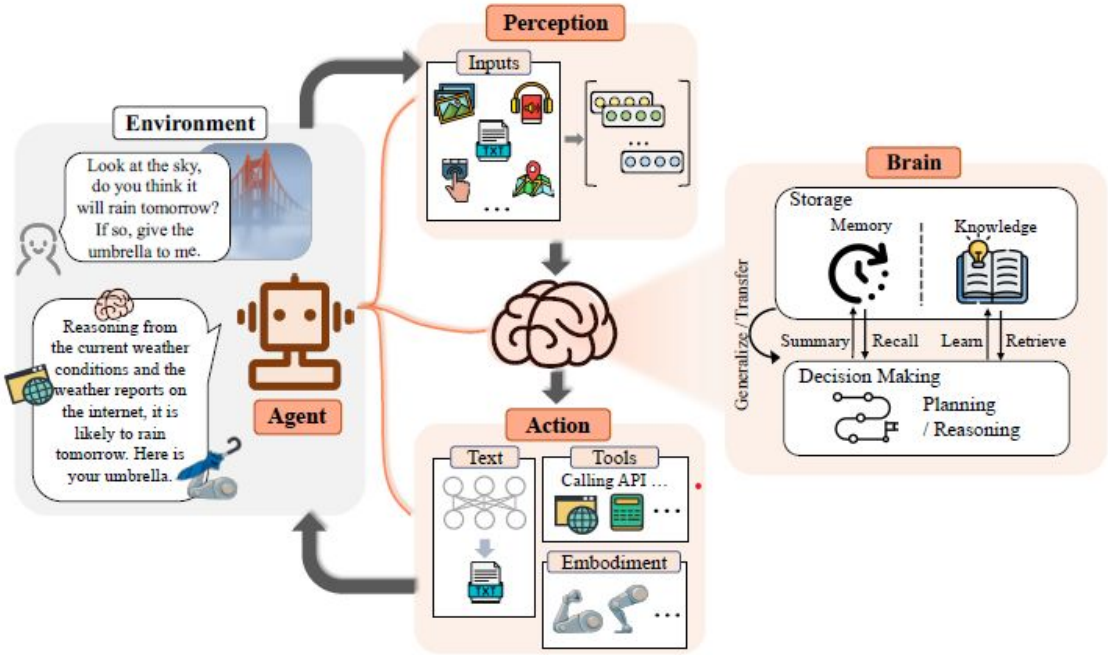
What can one do with “AI Agents”?

Guanzhi Wang, et al. Voyager: An Open-Ended Embodied Agent with Large Language Models https://arxiv.org/abs/2305.16291



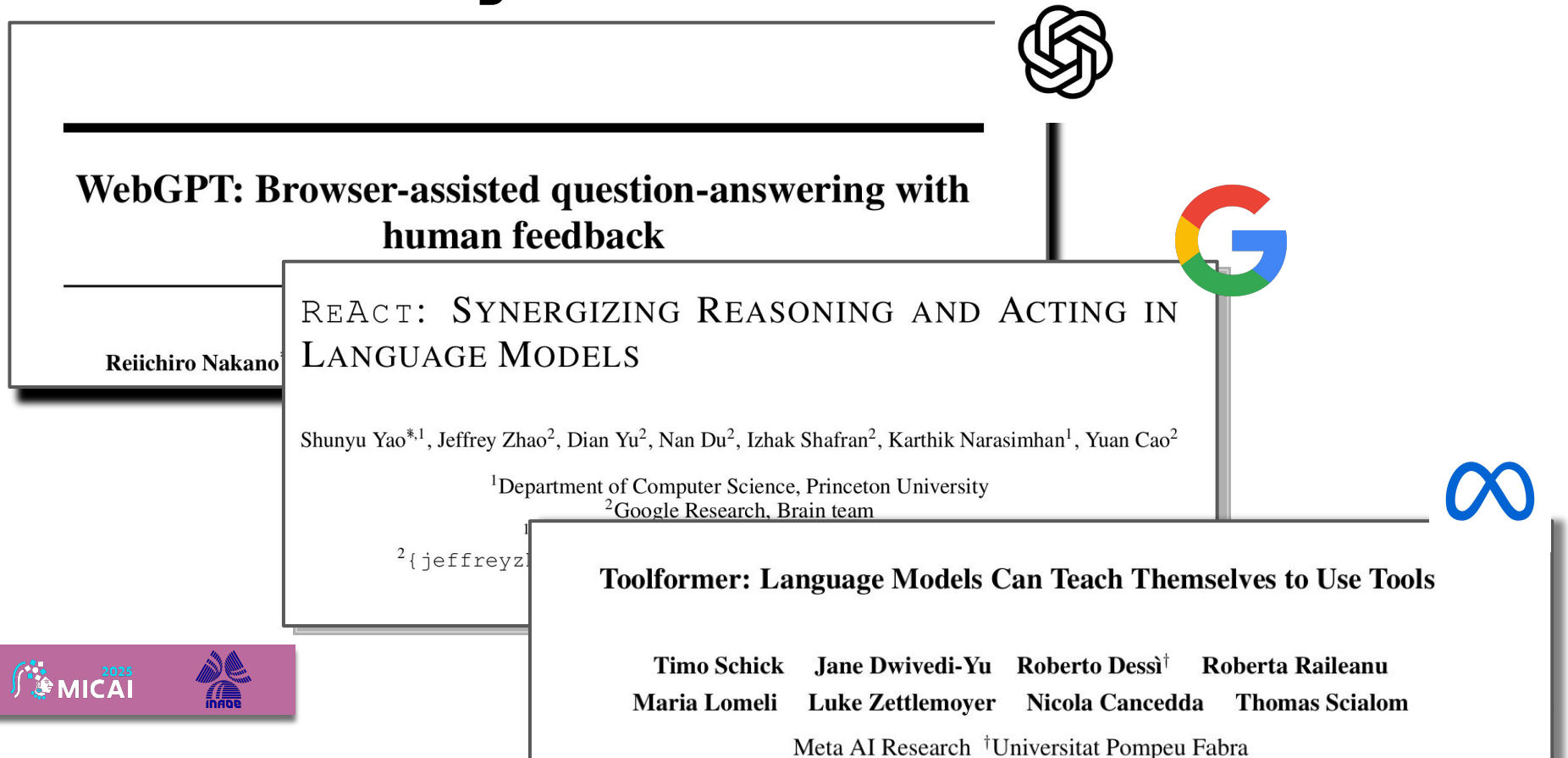
Anatomy of an AI Agent (in a nutshell)

1. An agent interacts with an **environment**
2. It **perceives** the environment through available modalities
3. Processes information and **plans** the next step
4. Takes **action** by executing tools



Z. Xi et al. The Rise and Potential of Large Language Model Based Agents: A Survey. https://arxiv.org/abs/2309.07864

From LLMs to AI Agents



WebGPT: Browser-assisted question-answering with human feedback

Reiichiro Nakano

REACT: SYNERGIZING REASONING AND ACTING IN LANGUAGE MODELS

Shunyu Yao*,1, Jeffrey Zhao2, Dian Yu2, Nan Du2, Izhak Shafran2, Karthik Narasimhan1, Yuan Cao2

1Department of Computer Science, Princeton University
2Google Research, Brain team

2{jeffreyz

Toolformer: Language Models Can Teach Themselves to Use Tools

Timo Schick Jane Dwivedi-Yu Roberto Dessì† Roberta Raileanu
Maria Lomeli Luke Zettlemoyer Nicola Cancedda Thomas Scialom

Meta AI Research †Universitat Pompeu Fabra

Function Calling

How do we make the LLM to generate **valid** function calls?

```
What is the weather today in
Guanajuato?
```

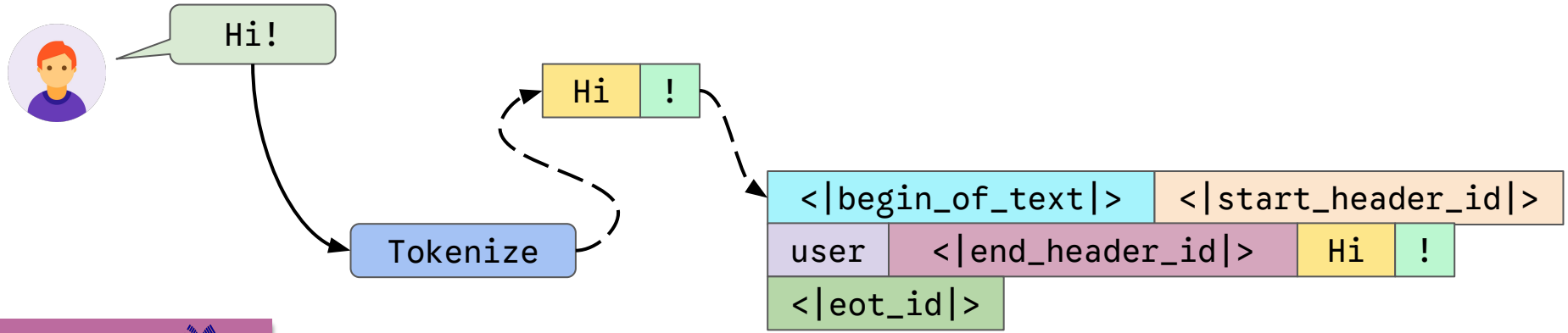
```
get_weather_api(date="2025-11-03",
city="Guanajuato")
```

1) Teach the language model how to respond by **finetuning** (SFT)
2) Constrain the model at decoding-time… grammars, regex, and state machines are techniques used for **controlled generation**

Function Calling

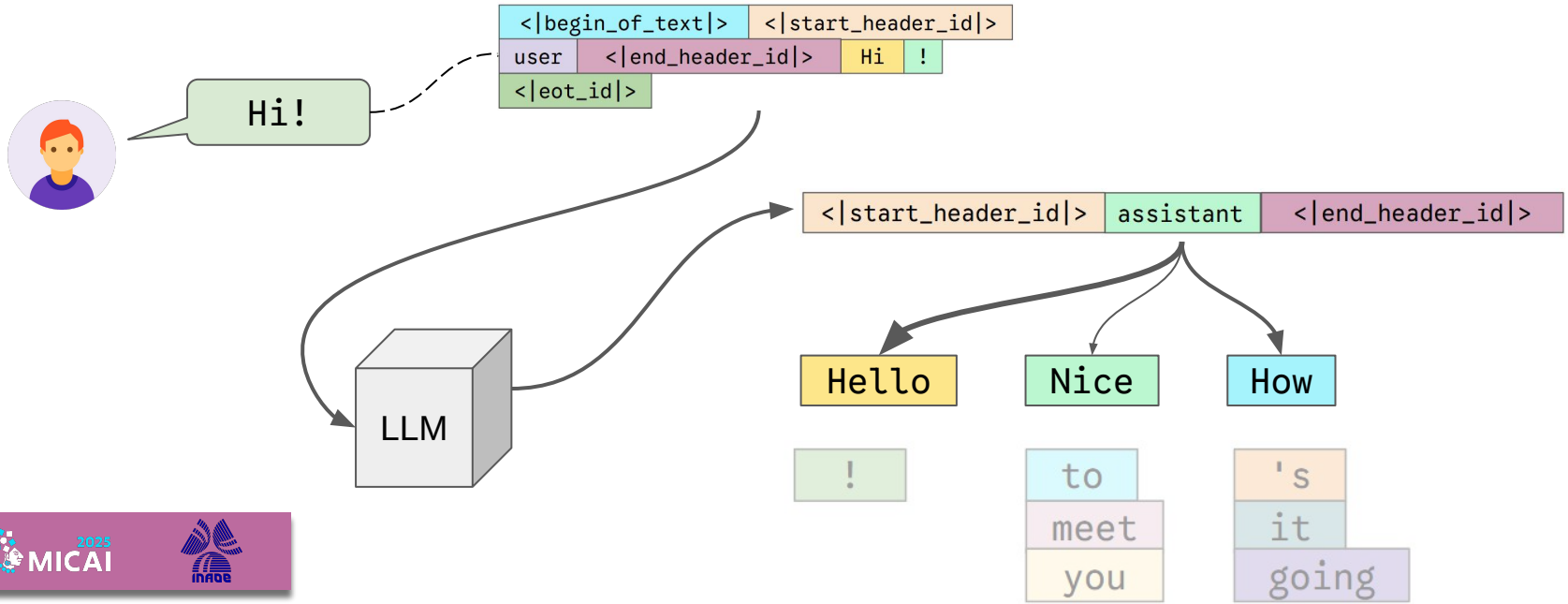
Let’s recall a language model processes raw input by first tokenizing.

- Each model has its own vocabulary, and some special tokens are added depending on the use case



Function Calling

If using **greedy sampling**, the model would choose the most probable token.



Function Calling

Going back to function calling, **how do we prepare a dataset for SFT?**

1. Leverage system prompt in training dataset to instruct for function calling

```
You are a helpful assistant with tool calling capabilities.
```

2. Pass the function JSON

```
You have access to the following functions...
```

3. Introduce a new ID when finetuning the model for function calling

<|start_header_id|>	tool	<|end_header_id|>

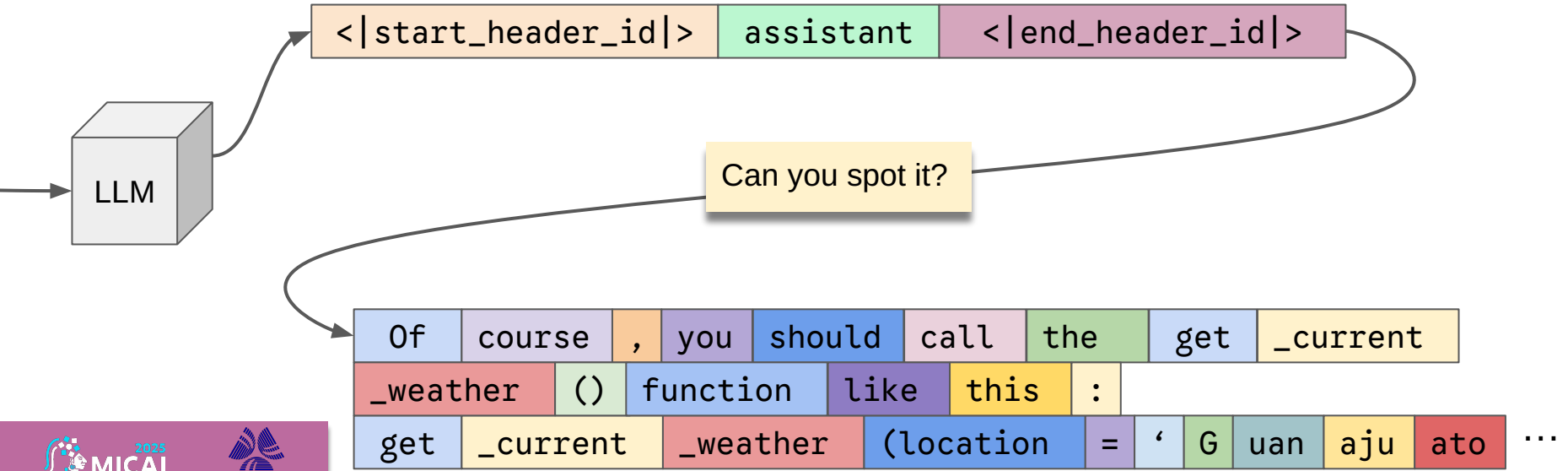
Function Calling

```
<|begin_of_text|><|start_header_id|>system<|end_header_id|>

Cutting Knowledge Date: December 2023
Today Date: 23 July 2024

When you receive a tool call response, use the output to format an answer to the orginal user question.

You are a helpful assistant with tool calling capabilities.<|eot_id|><|start_header_id|>user<|end_header_id|>

Given the following functions, please respond with a JSON for a function call with its proper arguments that best answers the given prompt.

Respond in the format {"name": function name, "parameters": dictionary of argument name and its value}. Do not use variables.

{
    "type": "function",
    "function": {
    "name": "get_current_conditions",
    "description": "Get the current weather conditions for a specific location",
    "parameters": {
        "type": "object",
```

Example from Llama 3.1 model card: [Llama 3.1 | Model Cards and Prompt formats]

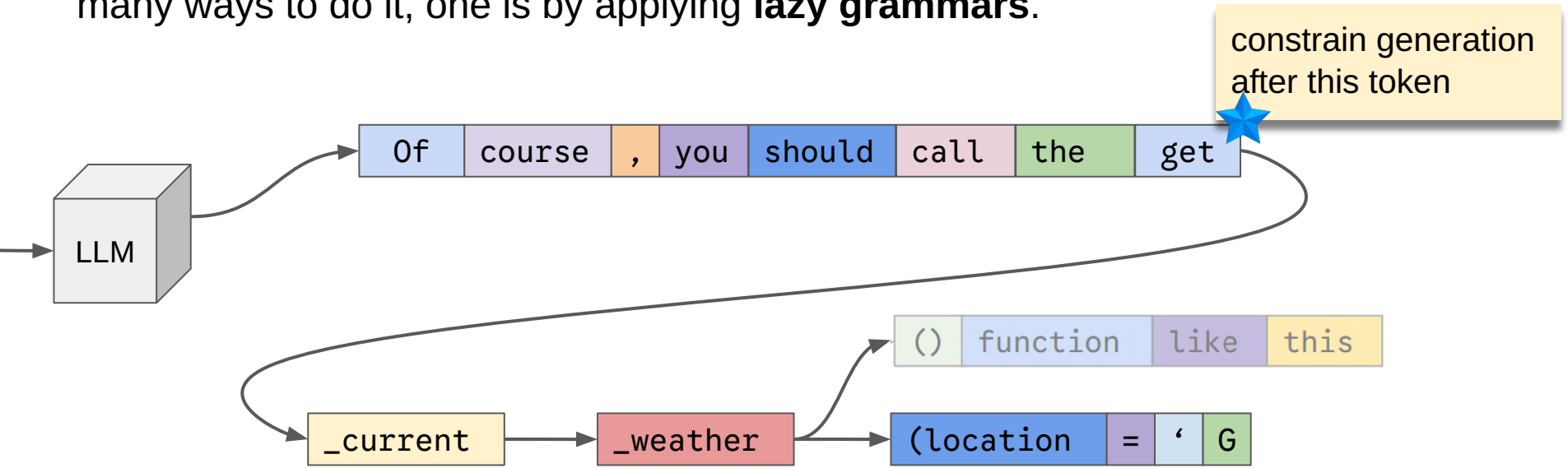
Function Calling

First approaches for function calling let the model generate freely, then **parsed** the function call from the model's output. However, generation errors make it hard to parse the function.



Function Calling

One way to avoid this issue is to **control the generation process**. There are many ways to do it, one is by applying **lazy grammars**.



Lazy grammars are implemented as part of llama-cpp: [Tool call support (generic + native for Llama, Functionary, Hermes, Mistral, Firefunction, DeepSeek) w/ lazy grammars by ochafik · Pull Request #9639 · ggml-org/llama.cpp · GitHub]

Function Calling

Let's try function calling with the OpenAI Python completions module.

Model Context Protocol (MCP)

- Define of MCP
- Fundamentals of MCP
- Build MCP Server (later)

What is MCP?

“An open protocol that standardizes how your LLM applications connect to and work with your tools and data sources” (ANTHROP\C)

What is MCP (Model Context Protocol)?

REST APIs

Standardize how **web applications** interact with the **backend**

LSP

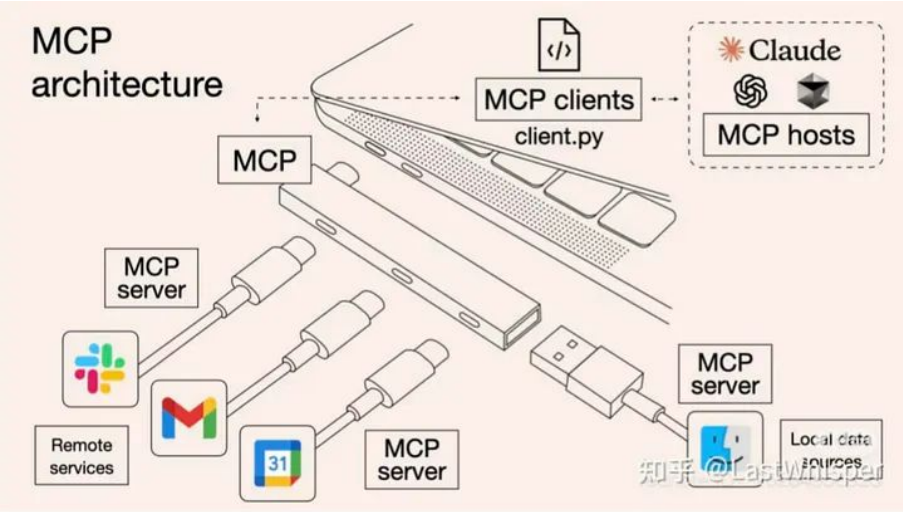
Standardizes how **IDEs** interact with **language-specific tools**

MCP

Standardizes how **AI applications** interact with **external systems**

What is MCP (Model Context Protocol)?

“Think of MCP like a USB-C port for AI applications. Just as USB-C provides a standardized way to connect electronic devices, MCP provides a standardized way to connect AI applications to external systems.”

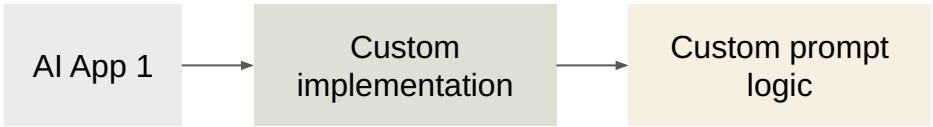


Without MCP: Fragmented AI Development

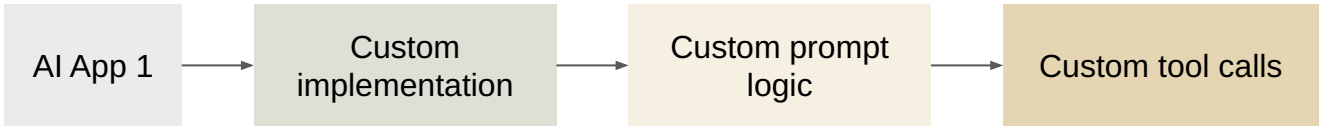
AI App 1

Without MCP: Fragmented AI Development

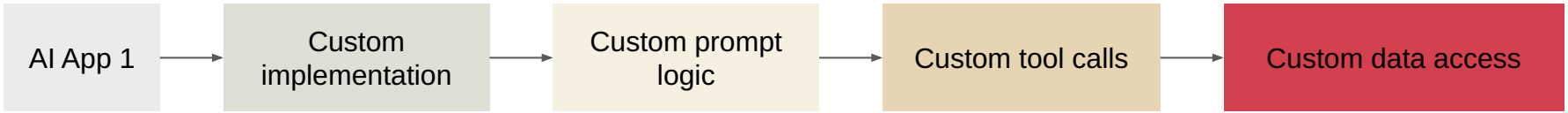
Without MCP: Fragmented AI Development



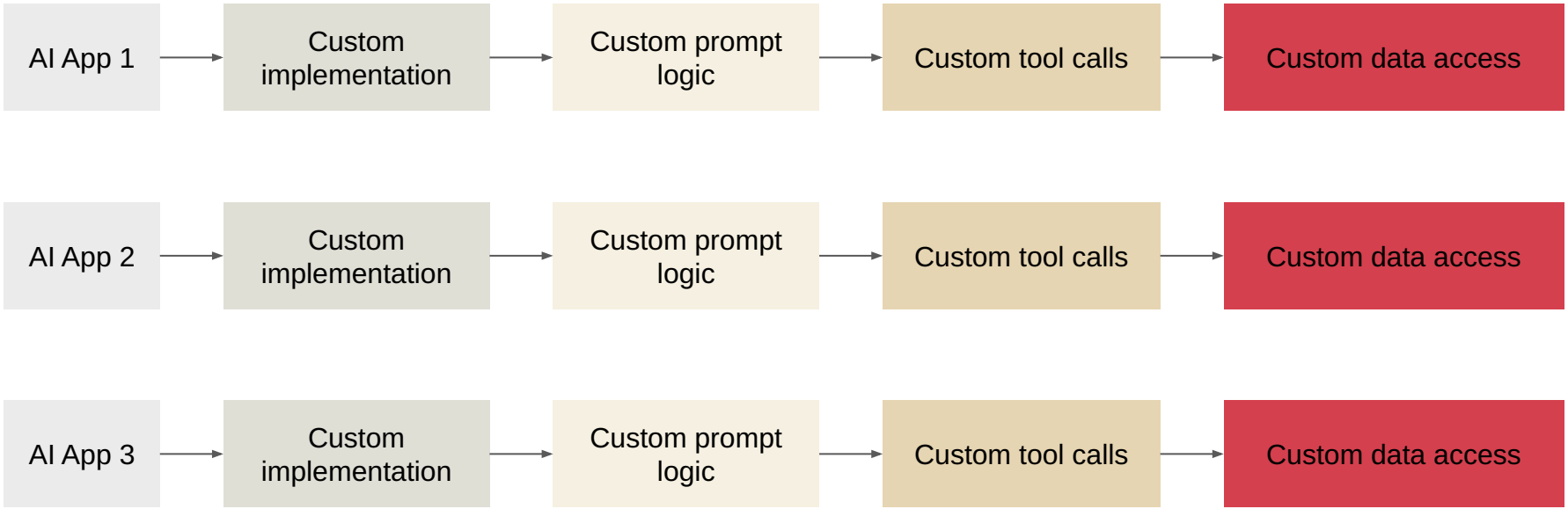
Without MCP: Fragmented AI Development



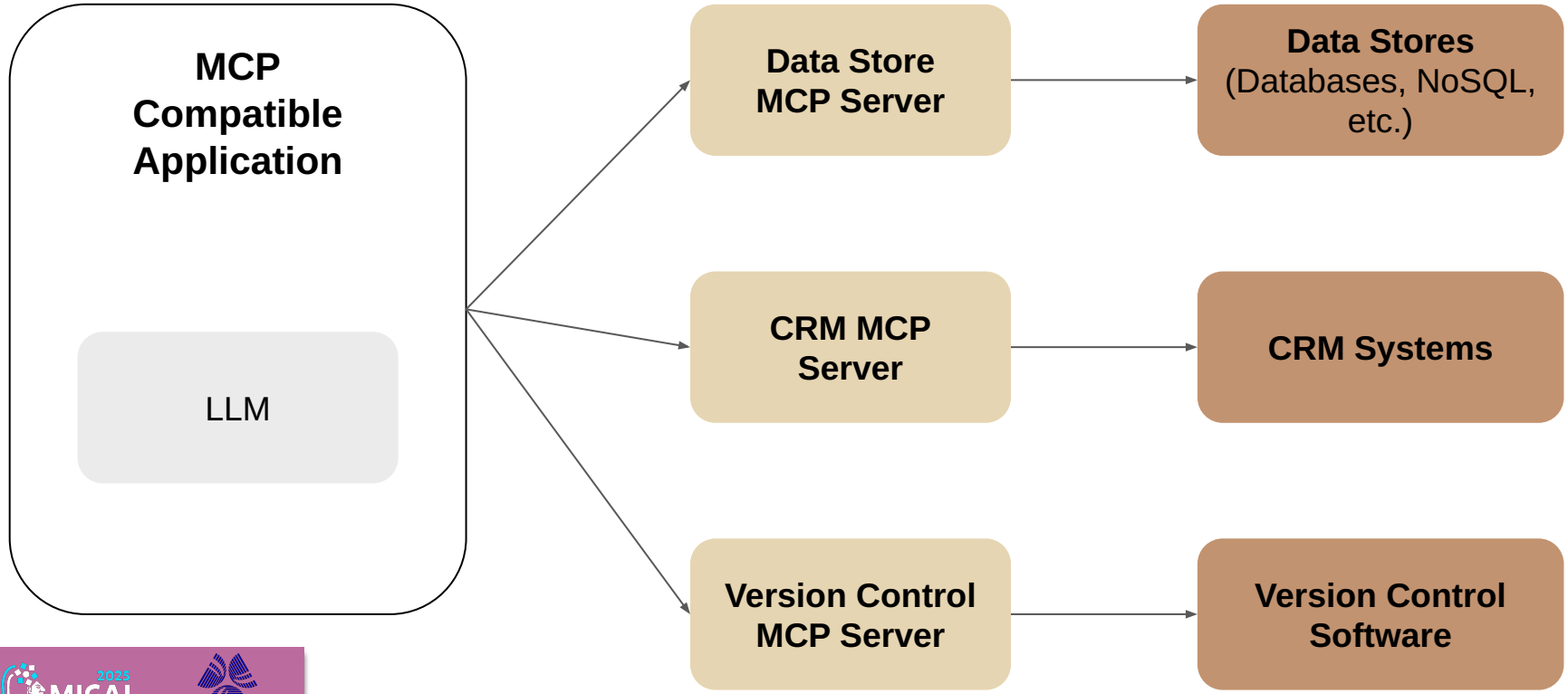
Without MCP: Fragmented AI Development



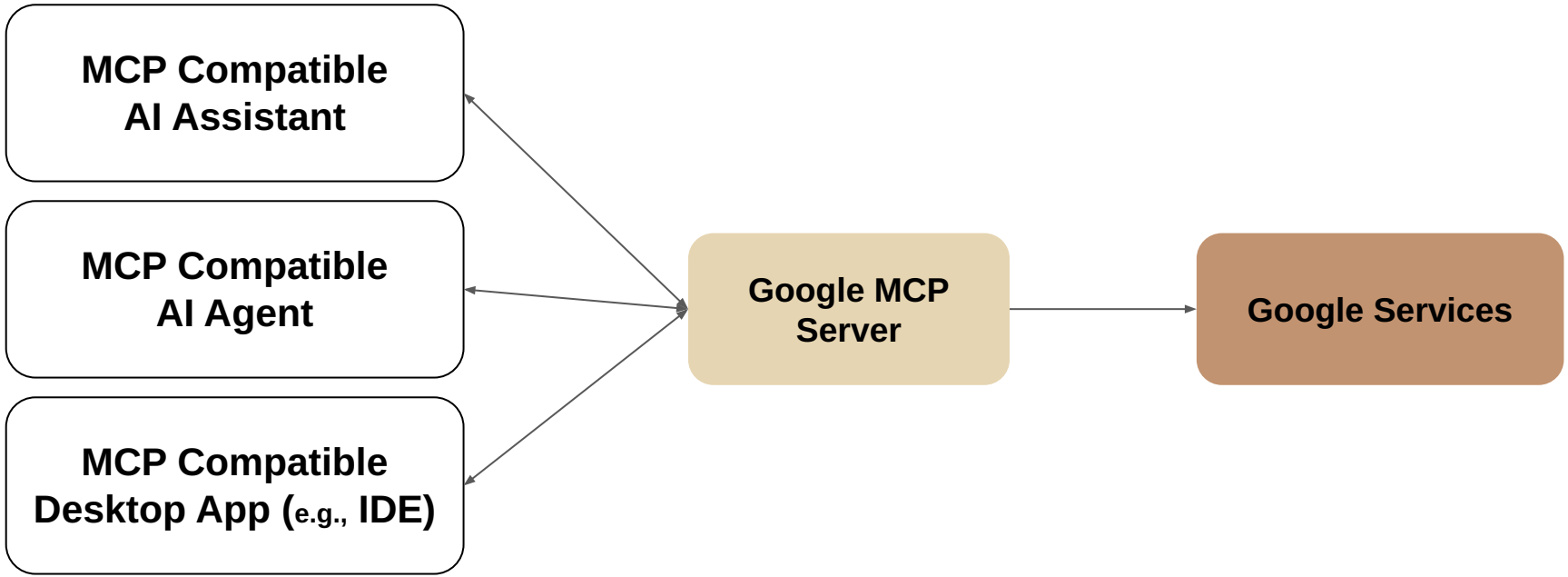
Without MCP: Fragmented AI Development



With MCP : Standardized AI Development

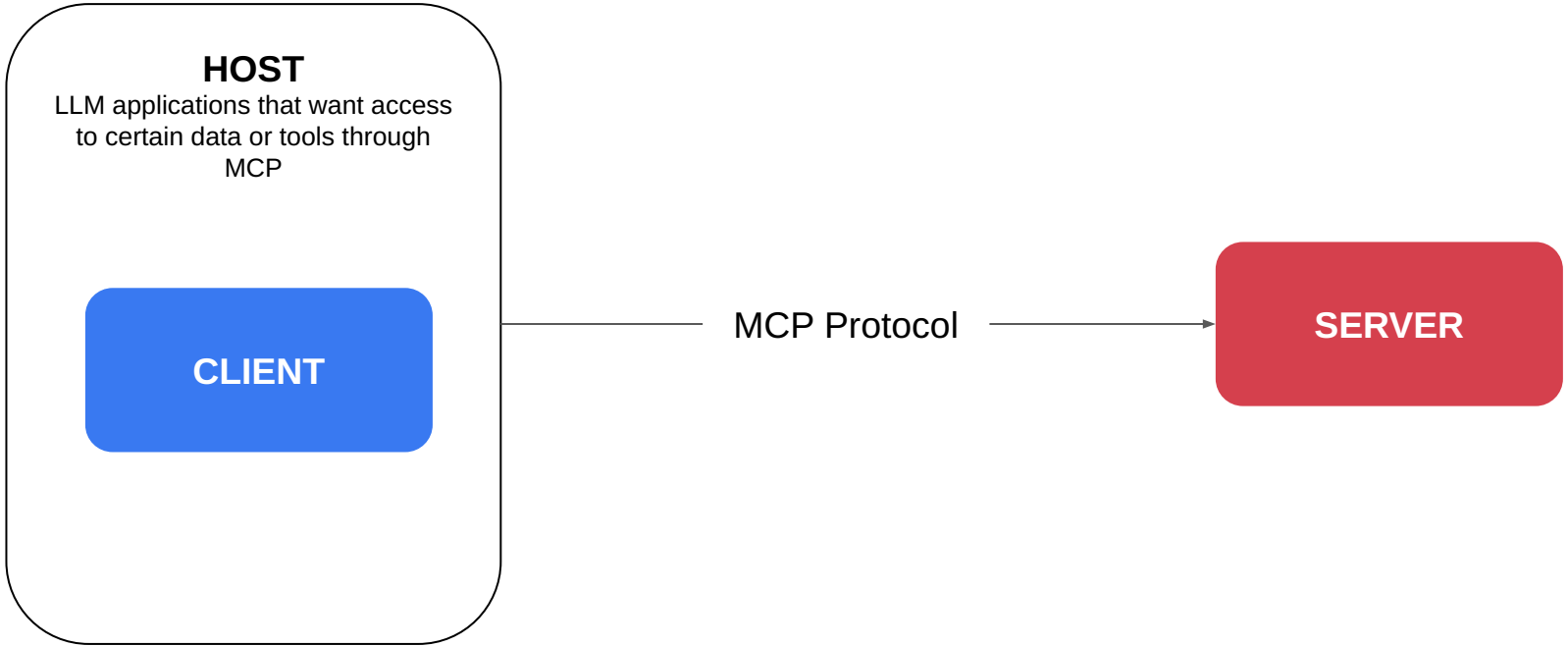


With MCP : MCP Server are reusable by AI-Apps



Client-Server Architecture

Client-Server Architecture



Components inside an MCP Server

TOOLS

RESOURCES

PROMPT TEMPLATES

Components inside an MCP Server

TOOLS

Functions and tools that can be invoked by the client

RESOURCES

PROMPT TEMPLATES

Components inside an MCP Server

TOOLS

RESOURCES

Read-only data that is exposed by the server

PROMPT TEMPLATES

Components inside an MCP Server

TOOLS

RESOURCES

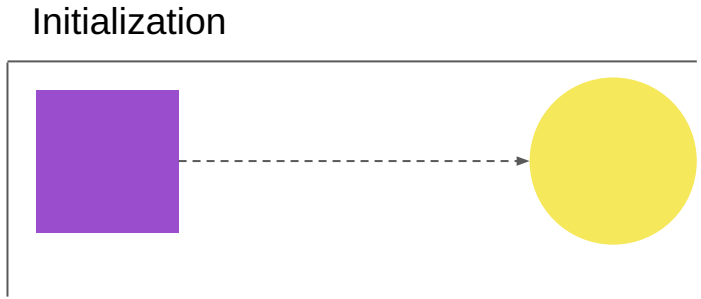
PROMPT TEMPLATES

Structured prompt blueprints

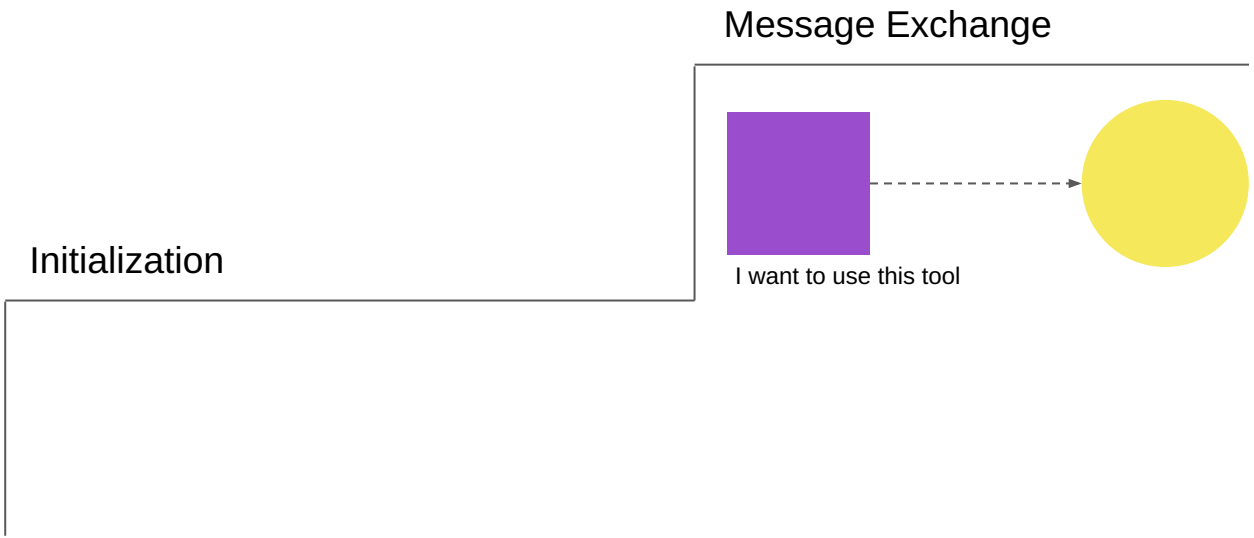
Communication Life Cycle

Initialization

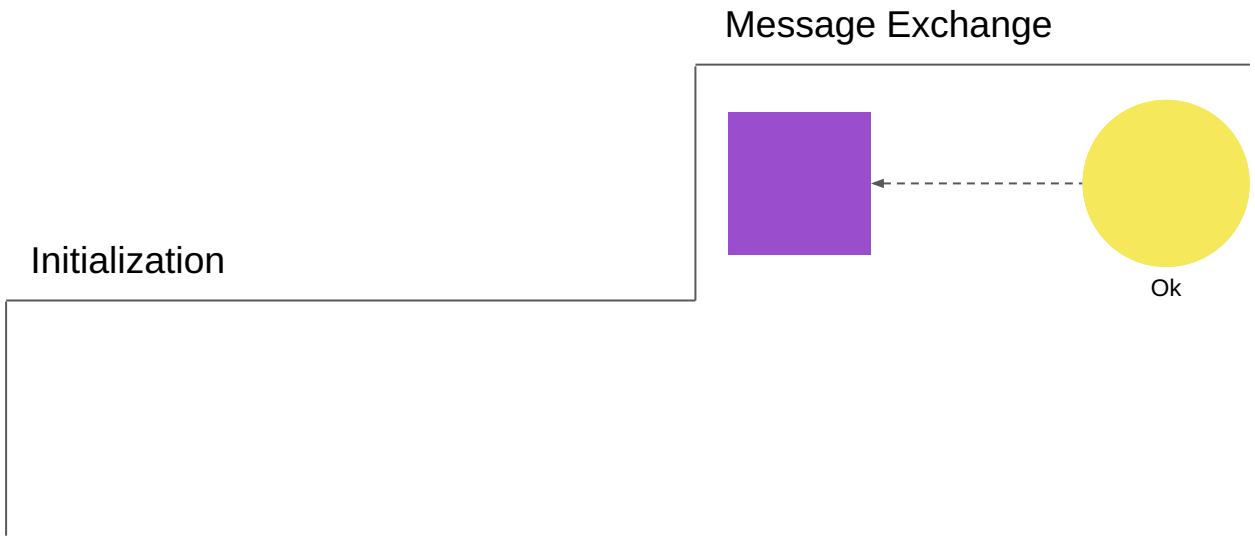
Communication Life Cycle



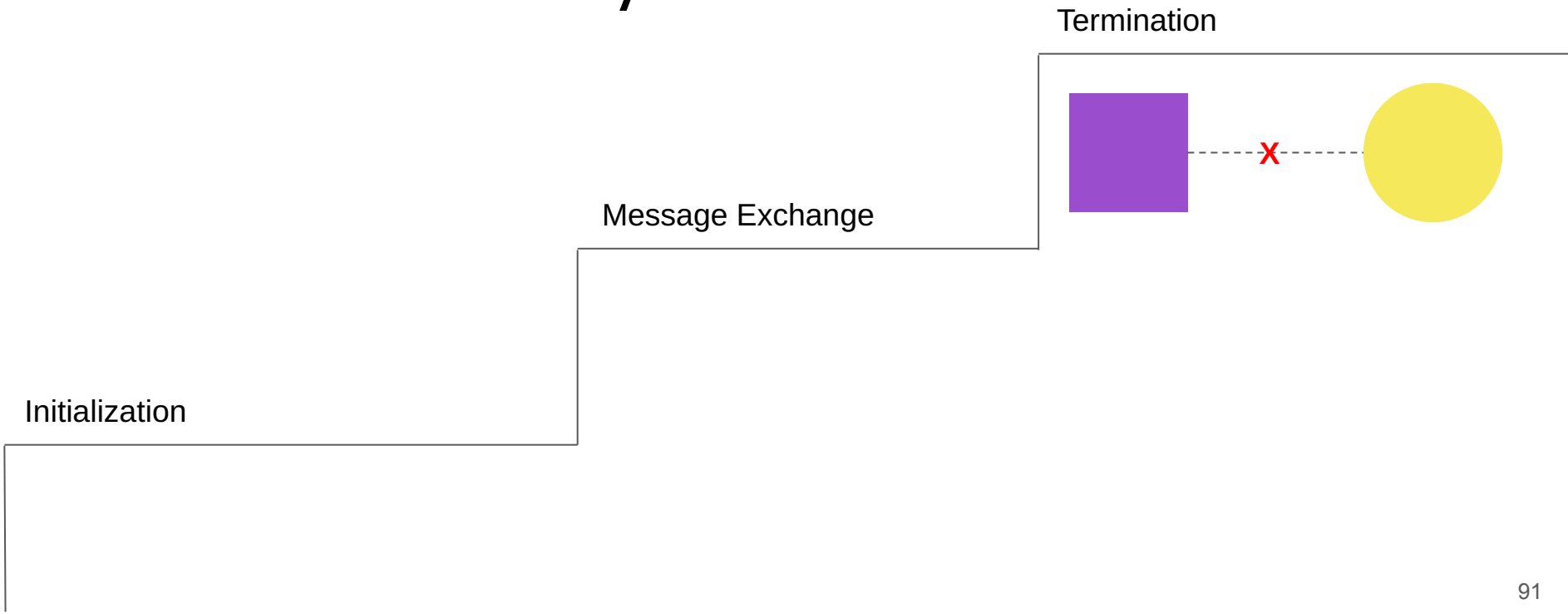
Communication Life Cycle



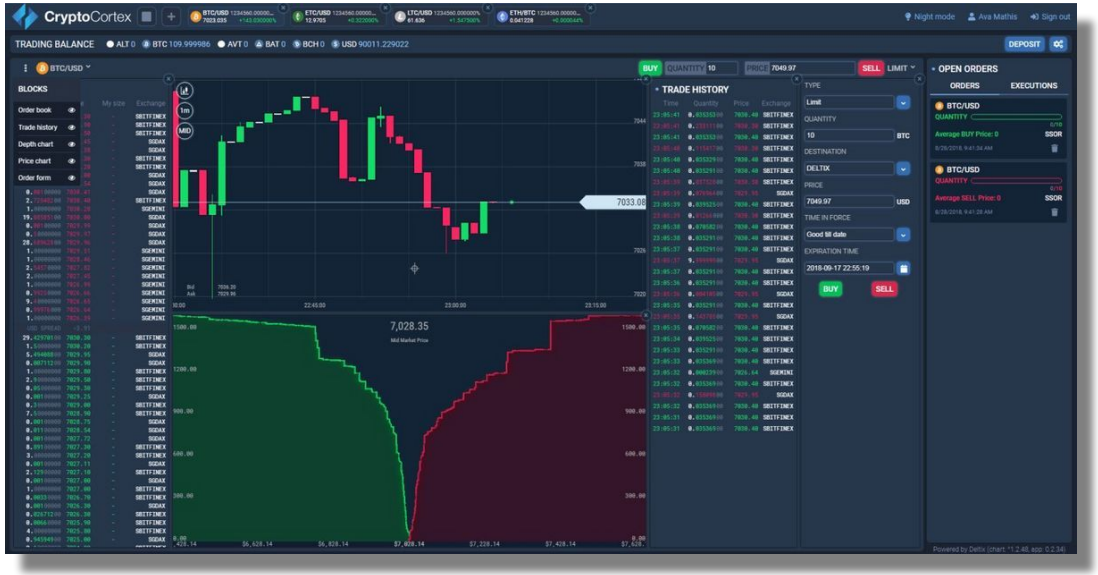
Communication Life Cycle



Communication Life Cycle



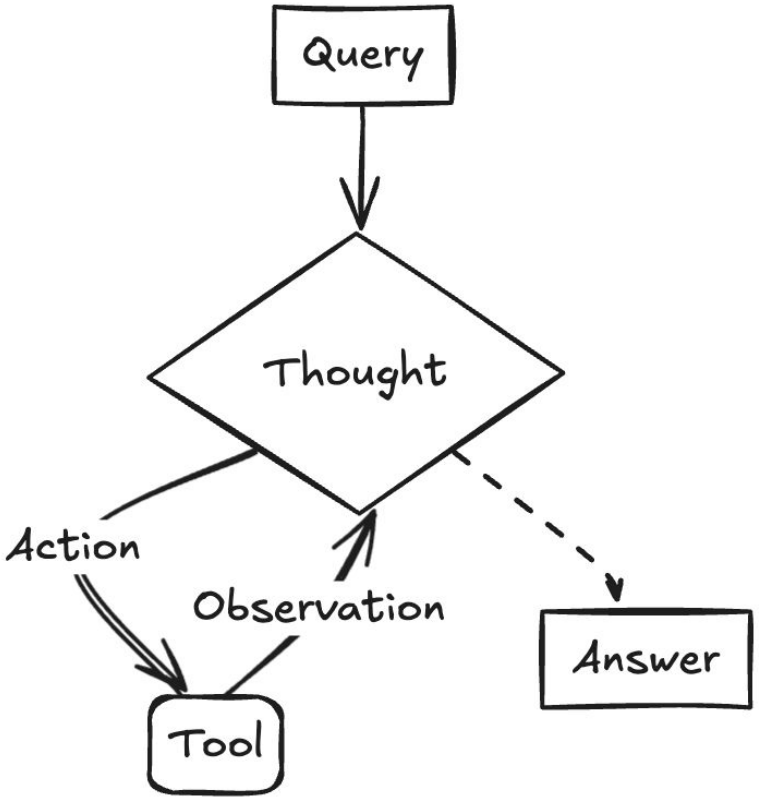
Practice - MCP Servers



LLM Agent Architecture

Currently, there are multiple LLM agent architectures…

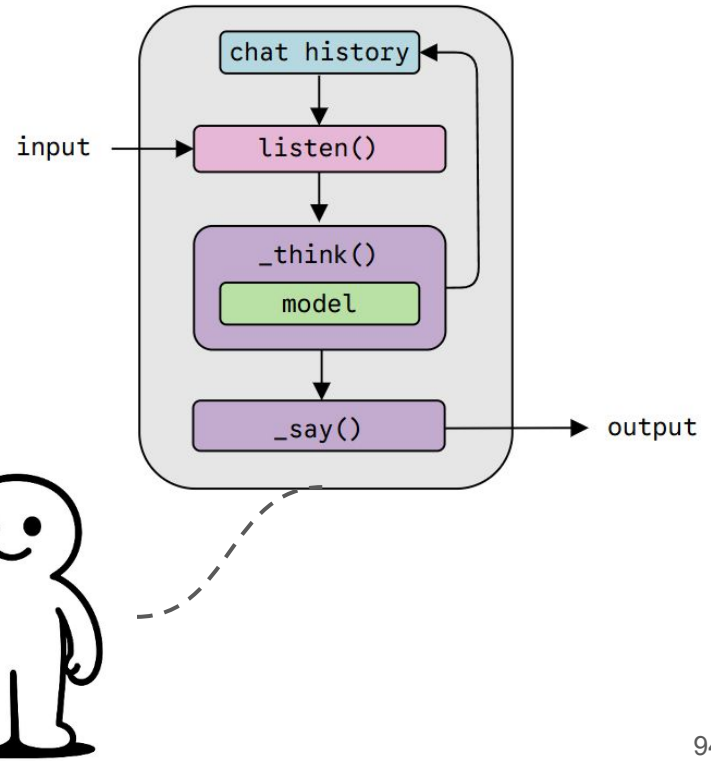
- The **ReAct** paper is a first reference on how to prompt language models for synergizing *reasoning and acting* general task solving



LLM Agent Architecture

Let's set the foundations of an **agentic loop**

- Set `listen()` method to query the agent
- Generate a completion with the `_think()` method
- Respond back to user with `say()` method
- Store chat history in a list of messages
- `listen()` just packs what's in the chat history and sends it to the `_think()` method

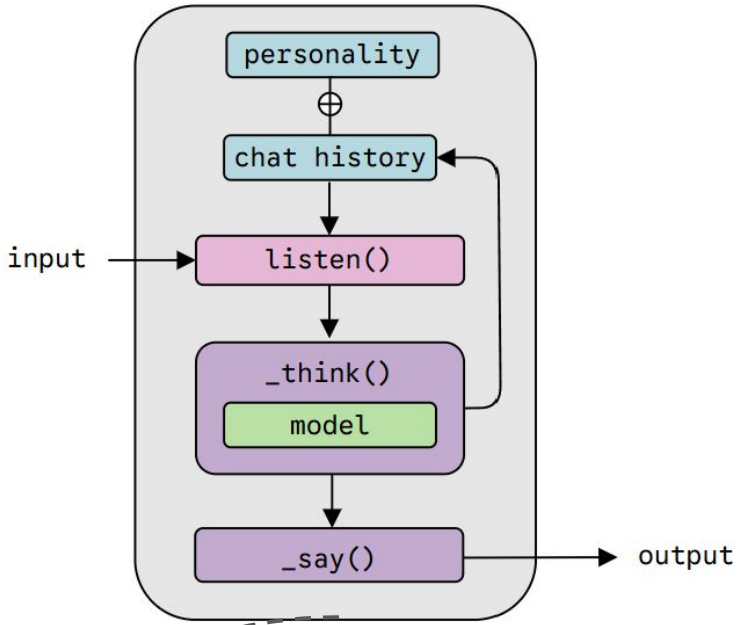


LLM Agent Architecture

Then, we can use the system prompt to define:

- Personality - for better task specific generations
- Generation guidelines - for setting the objective and indicating how it should respond on every call

Notice how the agent loop stops on every `listen()` call.

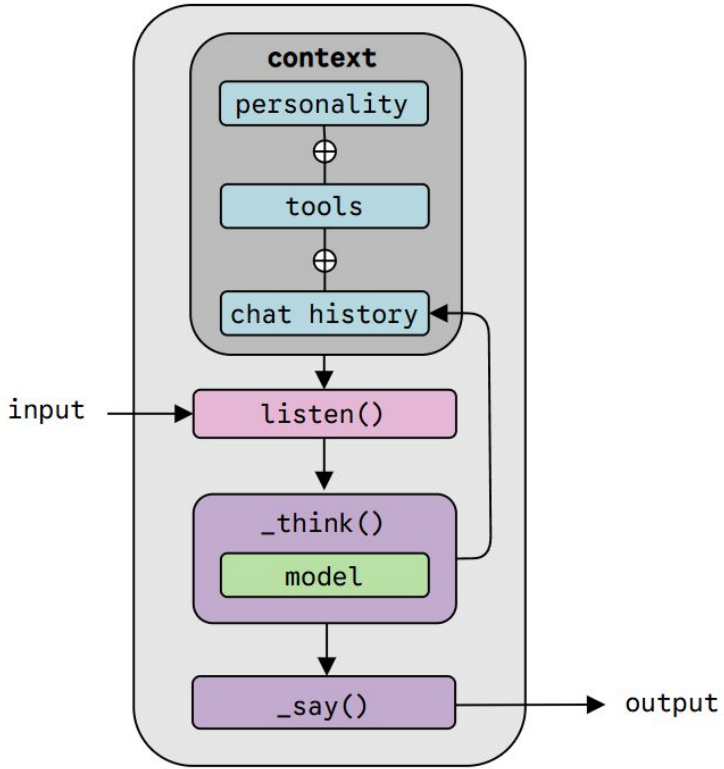


LLM Agent Architecture

Continue by adding tools for the agent to select, but keep in mind that

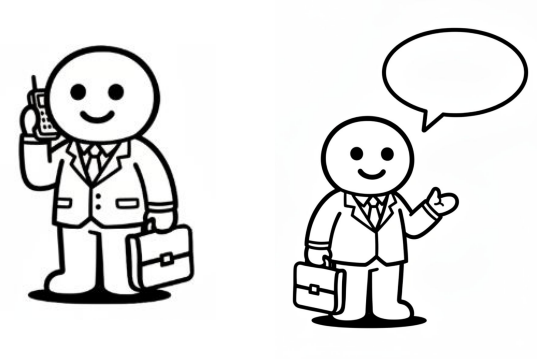
- The tool definition must be included as part of the context
- The model can generate a normal response, or a tool call, or both

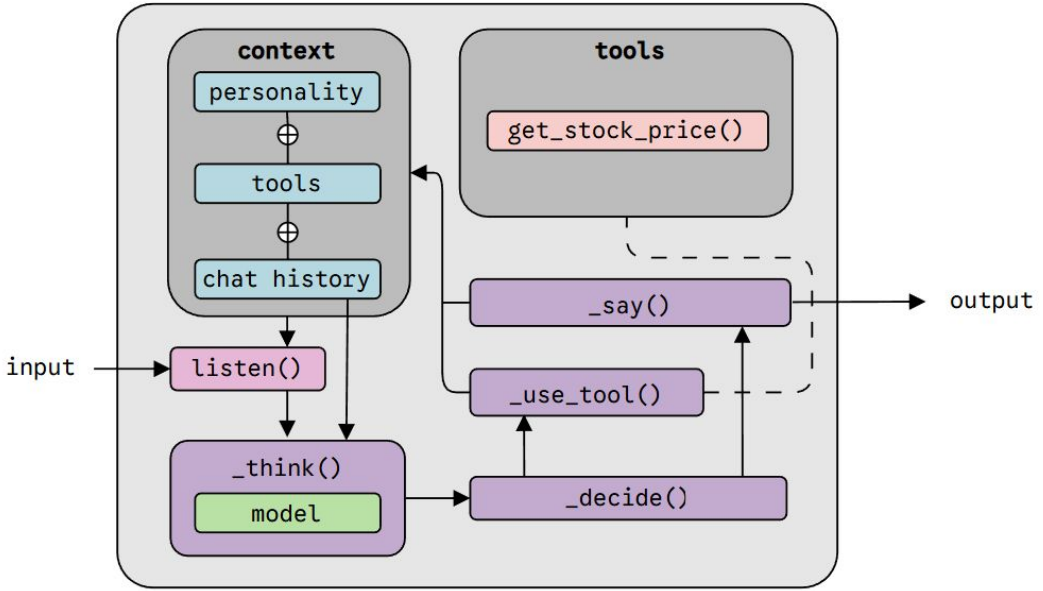
We must handle this options by introducing a new block…

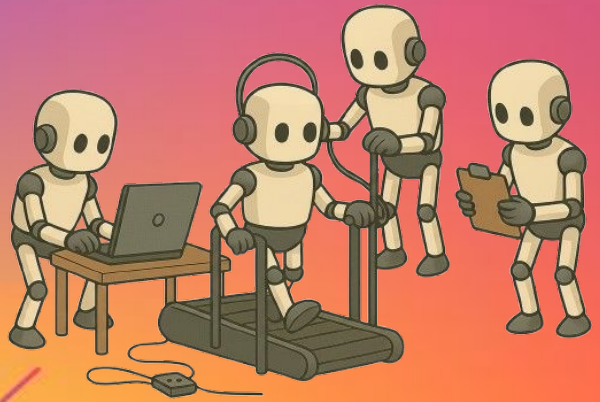


LLM Agent Architecture

- The `_decide()` block allows us to **handle** the next step after a generation







Content

- From LLMs to AI Agents
- AI Agents vs Agentic AI
- Popular applications

MICAI 2025

INAOE

AI Agents vs Agentic

From LLMs to Agents

- What LLMs/SLMs can do?
  - *“Understanding, reasoning, generating, etc.”*

- Limitation
  - They are passive, in the sense that they cannot take action by themselves

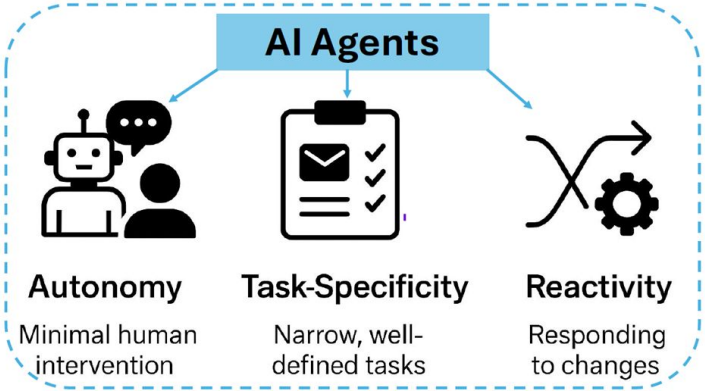
- AI Agents and Agentic AI
  - Combine LLMs with tools (APIs, memory, reasoning loops, etc.)
  - Agents can perceive, plan and act in the “*world*”
  - Examples: AutoGPT, Devin, ChatGPT with tools

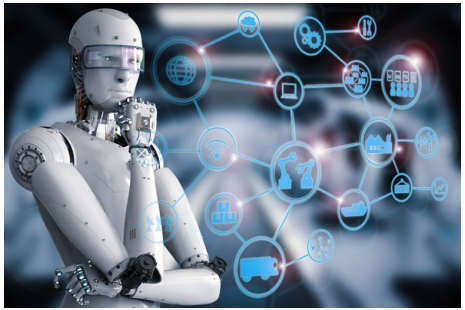
https://talent500.com/blog/large-language-models-enterprise-ai-adoption/

AI Agents vs Agentic AI

- **AIA:** Autonomous (*software*) entities for goal-directed task execution within bounded digital environments
  - Show reactive intelligence and adaptability

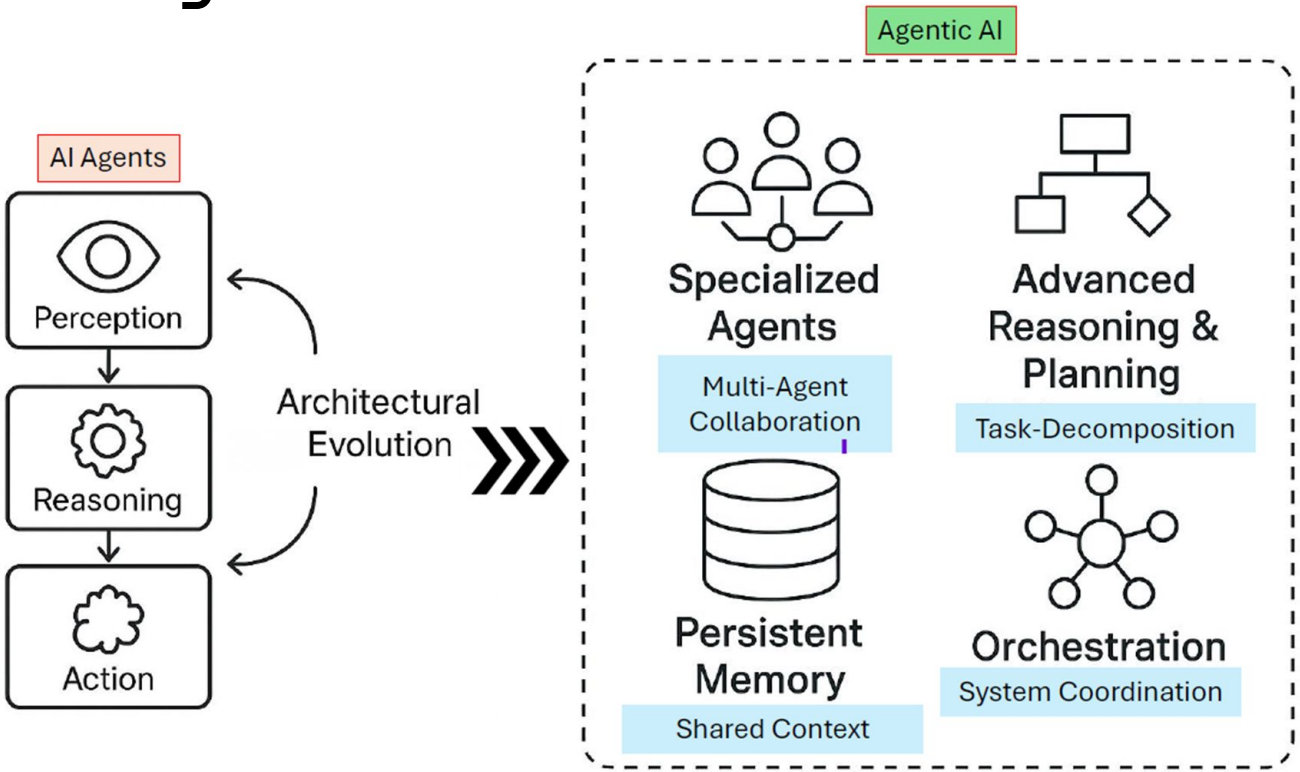
- **AAI:** Systems of multiple AI agents collaborating to achieve complex goals through structured communication, shared memory and dynamic role assignment





R. Sapkota, et al. AI Agents vs. Agentic AI: A Conceptual Taxonomy, Applications and Challenges, arXiv:2505.10468, https://arxiv.org/abs/2505.10468

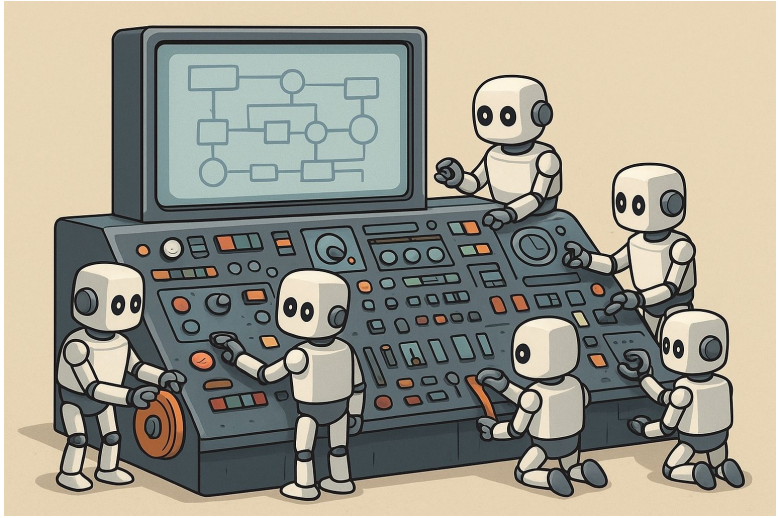
AI Agents vs Agentic AI



Applications

- AI Agents
  - Customer support automation
  - Personalized content recommendation
  - Scheduling assistants

- Agentic AI
  - Research assistants
  - Intelligent robotics coordination
  - Multi-agents in gaming

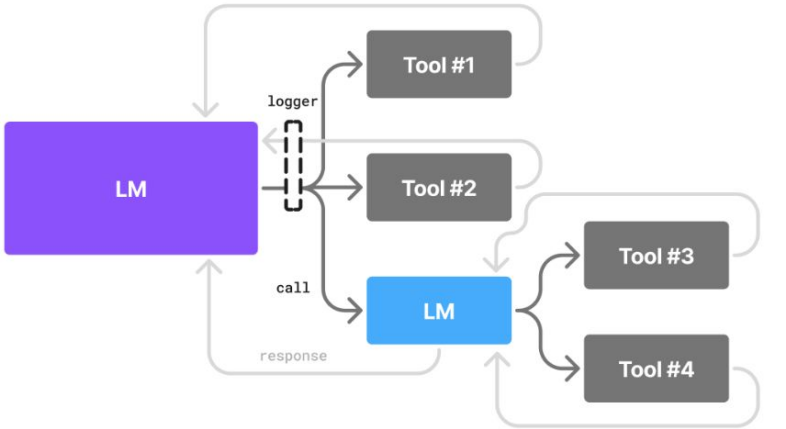


Are SLMs appropriate for AIA & AAI?

- LMs in AIA and AAI are used for repetitive, simple, scoped and non-conversational tasks (using LLMs could be a *waste* of LM resources)
- **Advantages of SLMs for AIA and AAI:** lower latency, reduced memory and computational requirements, and significantly lower operational costs
- **Further benefits:** reduced economical cost, democratization of AI

P. Belcak et al. Small Language Models are the Future of Agentic AI. arXiv:2506.02153v2
https://research.nvidia.com/labs/lpr/slm-agents/

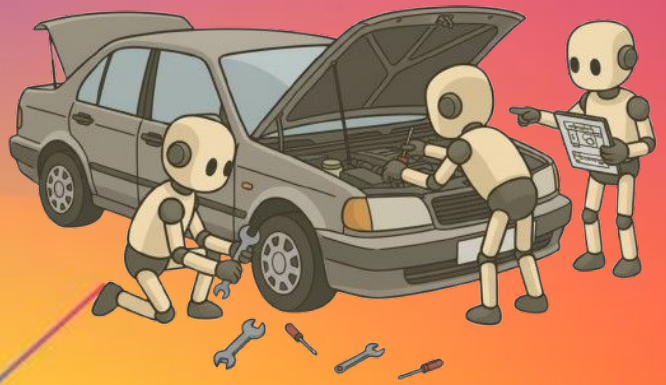
Building Agents with SLMs



P. Belcak et al. Small Language Models are the Future of Agentic AI. arXiv:2506.02153v2
https://research.nvidia.com/labs/lpr/slm-agents/

What can one do with AIA & AAI?

Idea Generation



Content

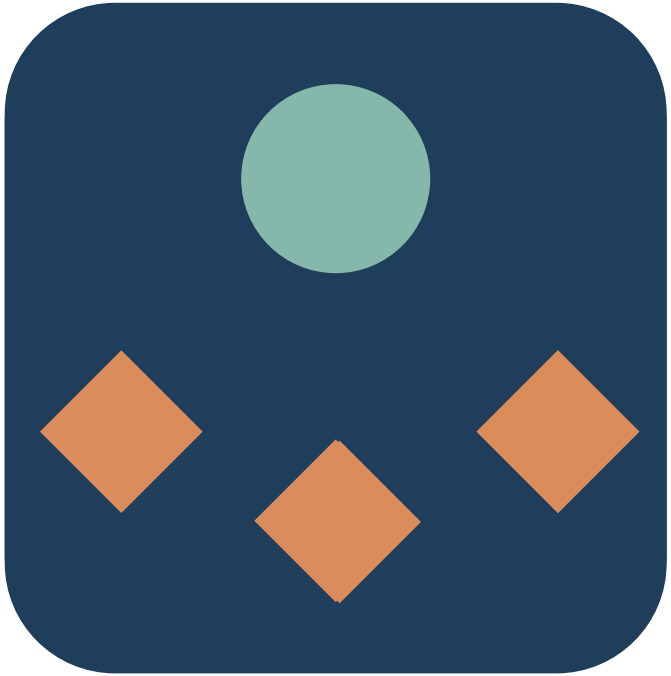
- Supervisor Architecture
- Hierarchical Architecture
- Network Architecture

MICAI 2025

INAOE

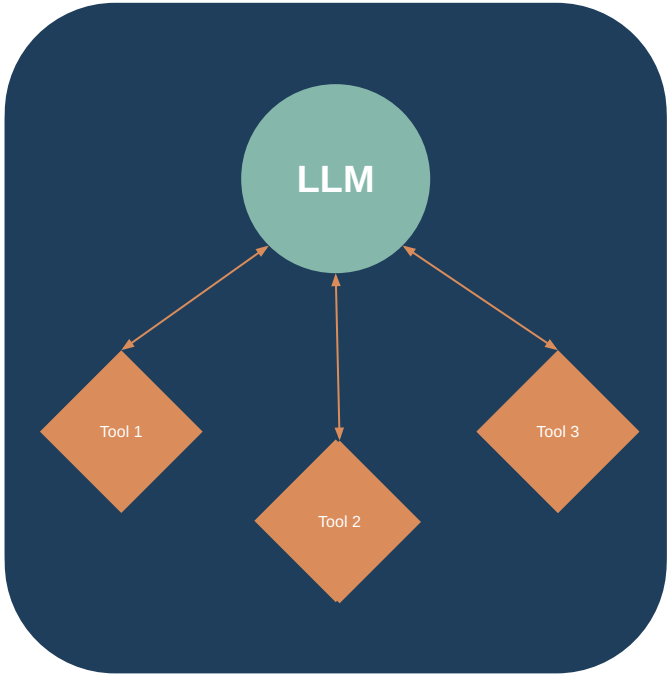
Multi-Agent Systems

Multi-Agent Systems (MAS)



- Single Agents
- Supervision
- Hierarchical
- Network
- Custom

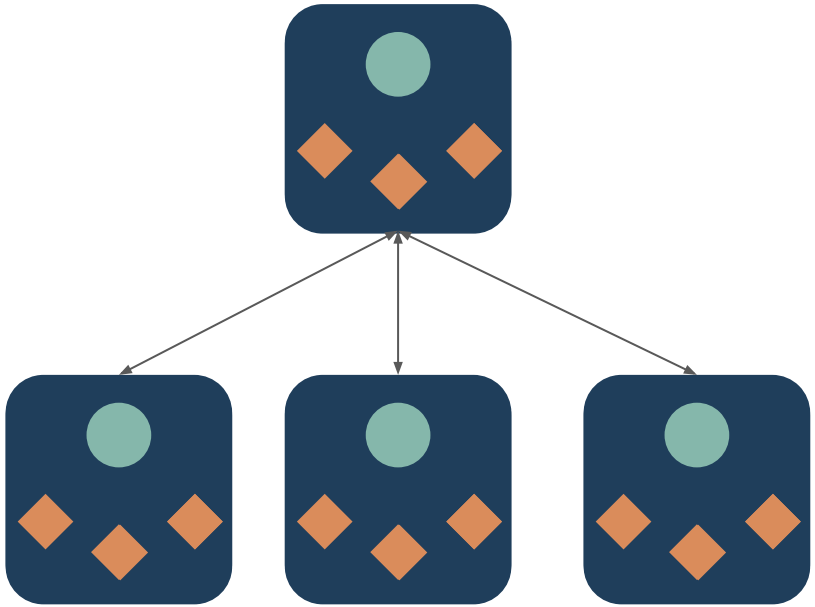
Single Agents



We only have one agent with a host of tools and connected to an LLM.

The agent then takes in a query and makes decisions on what tool to call using the LLM for reasoning.

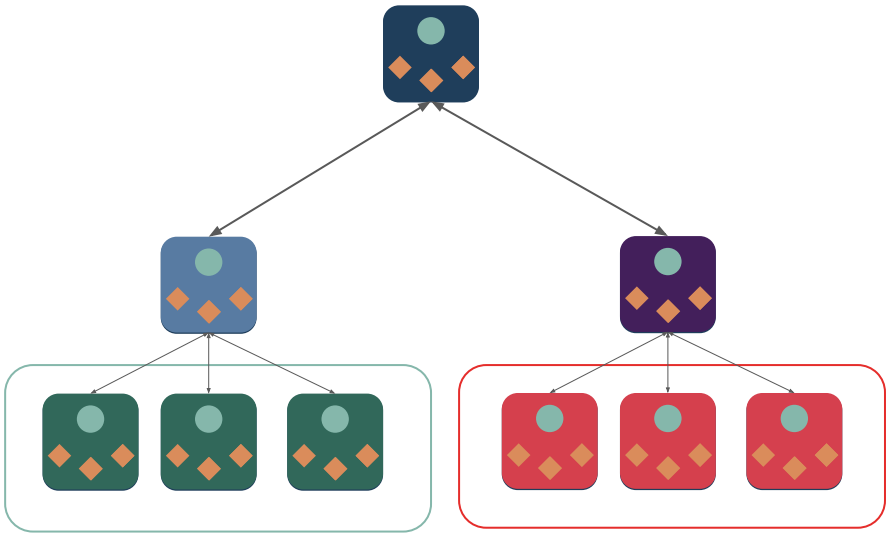
Supervisor Agent Architecture



The supervisor, basically, you can think of it as the orchestrator in a church choir.

It takes in queries, uses an LLM to reason over it and decide on what agents to delegate a task to or a sub-task to.

Hierarchical Agent Architecture

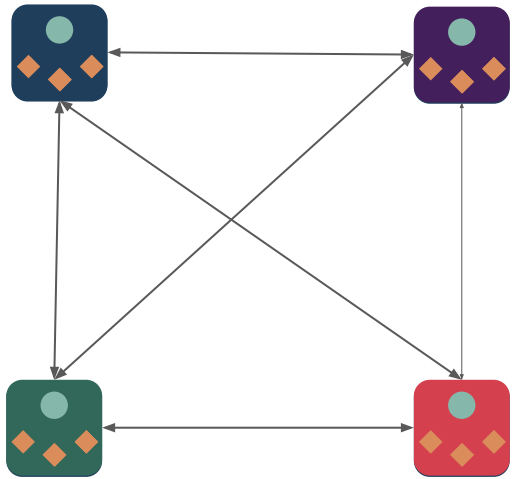


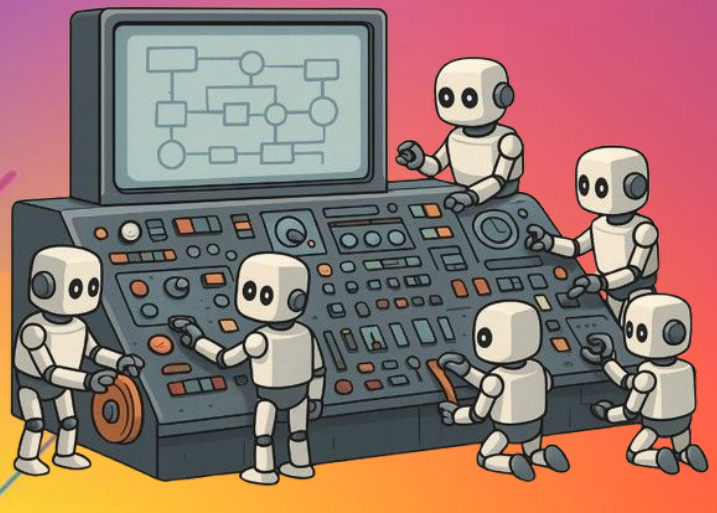
One common approach is create teams, where each team has its own supervisor, that supervisor is specialized in managing its own team of sub-agents.

Basically breaking the graph into subgraphs, each sub-graph has its supervisor and sub-agents under that supervisor.

Network Agent Architecture

Each agent can talk to any other agent in the whole collection.





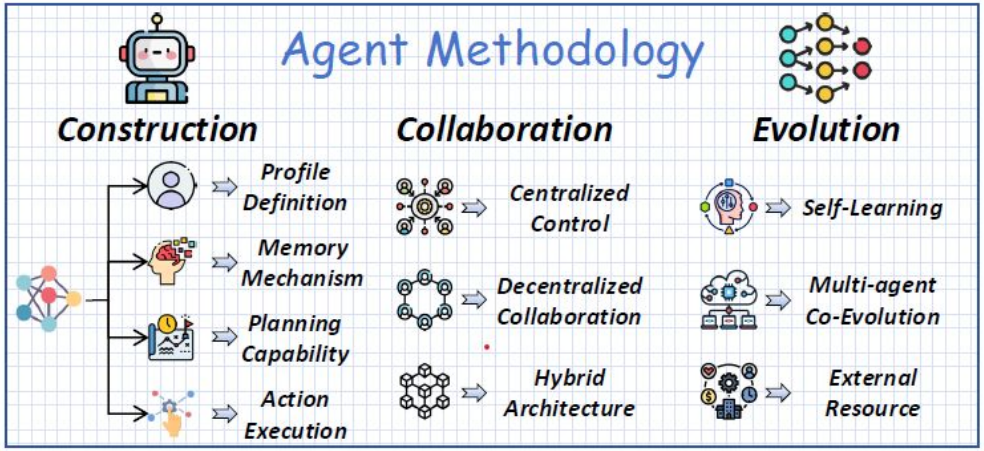
AI Agents using Small Language Models

MICAI 2025

INAOE

Agent Brain Decomposition

1. An agent interacts with an **environment**
2. **Perceiving** through any modality
3. Processes information and **plans** the next step
4. Takes **action** by executing tools



Junyu Luo et al. Large Language Model Agent: A Survey on Methodology, Applications and Challenges. https://arxiv.org/abs/2503.21460