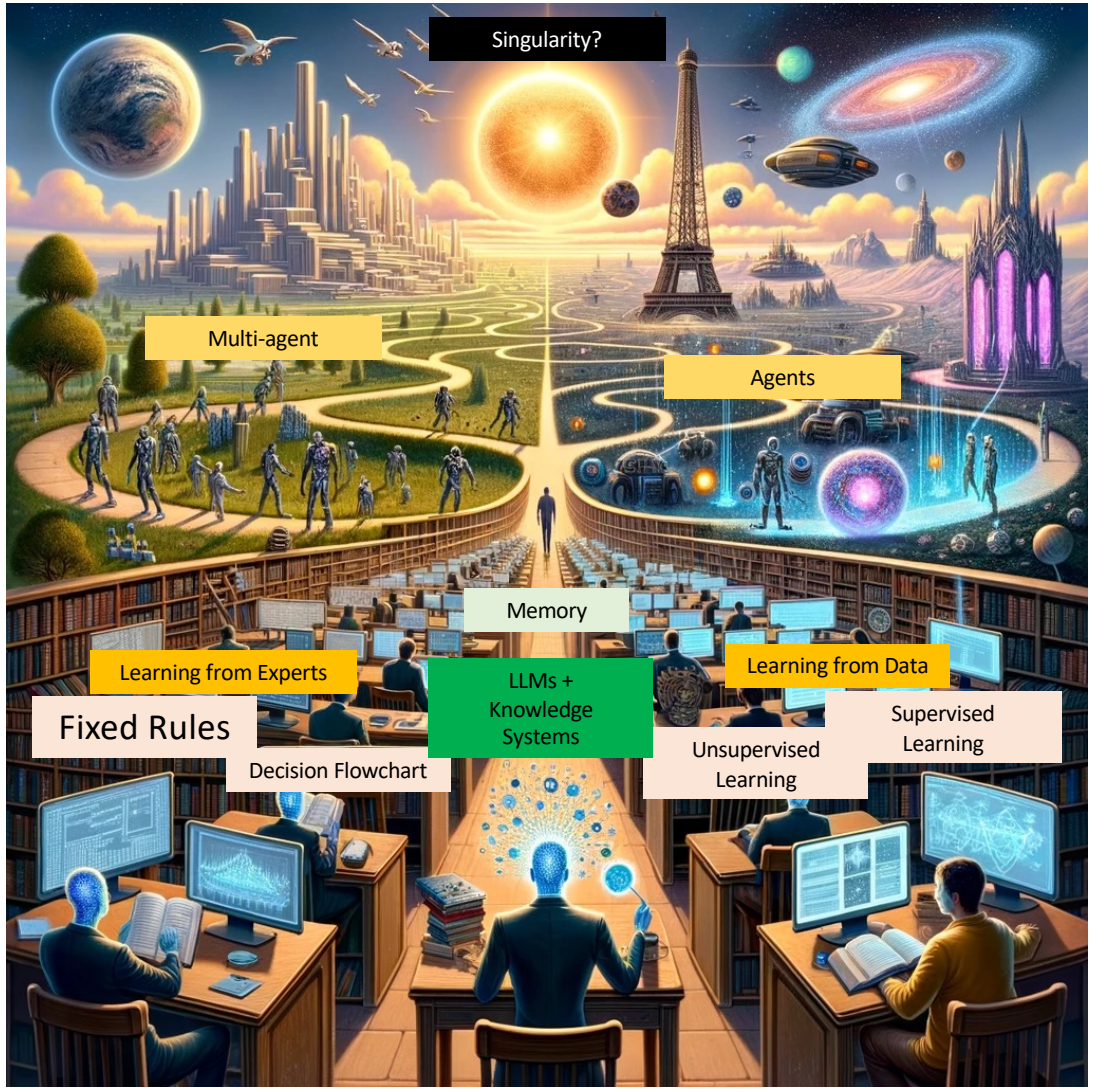


A Roadmap for AI

From Humble Beginnings to Imagining the Future

Presented by:
John Tan Chong Min

Key Takeaway

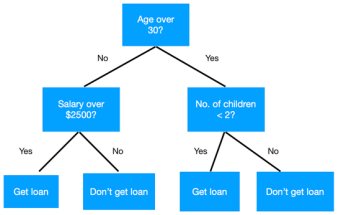
Artificial SuperIntelligence

“Multiple Sampling and Filtering”
within each agent, across agents

Summary for Part 1: Expert Systems, Supervised, Unsupervised (self-supervised) Learning

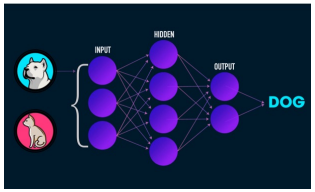
Expert Systems

Observations (Input-Output) → Interpreted by Human Experts → Codified into Rules → Build Model



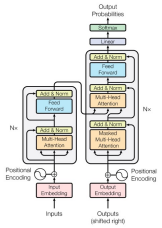
Supervised Learning

Observations (Input-Output) → Human-Labelled Data → Learn Model



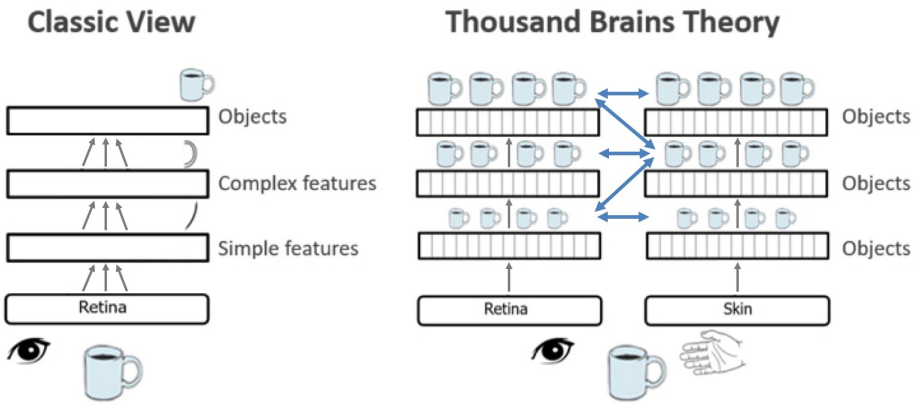
Unsupervised / Self-supervised Learning

Observations (Input-Output) → Learn Model

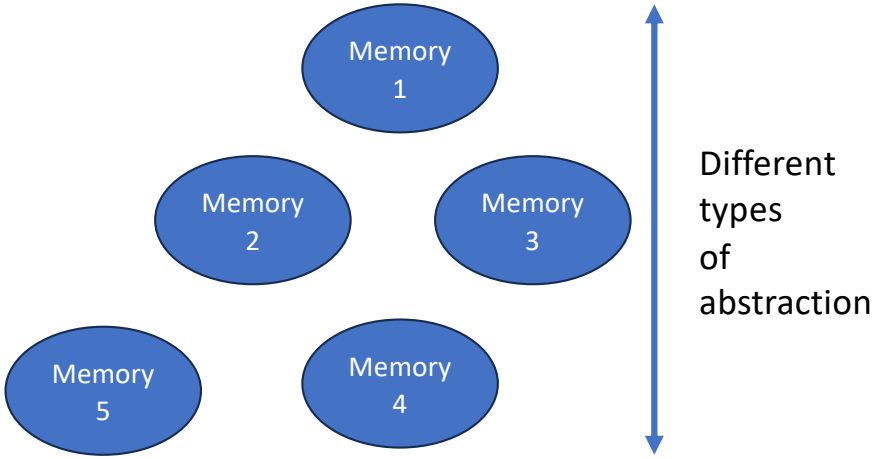


Summary for Part 2: Memory and Hierarchy

- All kinds of representations at various levels can be all put into one common **memory soup**
- Planning is done by pattern matching to the right level of abstraction in the **memory soup**
- **Latent spaces can be enriched through hierarchy**, though hierarchy may not be strictly needed



Thousand Brains Theory of Intelligence (Jeff Hawkins). Numenta.



"Memory Soup" Theory
(from John's AI Discord group)

Summary for Part 3: AGI/ASI is just “Multiple sampling and filtering”

- **Artificial General Intelligence (AGI)** can be defined as an AI being better than an average human across all tasks
  - Only for tasks with large amounts of training data – we have yet to achieve this for locomotion / robotic tasks
  - Memory and goal-directed learning will help
- **Artificial Super Intelligence (ASI)** can be defined as an AI being better than the best human across all tasks
  - Self-improvement to a limited extent can be done with multiple agents
  - Each agent can sample multiple possibilities and choose the best one
  - Each population can sample the experiences of multiple agents and choose the best one

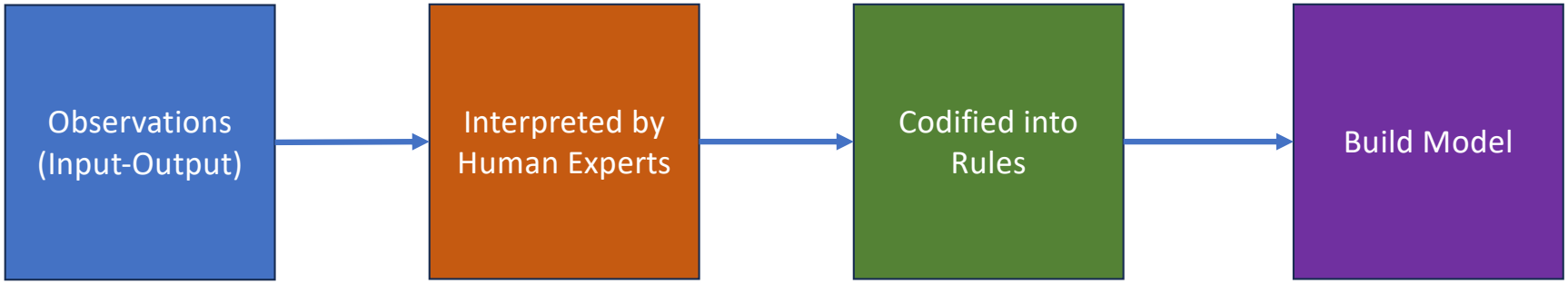
ASI has the advantage of being
able to simulate multiple futures all the time

Part 1

Overview

- Fixed Rules: Expert Systems
- Flexible: Data-Driven
  - Supervised Learning (Learning from data with human labels)
  - Unsupervised Learning (Learning from data without human labels)
- Merging Fixed + Flexible

- Towards the Future
  - Memory for Fast Learning
  - Hierarchical Prediction
  - Multi-agent systems
  - Collective Intelligence



Expert Systems

What if we can codify out the rules and just get a system to follow them?

Key Motivation

- Easy to visualize
- Easy to interpret
- Process-flow framework is easy to understand
- *“What if we can automate the expert?”*

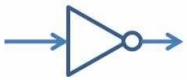
Logical Rules

Logic Operators

Name	Symbol	Example	Meaning
negation	$\neg$	$\neg a$	not a
conjunction	$\cap$	$a \cap b$	a and b
disjunction	$\cup$	$a \cup b$	a or b
equivalence	$\equiv$	$a \equiv b$	a is equivalent to b
implication	$\supset$ $\subset$	$a \supset b$ $a \subset b$	a implies b b implies a
universal	$\forall$ X.P		For all X, P is true
existential	$\exists$X.P		There exists a value of X such that P is true

NOT

x	F
0	1
1	0



AND

x	y	F
0	0	0
0	1	0
1	0	0
1	1	1

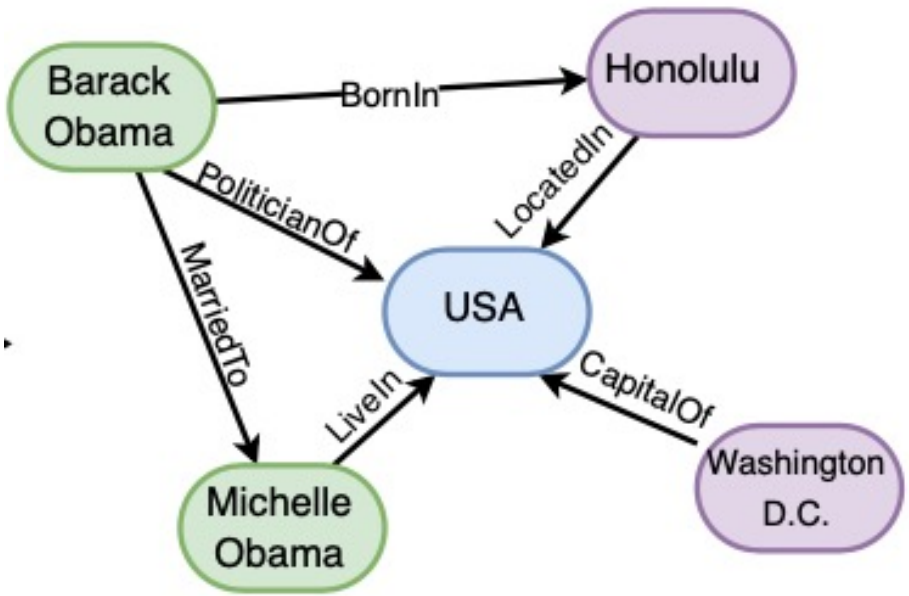
OR

x	y	F
0	0	0
0	1	1
1	0	1
1	1	1

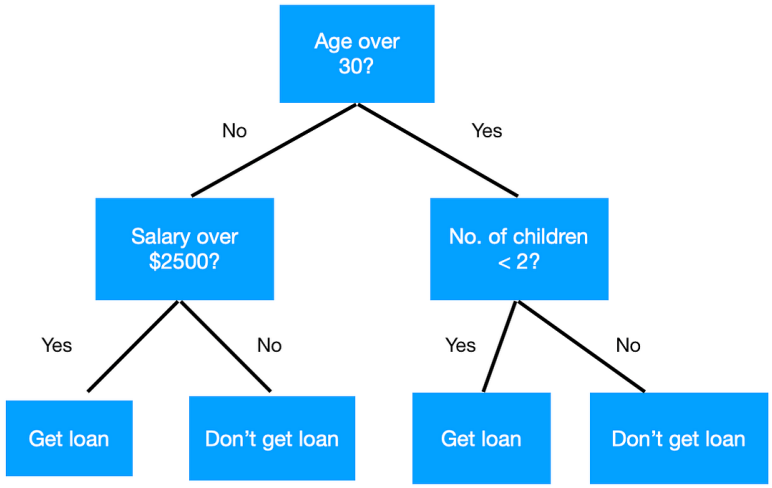
XOR

x	y	F
0	0	0
0	1	1
1	0	1
1	1	0

Fixed Representations

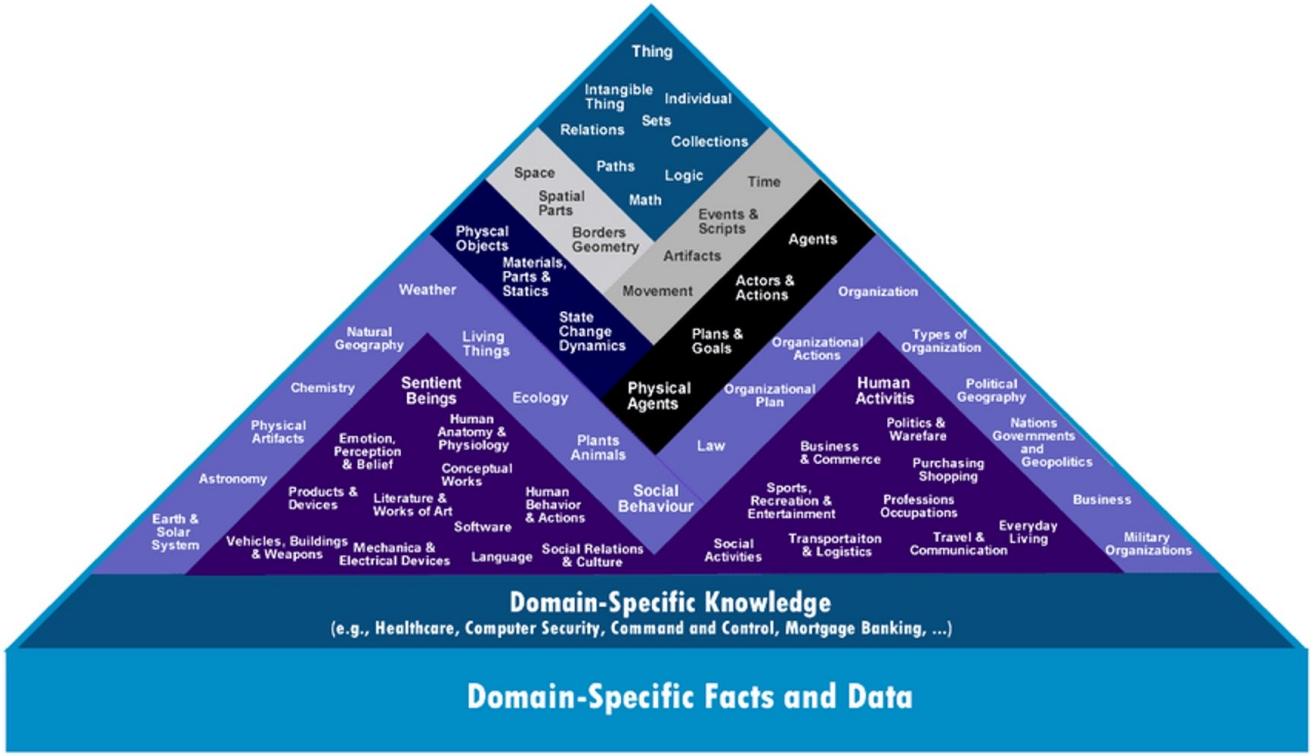


Knowledge Graphs



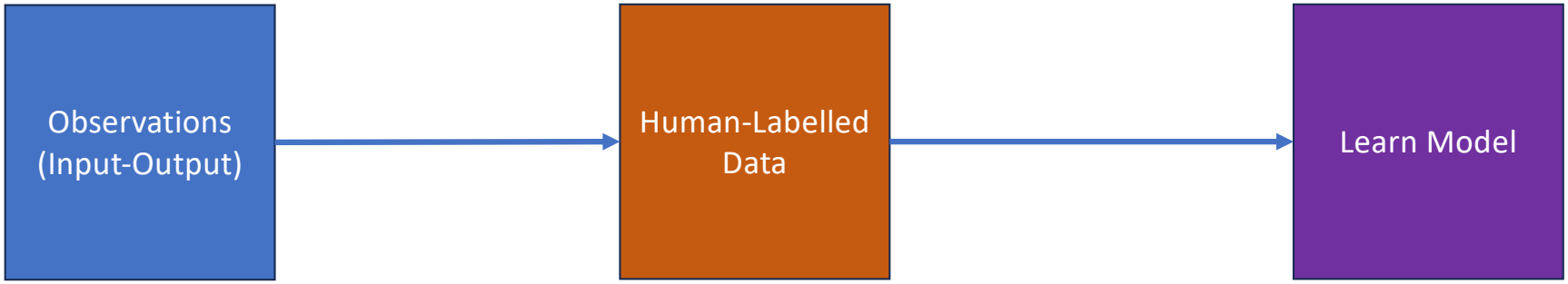
Decision Trees

Fixed Ontology in Knowledge Graphs



Cyc Knowledge Base.

Computational Analysis on a Corpus of Political Satire Articles: A Theoretical and Experimental Study. Stingo et al. 2016.



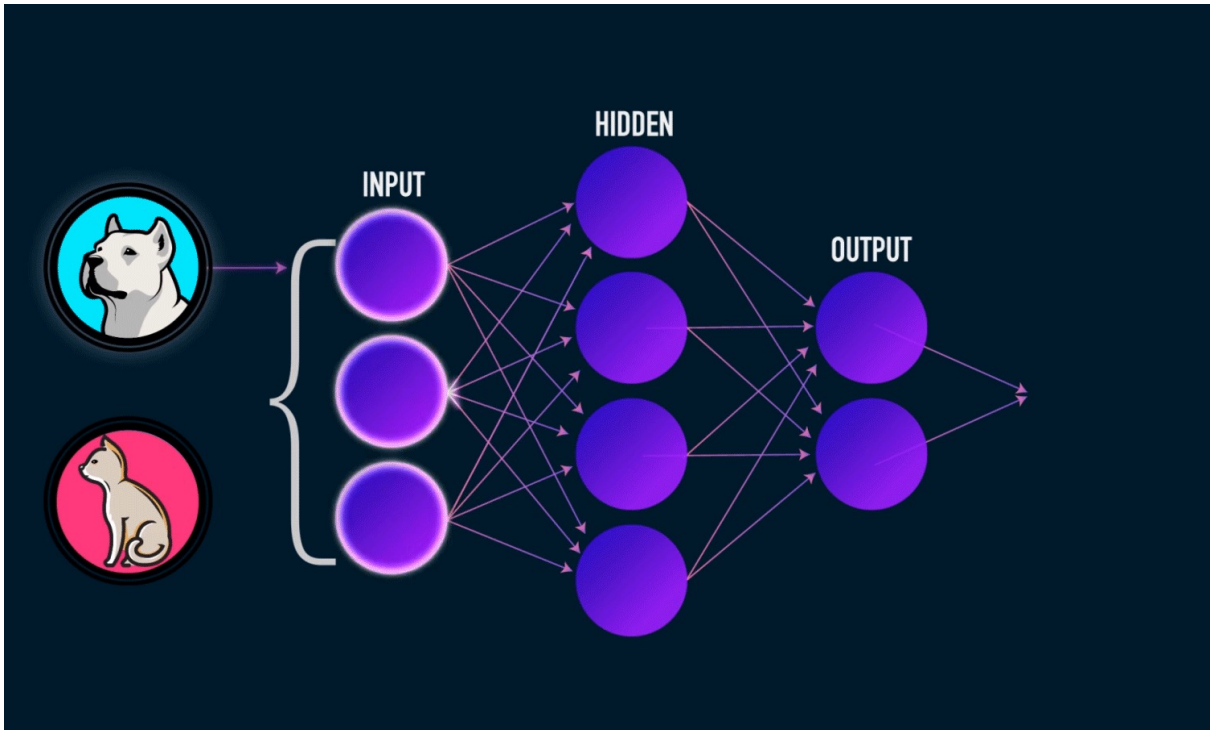
Supervised Learning

Can we get a system to learn from human-labelled data?

Key Motivation

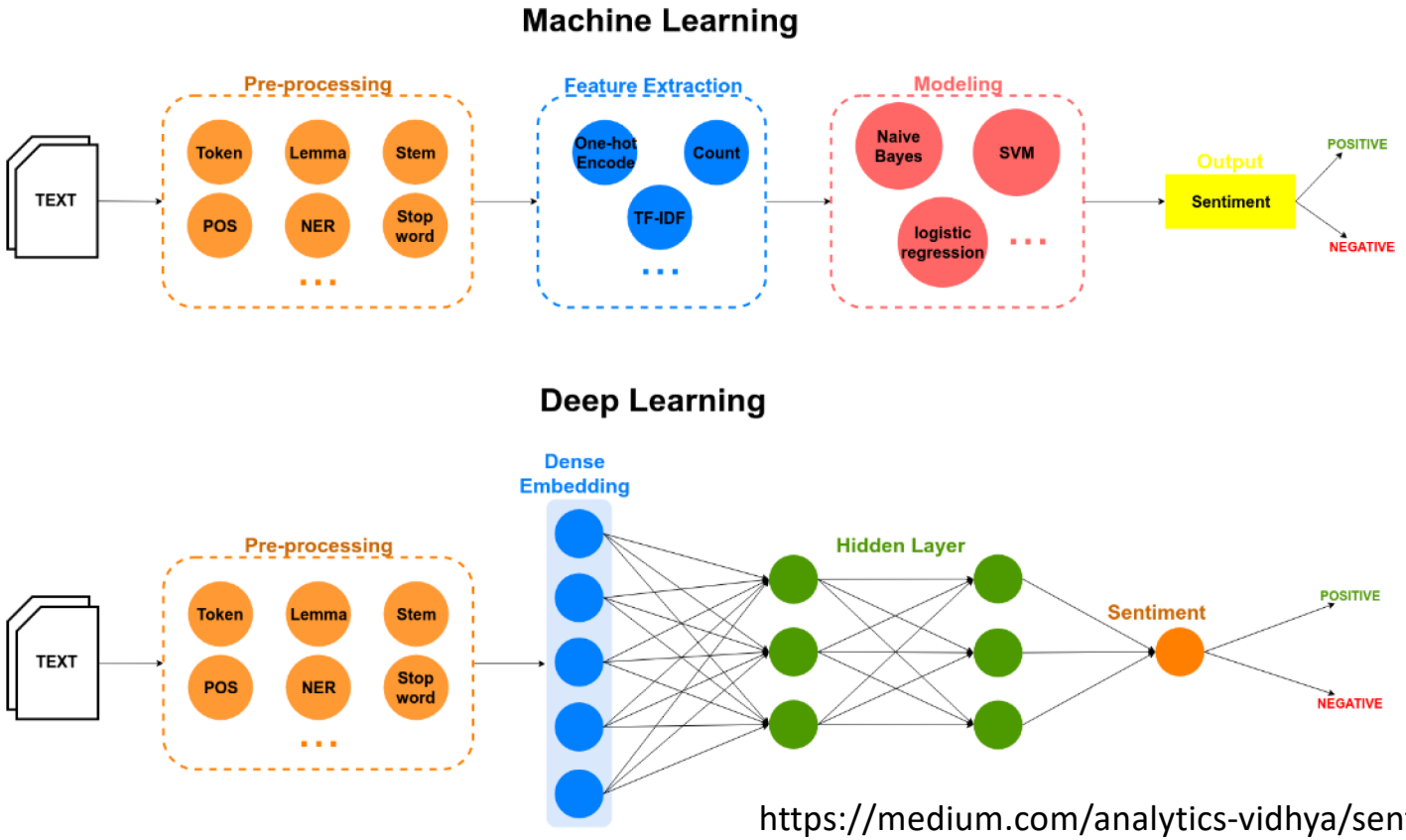
- Human expert's guidance can be expensive and time-intensive to obtain
- What if even human experts do not know how to codify into rules?
- Rules are extremely inflexible
- Can we learn from just human-labelled data alone?

Example: Neural Network-based Classifier



https://towardsdatascience.com/everything-you-need-to-know-about-neural-networks-and-backpropagation-machine-learning-made-easy-e5285bc2be3a

Example: Sentiment Analysis



https://medium.com/analytics-vidhya/sentiment-analysis-using-deep-learning-a416b230ca9a

Unsupervised Learning

Can we get a system to learn without human-labelled data?

Key Motivation

- **Human-labelled data** can be expensive and time-intensive to obtain
- Can we learn from just data in the wild?
- Just training on next-token prediction task can be useful for multiple arbitrary tasks

Transformers: Representation via Prediction

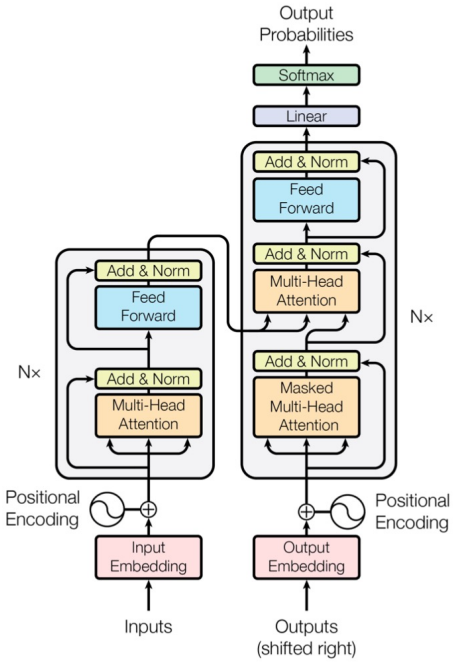
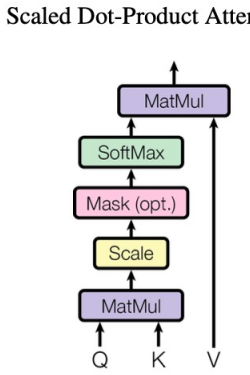


Figure 1: The Transformer - model architecture.



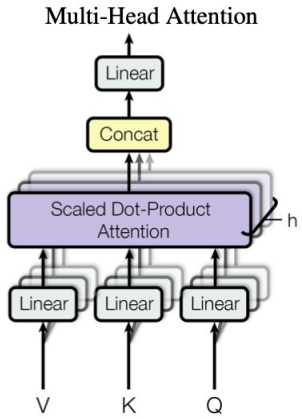
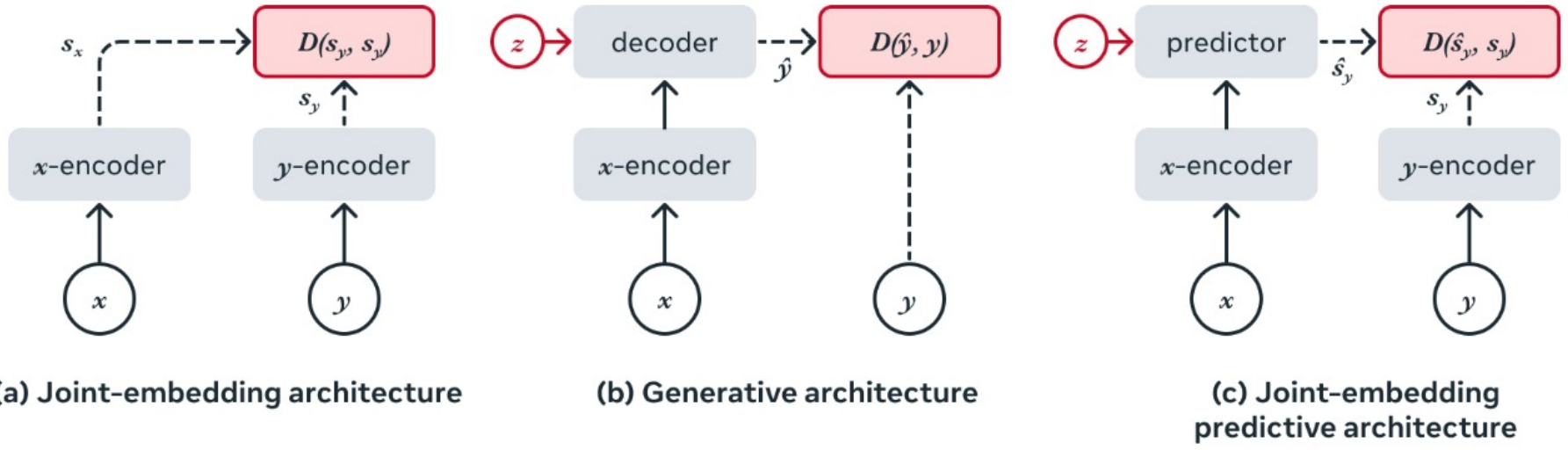


Figure 2: (left) Scaled Dot-Product Attention. (right) Multi-Head Attention consists of several attention layers running in parallel.

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

Taken from: Attention is all you need. Vaswani et al. 2017

Prediction in Latent Space is powerful



(a) Joint-embedding architecture

(b) Generative architecture

(c) Joint-embedding predictive architecture

e.g. Contrastive Language-Image Pre-training (CLIP)

Audio and text embeddings

e.g. Generation of next token in Transformers

e.g. Generation of patches of images in latent space based on overall context

Unsupervised learning for multi-modality?

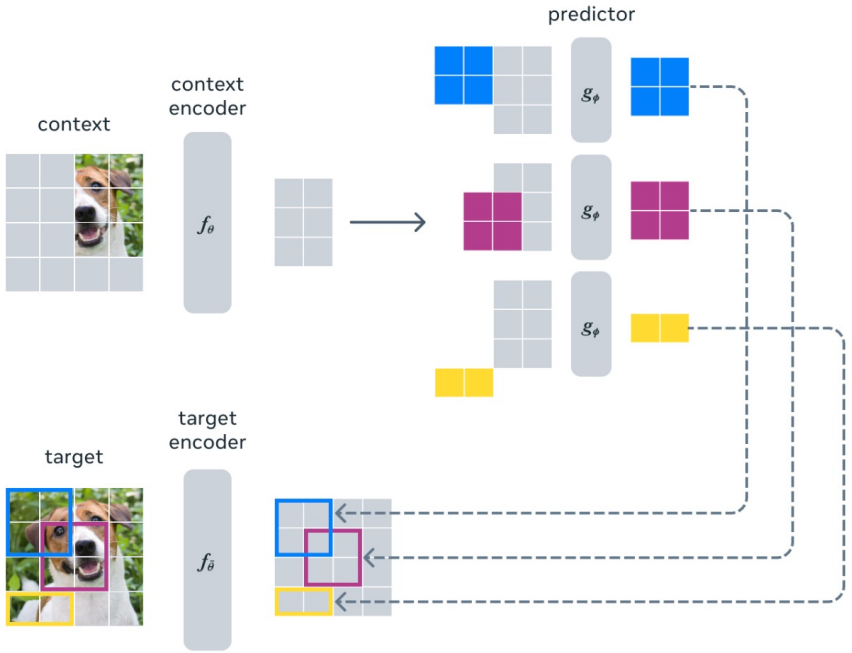
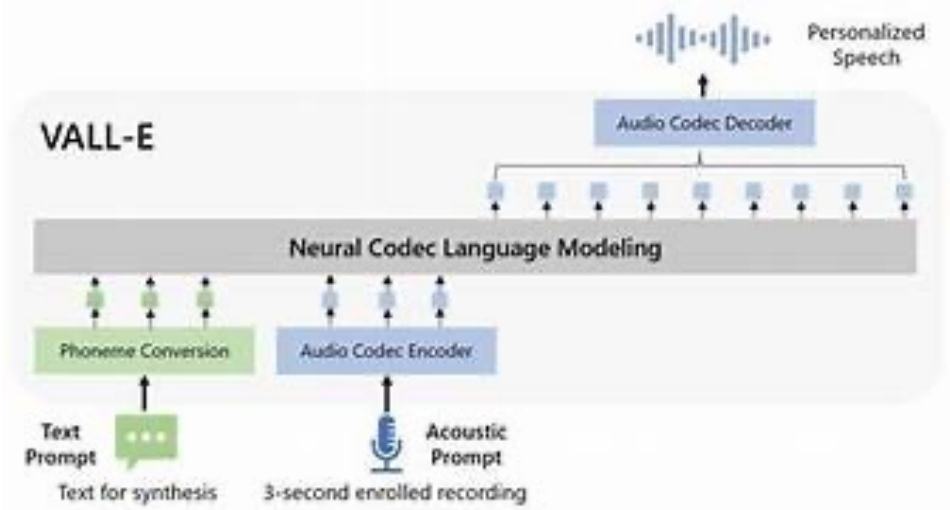


Image: I-JEPA



Audio: VALL-E

Unsupervised learning for reinforcement learning?

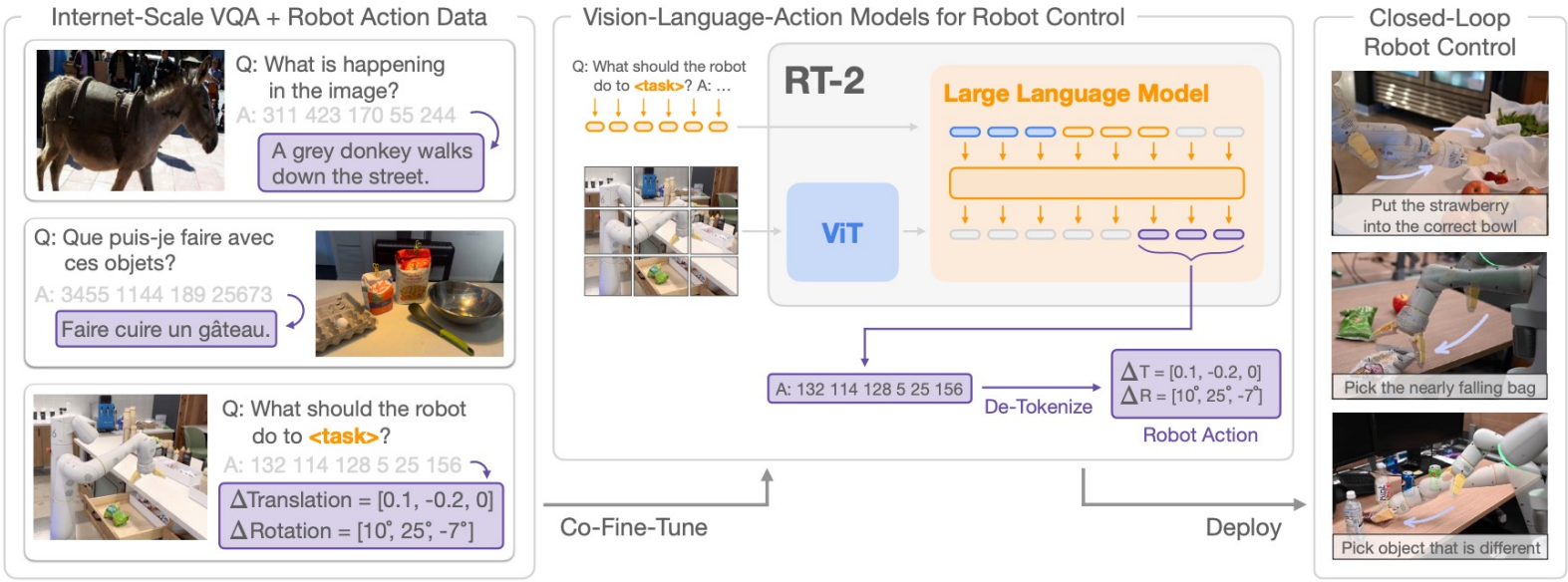
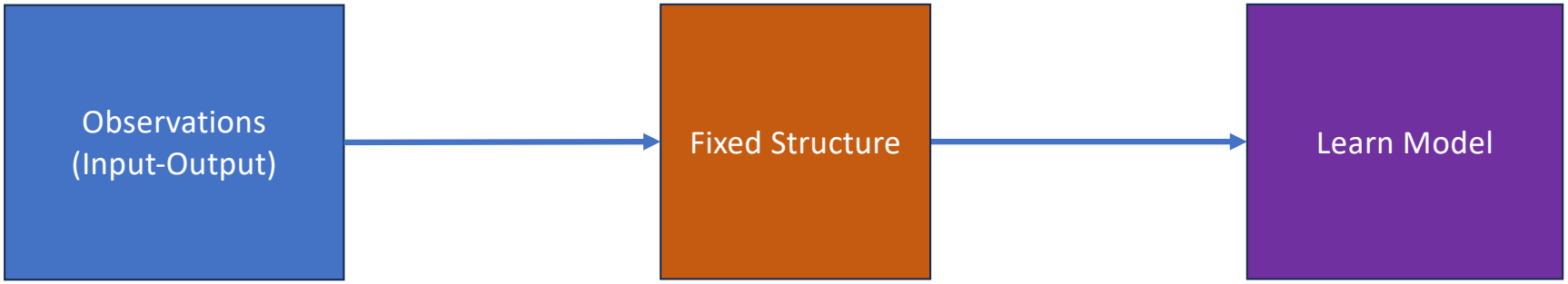


Figure 1 | RT-2 overview: we represent robot actions as another language, which can be cast into text tokens and trained together with Internet-scale vision-language datasets. During inference, the text tokens are de-tokenized into robot actions, enabling closed loop control. This allows us to leverage the backbone and pretraining of vision-language models in learning robotic policies, transferring some of their generalization, semantic understanding, and reasoning to robotic control. We demonstrate examples of RT-2 execution on the project website: robotics-transformer2.github.io.

Part 2



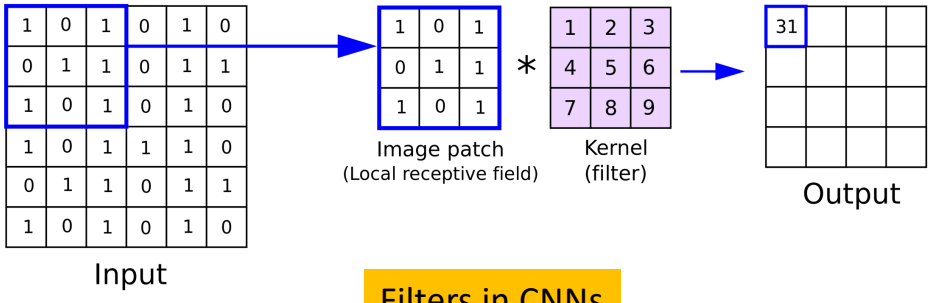
Fixed + Flexible

We want flexible learning, but we want some structure to ground it

Full End-to-End learning is not good

- Some structural bias is needed for faster learning
  - Filters in Convolutional Neural Networks
  - Token embeddings in LLMs

- Perhaps we now need to combine some earlier structural techniques used in **Expert Systems** to **Supervised / Unsupervised Systems**



Filters in CNNs

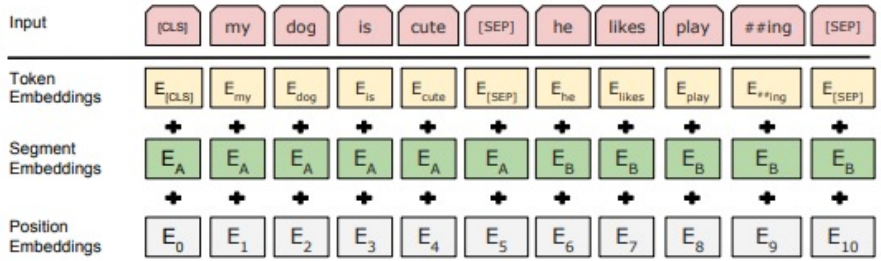


Figure 2: BERT input representation. The input embeddings is the sum of the token embeddings, the segmentation embeddings and the position embeddings.

Tokenisation in LLMs

LLMs vs Rule-based programs


```
# Print to console
print("Hello, World!")

# Request user input from command line
text = input()
```

- Customizable with zero-shot / few-shot prompting out of the box
- Performance may not be replicable
- Can do intent processing well, even for out of distribution cases

- Needs to be programmed extensively to perform a task
- Performance replicable
- Intent processing only based on what is programmed in

Towards the Future - Memory for Fast Learning

Context as Memory: Prompting for LLMs

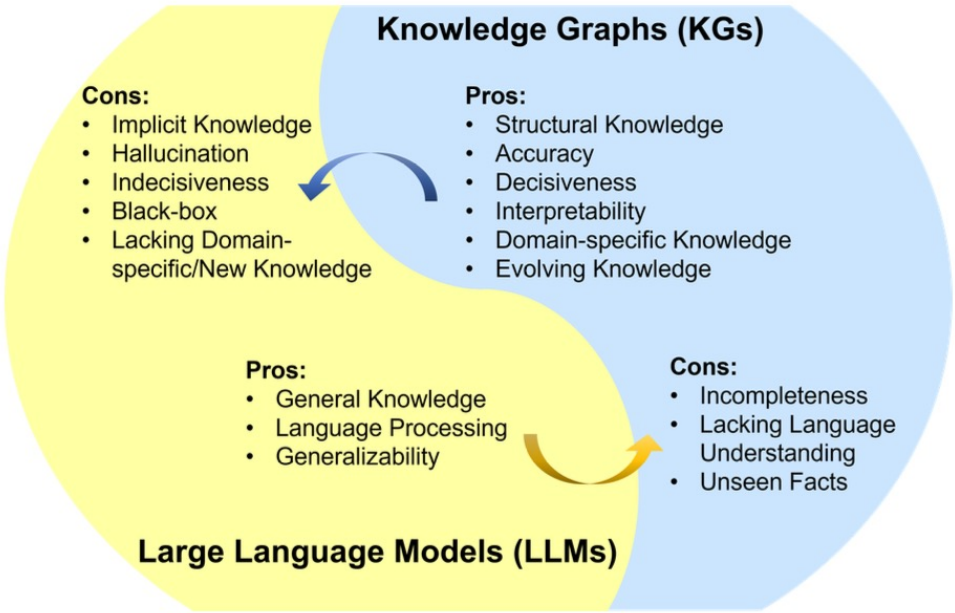
- Can be zero-shot prompted
  - You are a sentiment analysis tool …
- Can be few-shot prompted
  - These are a few examples: [Example A], [Example B], [Example C]
- Can be prompted sequentially
  - “Let’s think step by step” (Large Language Models are Zero-Shot Reasoners)
  - Broad to specific prompting to encourage better generation

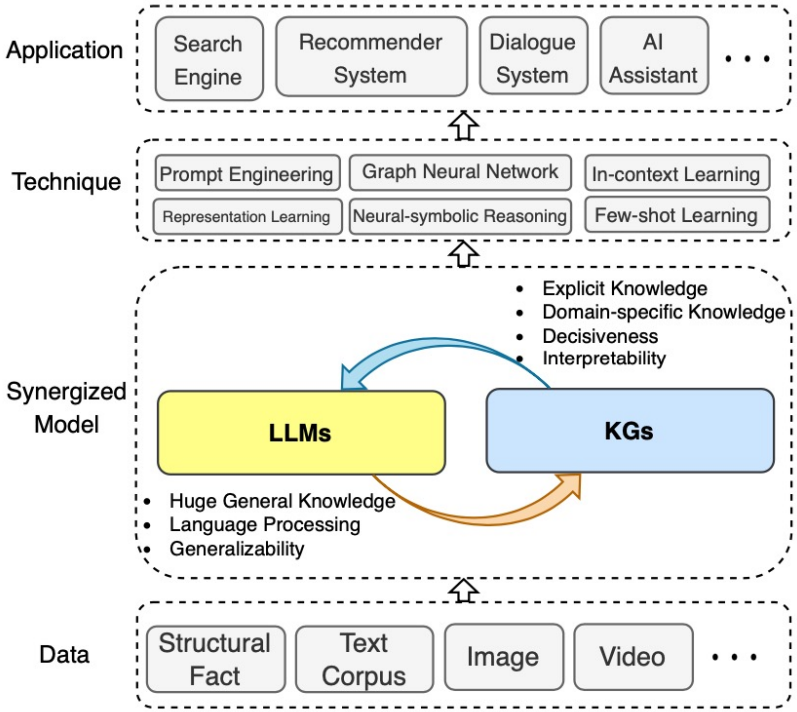
Context as Memory:
Retrieval Augmented Generation (RAG)

- <Context 1>
- <Context 2>
- <Context 3>

- <Query>

Knowledge Graphs and LLMs: Consistency in Grounding of LLM based on Memory in KG

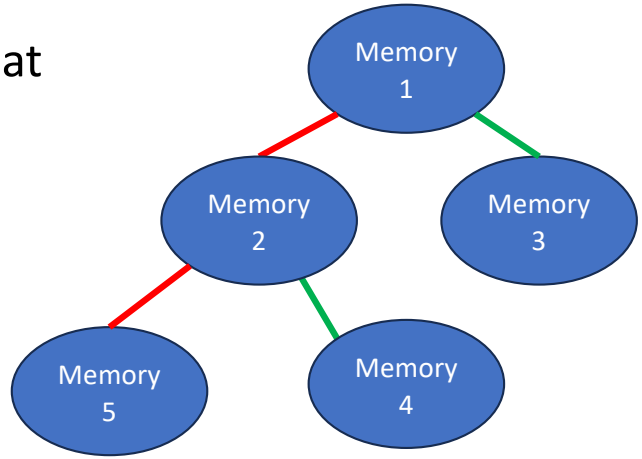




Unifying Large Language Models and Knowledge Graphs: A Roadmap. Pan et al. 2023

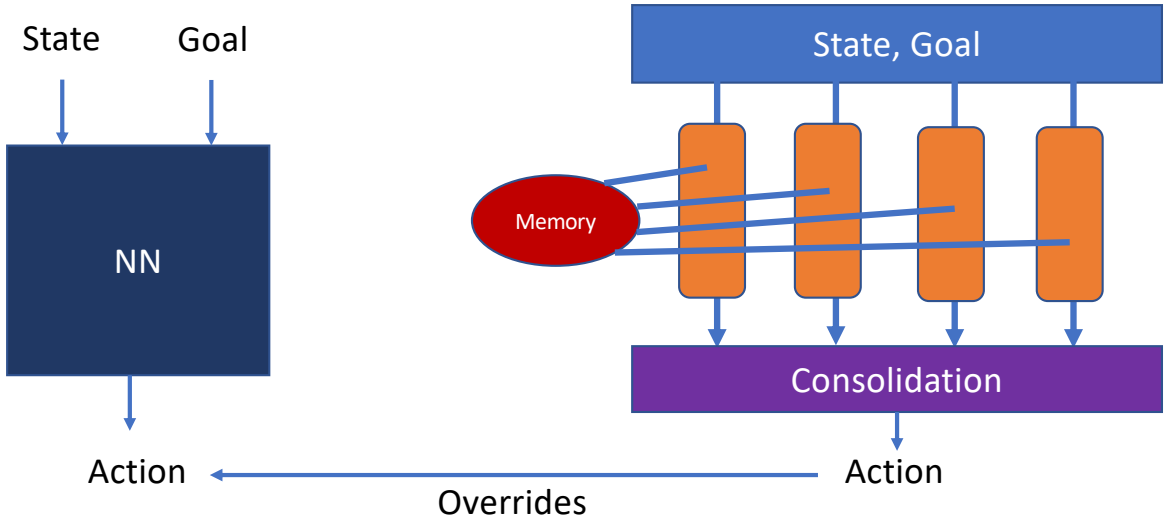
Chain of Memories – A New Kind of Knowledge Graph

- Perhaps rather than fixed ontology, we just store the memories in a **memory soup**
- Context-dependent links are form between memories by traversing the memory graph based on a specific context
- We jump from memory to memory as we retrieve what we need to and constantly search the **memory soup** until we get what we need
- Inference on the memories are done on the fly on an as-required basis



Two Networks – Fast and Slow

- Memory is important for fast adaptation before neural networks learn



Neural Networks: Fast retrieval, slow learning

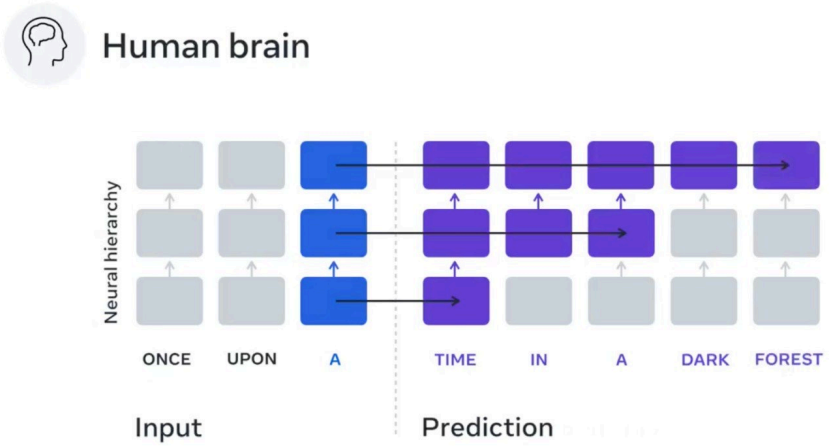
Memory: Slow retrieval, fast learning
(World Model planning as Memory Retrieval)

Learning, Fast and Slow: A Goal-Directed Memory-Based Approach for Dynamic Environments. John and Motani. 2023.

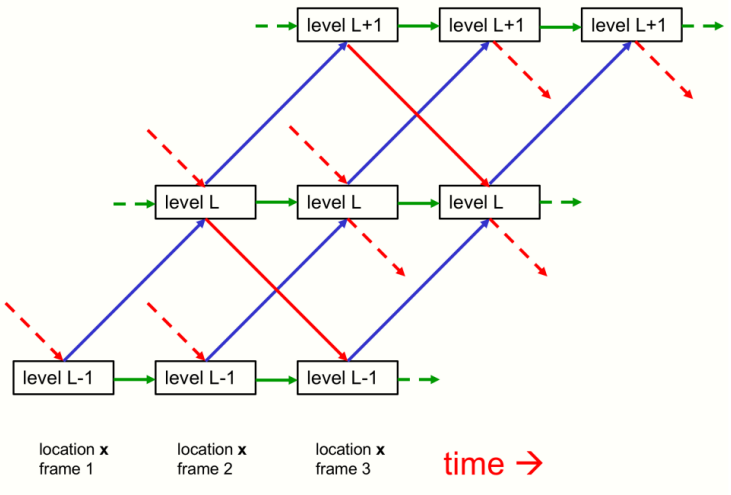
Towards the Future – Hierarchical Prediction?

Hierarchical Prediction is the future

- Hierarchical prediction of more than just next token, but broader prediction at higher levels
- Higher level prediction can be more abstract and less detailed than lower levels



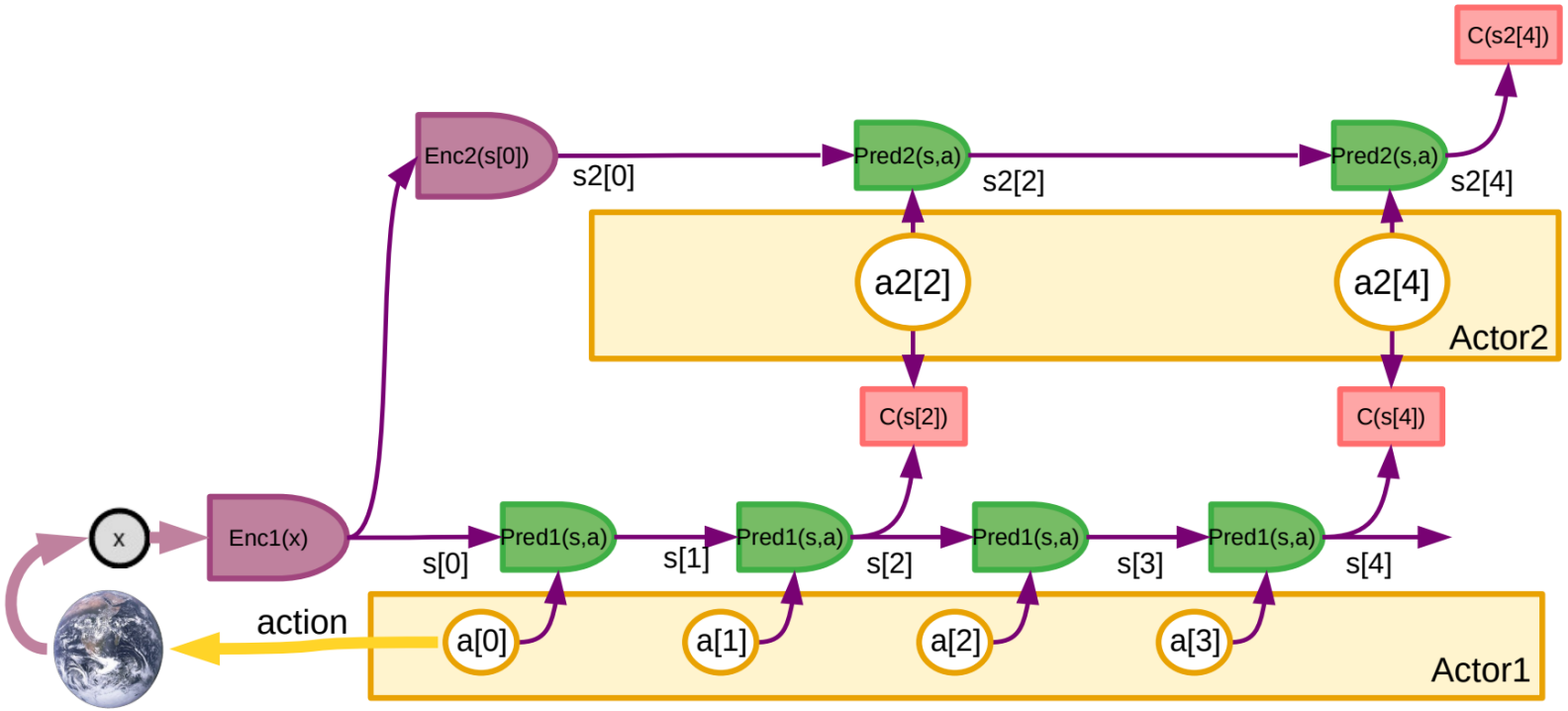
Evidence of a predictive coding hierarchy in the human brain listening to speech. Caucheteux. 2022. Nature Human Behaviour.



How to represent part-whole hierarchies in a neural network. Hinton. 2021.

Hierarchical JEPA

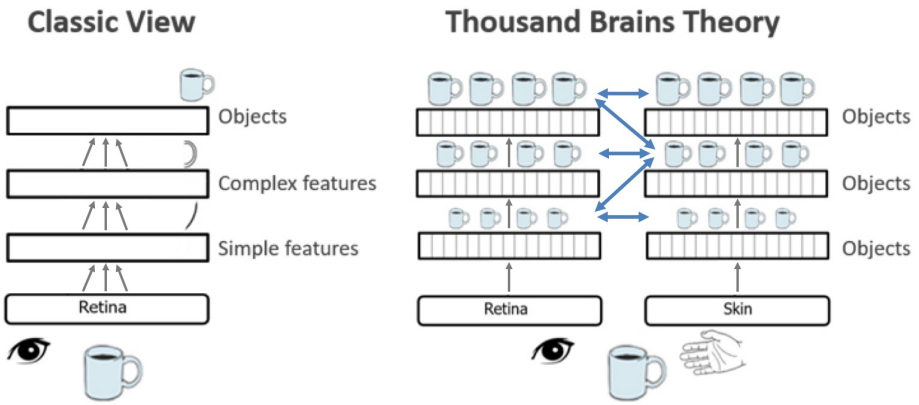
- Hierarchical prediction of actions from the highest level action to the lowest level action



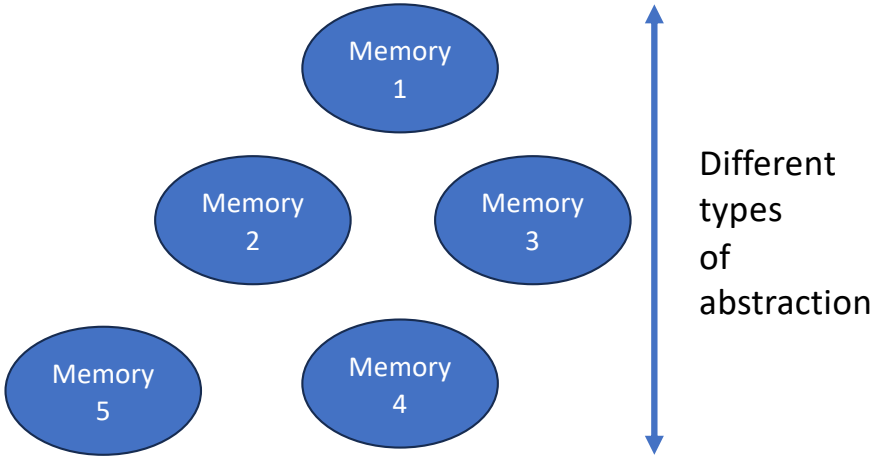
A Path towards Autonomous Machine Intelligence. Yann LeCun. 2022.

Or could it be a memory soup?

- All kinds of representations at various levels can be all put into one common **memory soup**
- Planning is done by pattern matching to the right level of abstraction in the **memory soup**



Thousand Brains Theory of Intelligence (Jeff Hawkins). Numenta.



"Memory Soup" Theory
(from John's AI Discord group)

Learning More from Experience: Reflection

Reflection helps to consolidate higher-order memories to make better decisions

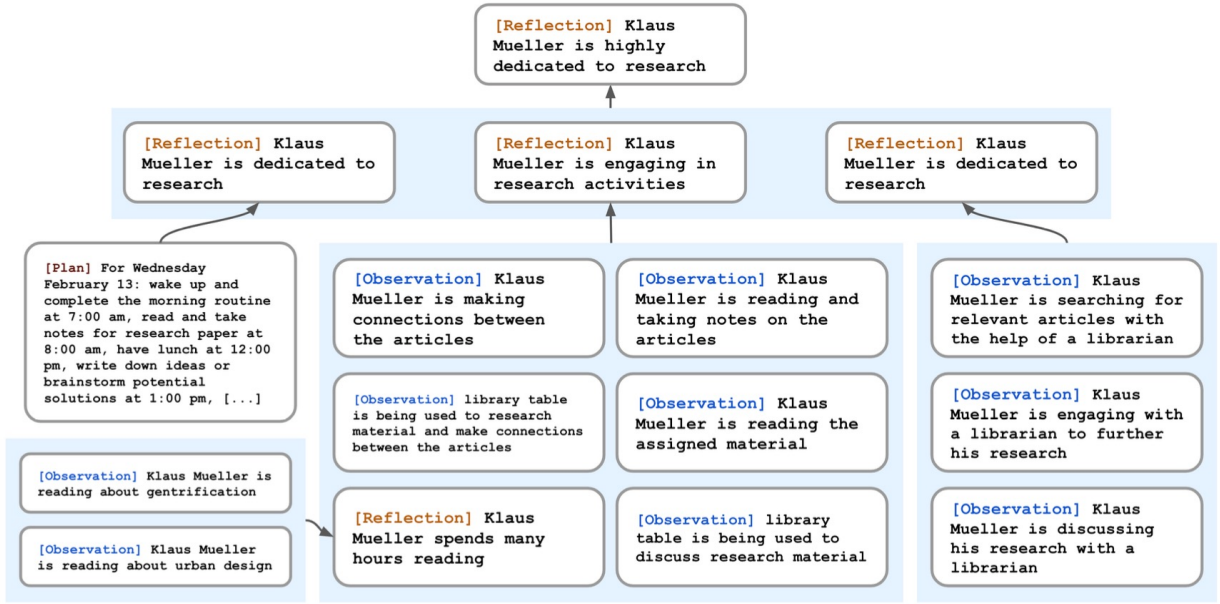


Figure 7: A reflection tree for Klaus Mueller. The agent's observations of the world, represented in the leaf nodes, are recursively synthesized to derive Klaus's self-notion that he is highly dedicated to his research.

Generative Agents. Park et al. 2023.

Part 3

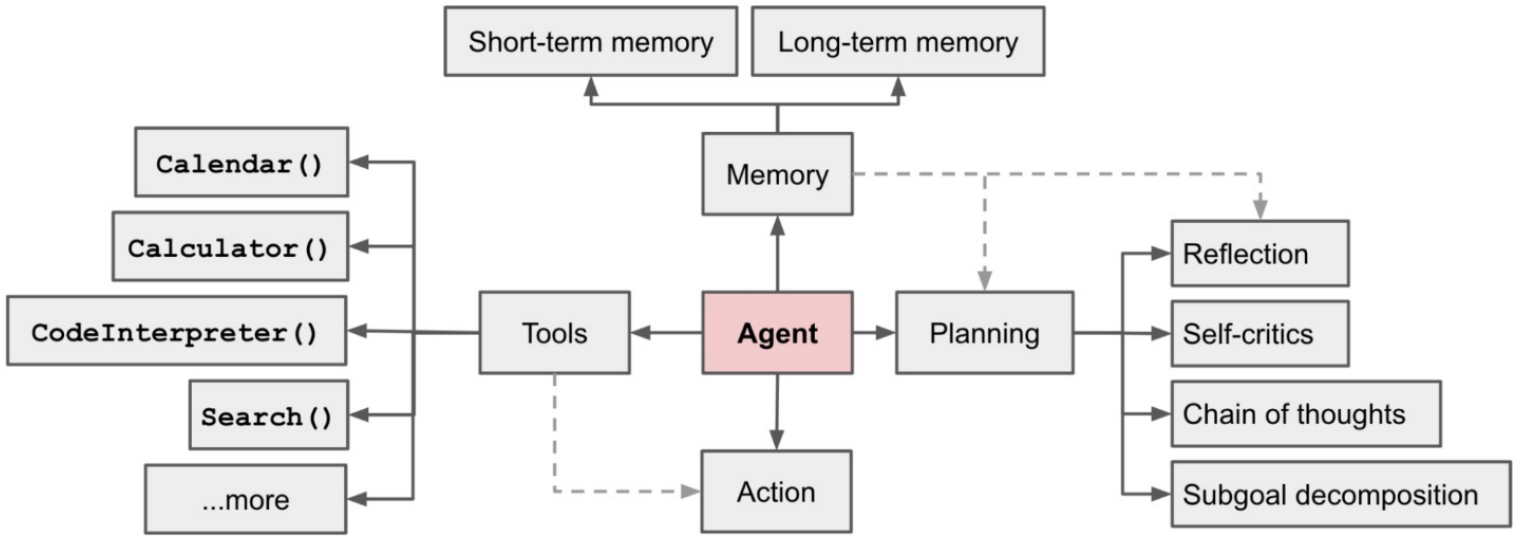
Towards Multi-Agent Systems

Agent Overview

Lilian Weng
@lilianweng

Agent = LLM + memory + planning skills + tool use

This is probably just a start of a new era :)



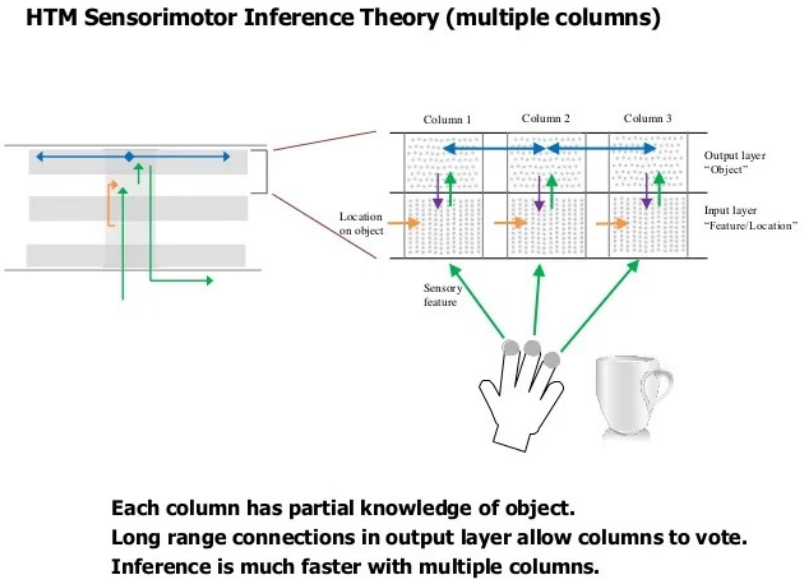
https://lilianweng.github.io/posts/2023-06-23-agent/

Example of Agents (basic): GPTs

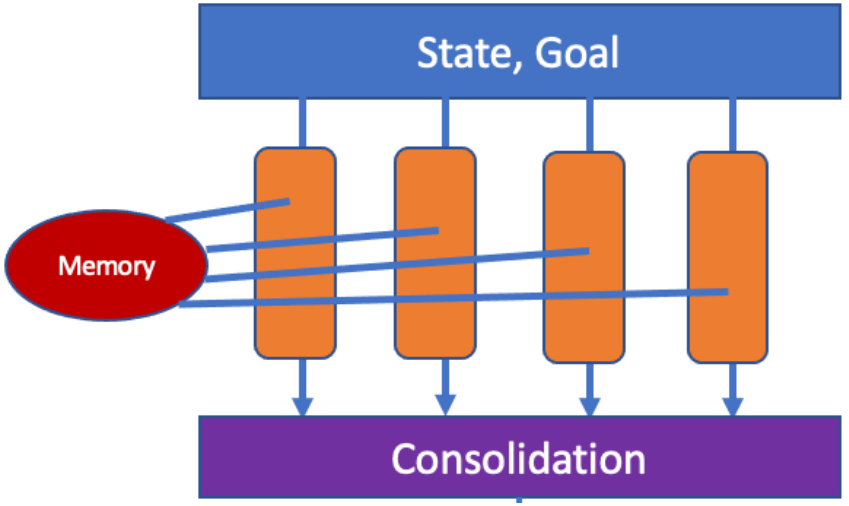
Incorporates some tools and memory
Missing out on memory learning, agent neural network learning from environment, multi-agent etc.

Multiple Agents within same system

- Provides a way of sampling to find out about complex observations



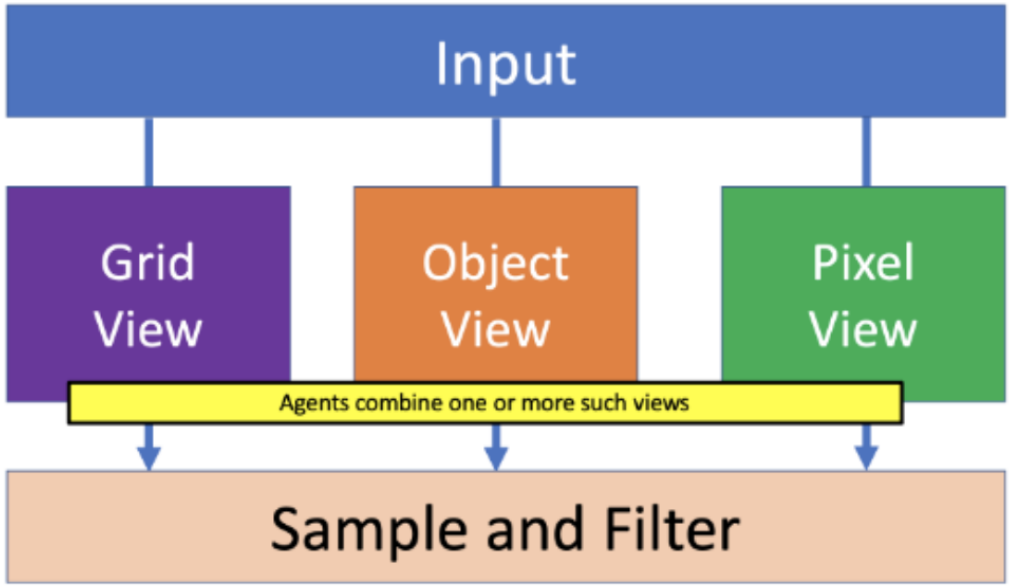
Hierarchical Temporal Memory (HTM). Numenta. 2016



Memory Retrieval Component in "Learning, Fast and Slow"

Multiple Specialized Agents within same System

- Each agent views the grid differently, like grid, object, or pixel level
- Can be multi-modal in the future
- Future work: Each agent can have different action spaces
- Generate multiple potential Python programs



LLMs as a System of Multiple Expert Agents. John and Motani. 2023.

Can skills be learned through environment?

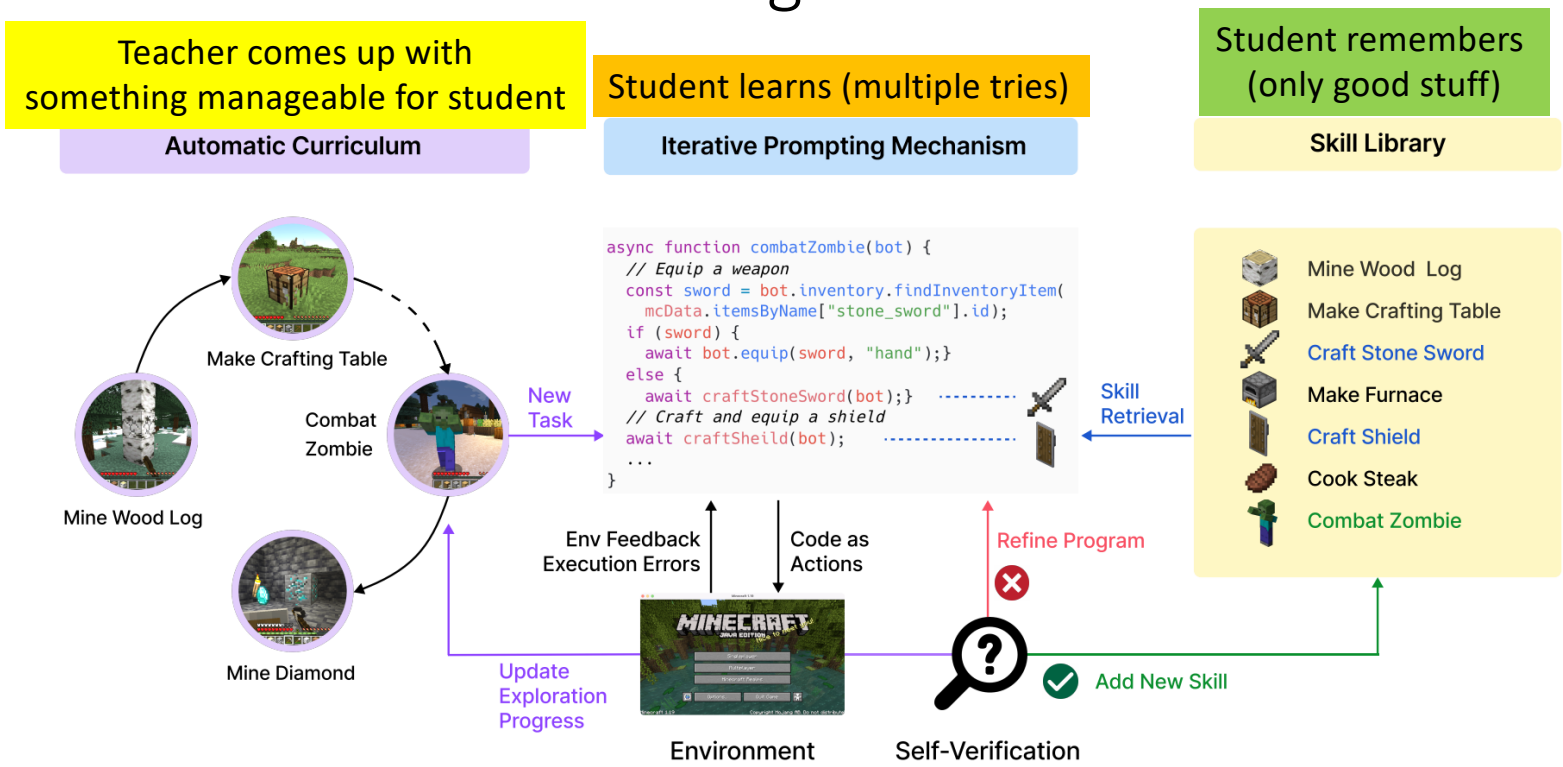
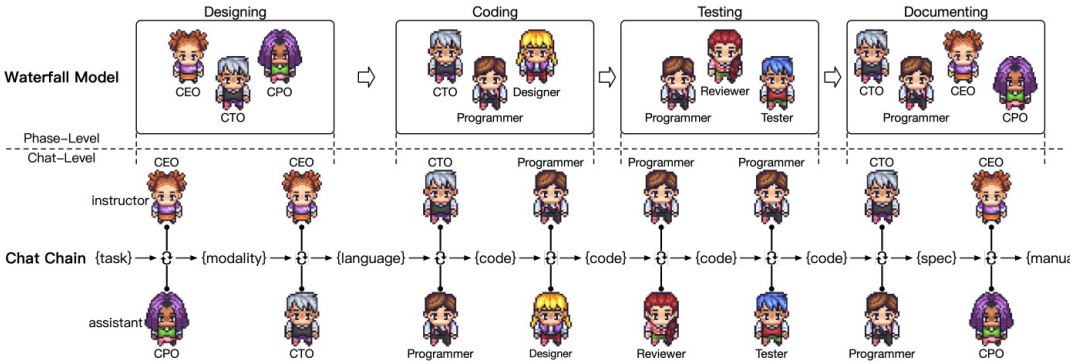


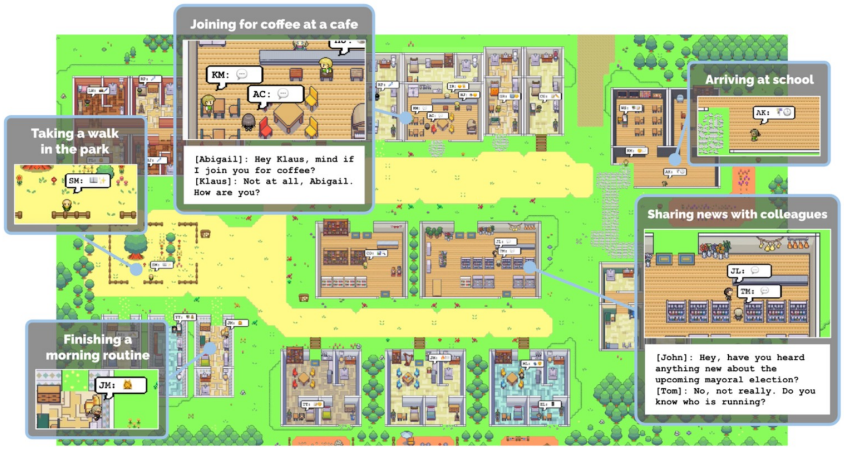
Figure 2: VOYAGER consists of three key components: an automatic curriculum for open-ended exploration, a skill library for increasingly complex behaviors, and an iterative prompting mechanism that uses code as action space.

Voyager. Wang et al. 2023.

Collective Intelligence



ChatDev. Qian et al. 2023.



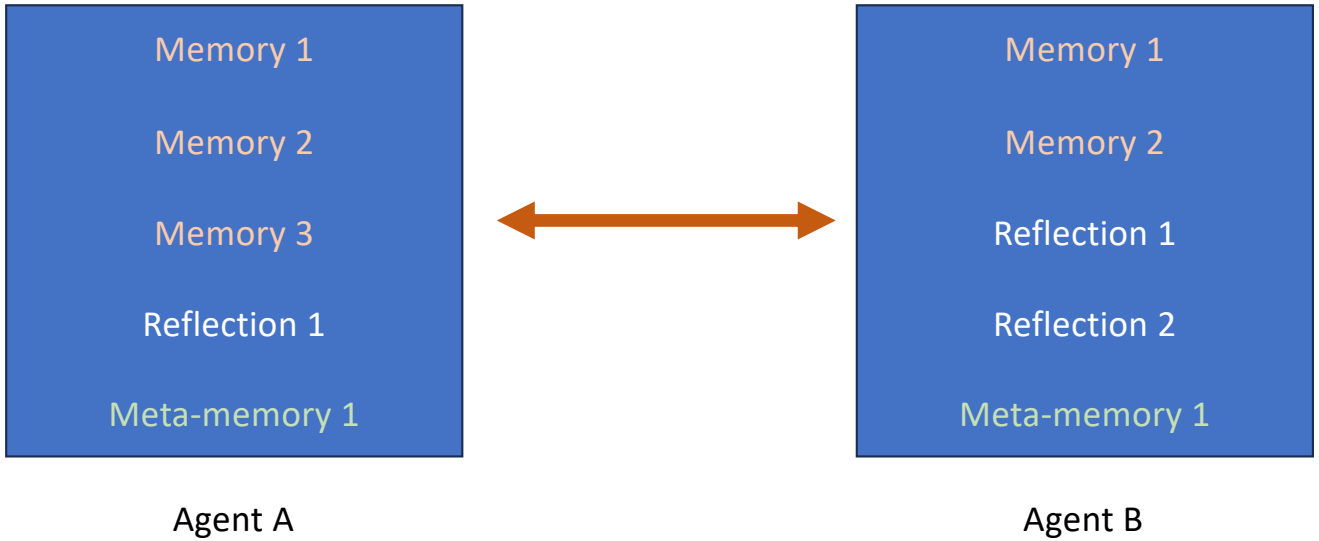
Generative Agents. Park et al. 2023.

- Sharing of knowledge among multiple agents
- Interaction amongst various agents help to shape the whole ecosystem

Knowledge Sharing between Agents

- How much knowledge to share?

- Would sharing of too much information be bad for agents?



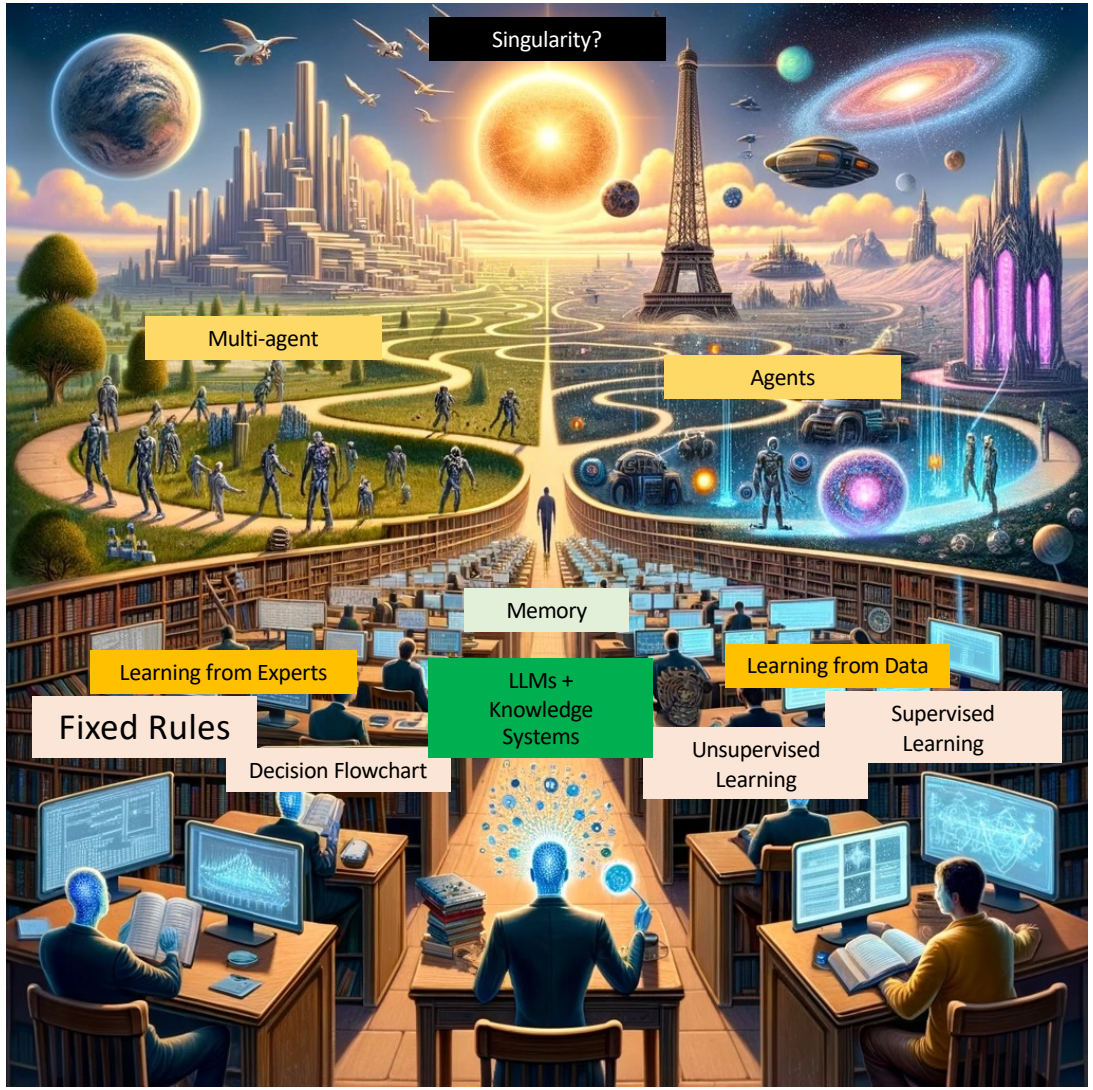
Intelligence via Multiple Populations

- Perhaps one population itself is not enough
- We need more populations and more simulations of them
- Choose the best one for performance
- Helps to “optimise” in changing environments

Can AGI/ASI be achieved?

- **Artificial General Intelligence (AGI)** can be defined as an AI being better than an average human across all tasks
  - Only for tasks with large amounts of training data – we have yet to achieve this for locomotion / robotic tasks
  - Memory and goal-directed learning will help
- **Artificial Super Intelligence (ASI)** can be defined as an AI being better than the best human across all tasks
  - Self-improvement to a limited extent can be done with multiple agents

ASI has the advantage of being
able to simulate multiple futures all the time



Can the singularity be reached?

- This question is essentially asking if an ASI system will be able to continually self-improve indefinitely
- **My answer: No**
- Reasons:
  - Exponential increase in search space for strict improvement
  - Dynamic environment changes means relearning is necessary most of the time -> continuous self-improvement is unlikely

Discussion

Questions to Ponder

- **There can only be so much data to learn from that is good**. After that, the system needs to learn to improve on its own. How is that done right now? Can we do better?
- Will compute and larger models solve the problem of self-improvement?
- Should we do planning based on hierarchy? If so, how can hierarchy be learned/constructed? Otherwise, how else to plan?
- Currently, we only do unsupervised learning well for textual data well. How could we do unsupervised learning for vision, motion etc. as well?
- How can we simulate multiple environments if we do not have an accurate enough model of the world?