Natural Language Processing: Introduction

Seung-Hoon Na

2024.3.5

Contents

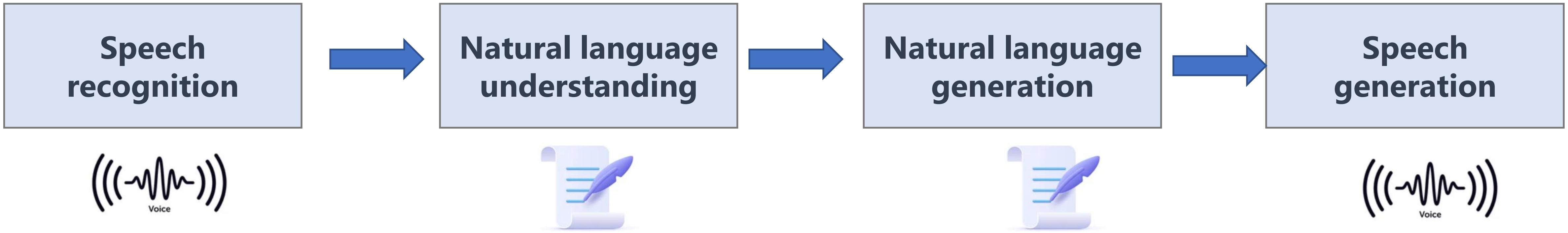
- Introduction
  - Goal & Tasks & Applications
  - Related disciplines
    - Artificial intelligence, Machine learning, Linguistics, etc.

- Natural language processing: Methods
  - Methods
    - Rule-based approach: ELIZA
    - Statistical method: HMM, PCFG
    - Deep learning: Word2vec, RNN, LSTM, Transformer
    - Pretrained language model: BERT
    - Large language model: GPT3
    - Human-aligned LLM: ChatGPT
    - LLM for X: Multimodal & Robotics, Others

- Course schedule

Natural Language Processing

- Goal
  - Develop computational models & algorithms to understand and generate human language
- Main tasks & components
  - Speech recognition
  - Natural language understanding
    - Information extraction, sentiment analysis, document classification, etc.
  - Natural language generation
    - Machine translation, summarization, dialogue/chatbot system
  - Speech generation (Text-to-speech)



Natural Language Processing

- Applications
  - Chatbot
    - More advanced and natural human-computer interface
  - Machine translation
    - Translate text or speech from one language to another.
  - Information retrieval & question answering
    - Given a query or a question, retrieve a relevant text and generate an answer
  - Etc.
- Domain-specific applications (NLP+X)
  - Legal NLP
    - Analyzing the complex, structured language found in legal documents, including statutes, contracts, case law, and regulatory materials
    - E.g.) Predictive Legal Analytics or Legal Judgment Prediction & Legal Language Modeling & Legal Information Retrieval
  - Patent NLP
    - The application of NLP techniques to patent documents and intellectual property (IP) texts
  - Medical NLP
    - Interpreting, analyzing, and generating medical and clinical text or speech data
    - E.g.) Health Record (EHR) Analysis, Clinical decision support
  - Financial NLP
    - Analyzing textual data within the finance and investment industries.
    - E.g.) Market Prediction and Trading Strategies
  - Etc.

Natural Language Processing

- Multimodal Extension (Text ↔ X): Taking natural language as a interface to generate or under other modal contents or to understand
  - Code & Image & Video & Table understanding / generation

```python
# Define a function to calculate the factorial of a number
def factorial(n):
    """Calculate the factorial of a number n."""
    if n == 0:
        return 1
    else:
        return n * factorial(n-1)

# Calculate and print the factorial of 5
print("The factorial of 5 is:", factorial(5))
```

Code

Image

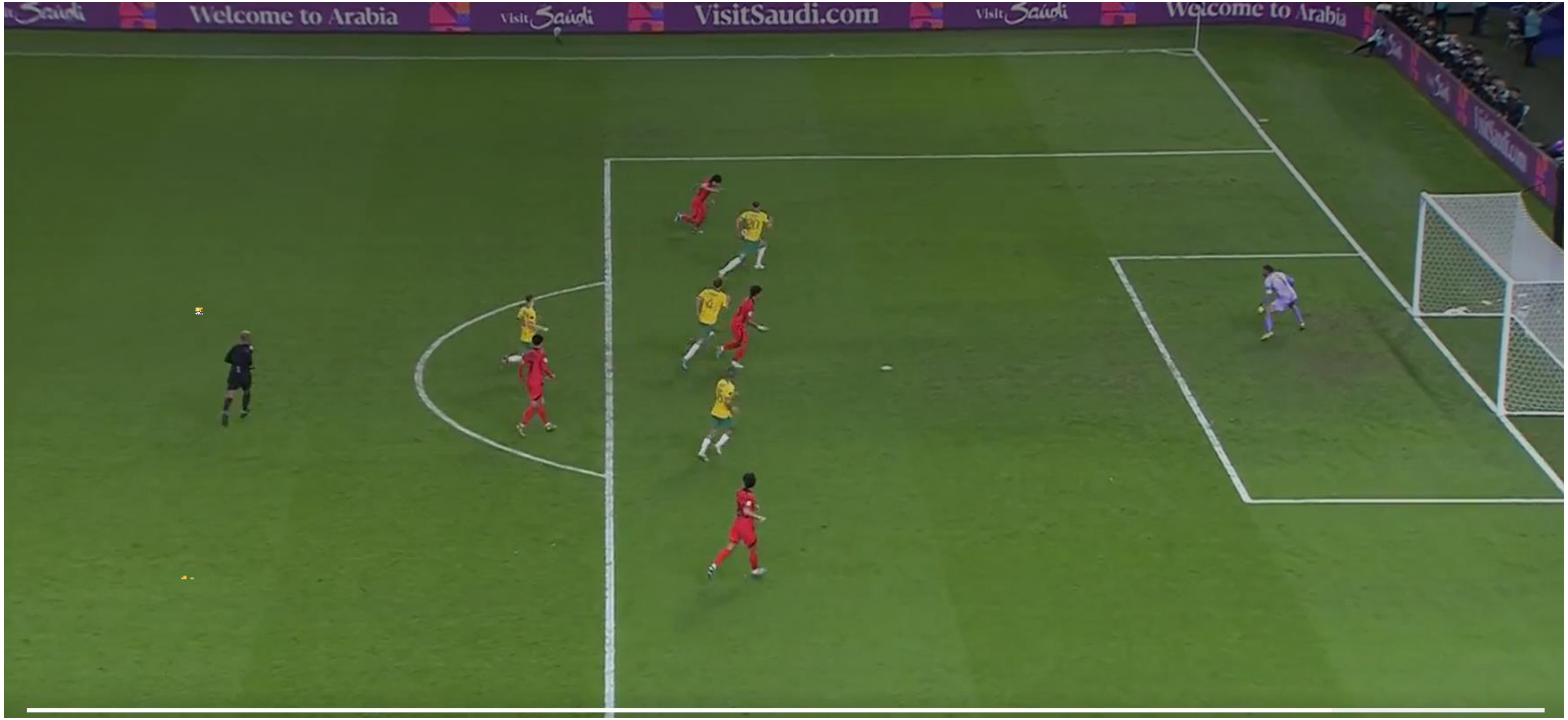
Video

Year	Population
2000	47,008,111
2010	49,554,112
2020	51,836,239
2024	51,751,065

Table & chart

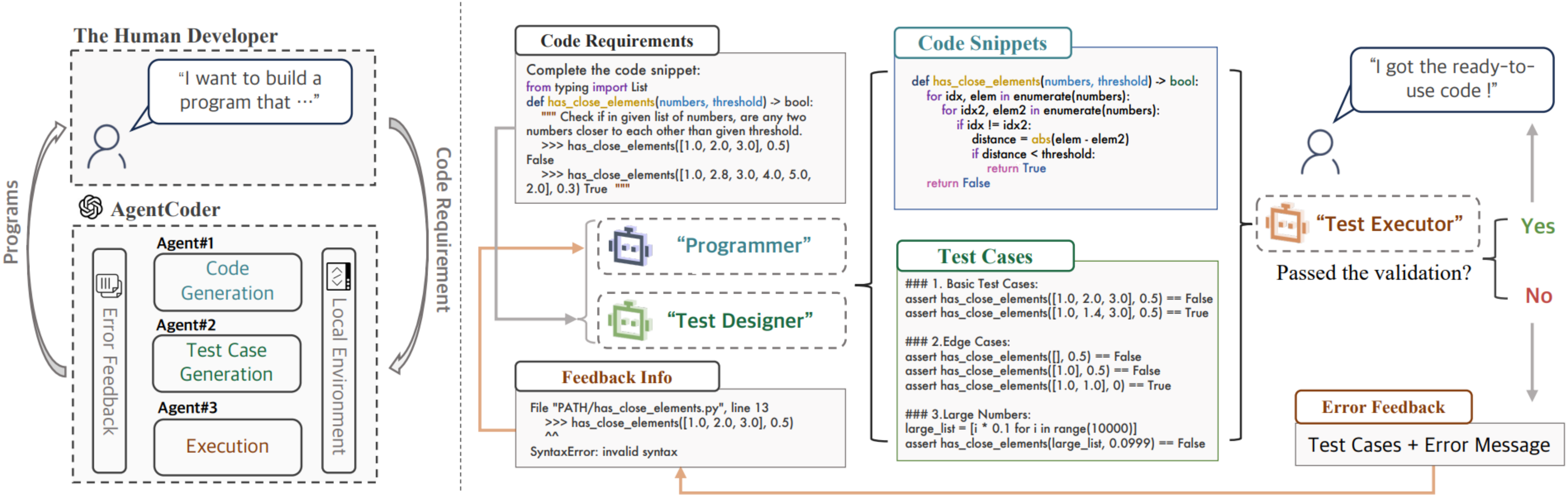
A illustration from a graphic novel. A bustling city street under the shine of a full moon. The sidewalks bustling with pedestrians enjoying the nightlife. At the corner stall, a young woman with fiery red hair, dressed in a signature velvet cloak, is haggling with the grumpy old vendor. the grumpy vendor, a tall, sophisticated man is wearing a sharp suit, sports a noteworthy moustache is animatedly conversing on his steampunk telephone.

https://web.stanford.edu/class/cs224n/slides/cs224n-2024-lecture01-wordvecs1-public.pdf



In this image, we see a moment from a soccer match. The scene is set in the penalty area of the field, where a key moment of play is occurring. The team in red seems to be on the attack, while the team in yellow is defending their goal. There is a sense of dynamism and anticipation as the players are strategically positioned, suggesting an ongoing intense match. The goalkeeper, dressed in purple, is focused on the ball and ready to react. The advertisement boards in the background indicate that the match is possibly being held in or is associated with Saudi Arabia. The image captures the universal appeal and drama of a soccer game, a sport that brings together fans and players from all corners of the world.

AgentCoder: Multi Agent-Code Generation with Iterative Testing and Optimisation [Huang et al 23]



AgentCoder: Multi Agent-Code Generation with Iterative Testing and Optimisation [Huang et al 23]

Write a python function to check whether the given array is monotonic or not.

CodeCoT

```
# Self-examination with test cases
assert isMonotonic([1, 2, 3, 4, 5]) == True  # Increasing sequence, should return True
assert isMonotonic([5, 4, 3, 2, 1]) == True  # Decreasing sequence, should return True
assert isMonotonic([1, 2, 2, 3, 2]) == False  # Non-monotonic sequence, should return False
```

AgentCoder

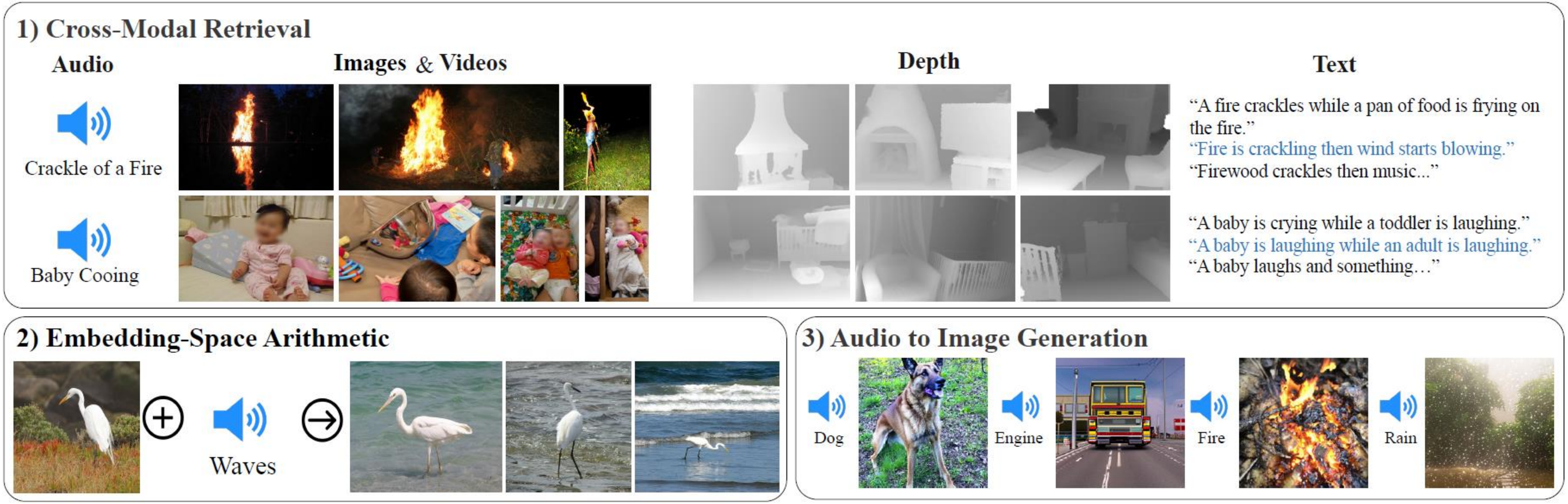
```
# Basic test cases
assert isMonotonic([1, 2, 3, 4, 5]) == True  # Increasing sequence
assert isMonotonic([5, 4, 3, 2, 1]) == True  # Decreasing sequence
assert isMonotonic([1, 2, 2, 3, 2]) == False  # Non-monotonic sequence

# Edge test cases
assert isMonotonic([1]) == True  # Single-element sequence
assert isMonotonic([]) == True  # Empty sequence
assert isMonotonic([1, 2]) == True  # Two-element increasing sequence
assert isMonotonic([2, 1]) == True  # Two-element decreasing sequence
assert isMonotonic([1, 1, 1, 1, 1]) == True  # All elements are equal, considered monotonic
assert isMonotonic([1, 1, 1, 2, 1]) == False  # Mixed values, not monotonic

# Large input
large_input = [i for i in range(1, 100001)]
# An increasing sequence of 100,000 elements
assert isMonotonic(large_input) == True  # Should be considered monotonic

# Create a large decreasing sequence
large_decreasing_input = [i for i in range(100000, 0, -1)]
# A decreasing sequence of 100,000 elements
assert isMonotonic(large_decreasing_input) == True  # Should be considered monotonic
```

IMAGEBIND: One Embedding Space To Bind Them All [Meta, 23]



IMAGEBIND: One Embedding Space To Bind Them All [Meta, 23]

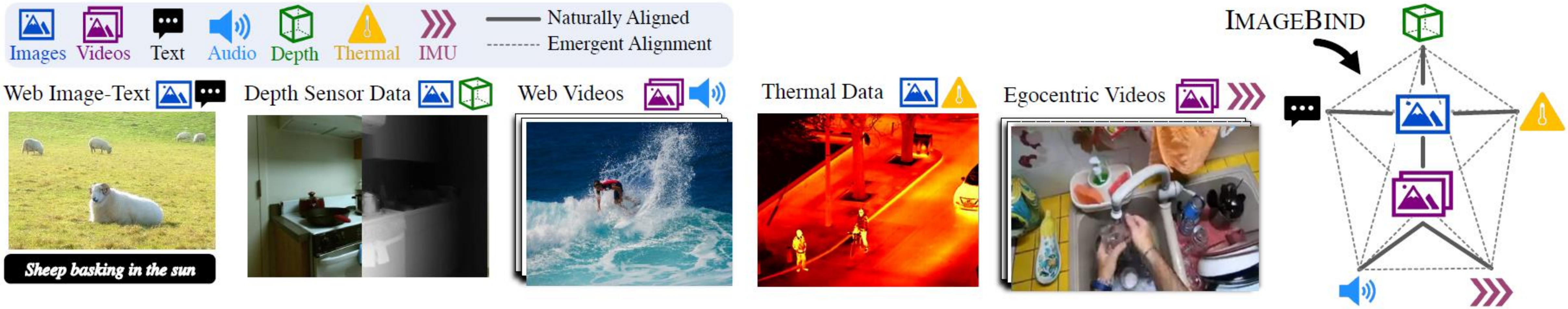


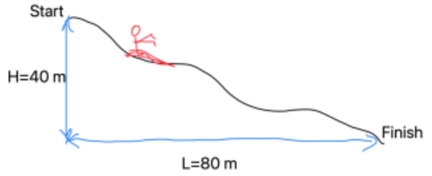
Figure 2. IMAGEBIND overview. Different modalities occur naturally aligned in different data sources, for instance images+text and video+audio in web data, depth or thermal information with images, IMU data in videos captured with egocentric cameras, *etc*. IMAGEBIND links all these modalities in a common embedding space, enabling new emergent alignments and capabilities.

Gemini: A Family of Highly Capable Multimodal Models [Google, 23]

Prompt

Here's a solution to a Physics problem by a student.

23. A skier slides down the frictionless slope as shown. What is the skier's speed at the bottom?



Solution:

The conservation of energy means that the energy of the skier is same at the beginning and at the end.

Start: $E = mgL$

End: $E = \frac{1}{2}mv^2$

$\Rightarrow mgL = \frac{1}{2}mv^2$

$\Rightarrow v = \sqrt{2gL} = \sqrt{2 \times 9.81 \times 80} = 39.6\ m.s^{-1}$

Try to reason about the question step by step.

Did the student get the correct answer? If the solution is wrong, please explain what is wrong and solve the problem.

Make sure to use LaTeX for math and round off the final answer to two decimal places.

Response

The student did not get the correct answer. The student made a mistake in the calculation of the potential energy at the start of the slope. The potential energy at the start of the slope is given by *mgh*, where *m* is the mass of the skier, *g* is the acceleration due to gravity, and *h* is the height of the slope. The student used the length of the slope, *L*, instead of the height, *H*, in the calculation of the potential energy.

The correct solution is:

The conservation of energy means that the total energy of the skier is the same at the beginning and at the end of the slope.

At the start of the slope, the skier has only potential energy, which is given by *mgh*.

At the end of the slope, the skier has only kinetic energy, which is given by $\frac{1}{2}mv^2$

Therefore, we can write:

$$mgh = \frac{1}{2}mv^2$$

Canceling out the mass *m* on both sides, we get:

$$gh = \frac{1}{2}v^2$$

Solving for *v*, we get:

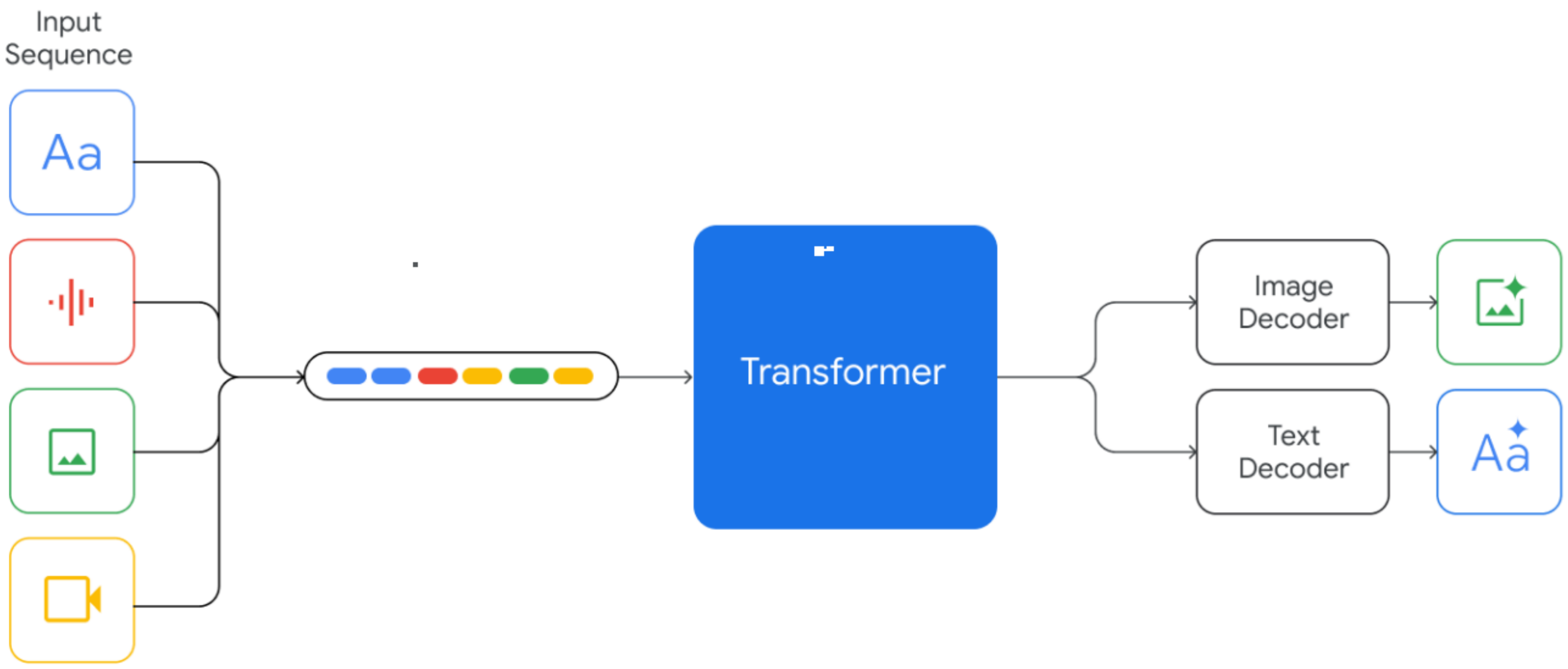
$$v = \sqrt{2gh}$$

Substituting the given values, we get:

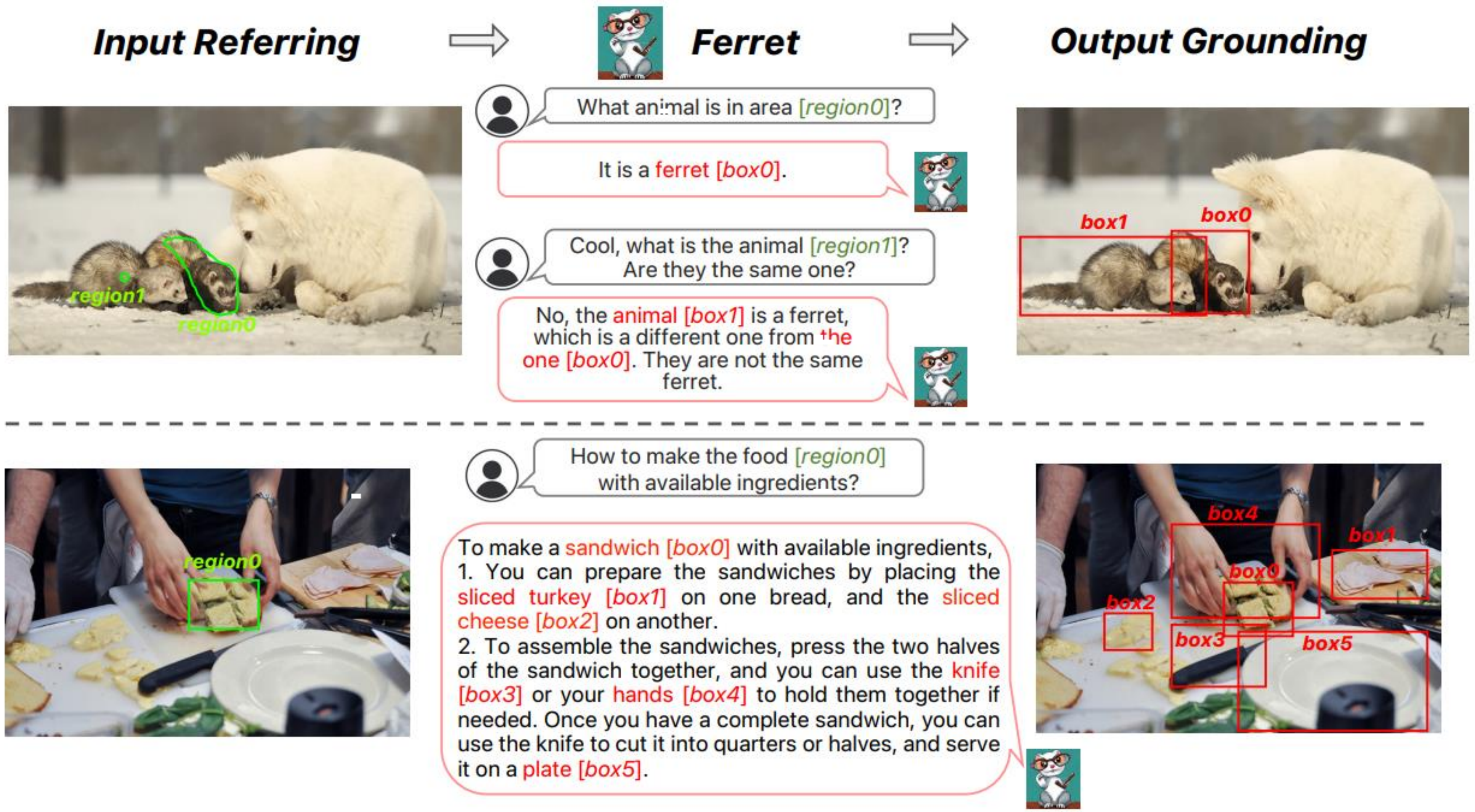
$$v = \sqrt{2 \times 9.81 \times 40} = 28.01 \text{ m/s}$$

Therefore, the skier's speed at the bottom of the slope is 28.01 m/s.

Gemini: A Family of Highly Capable Multimodal Models [Google, 23]

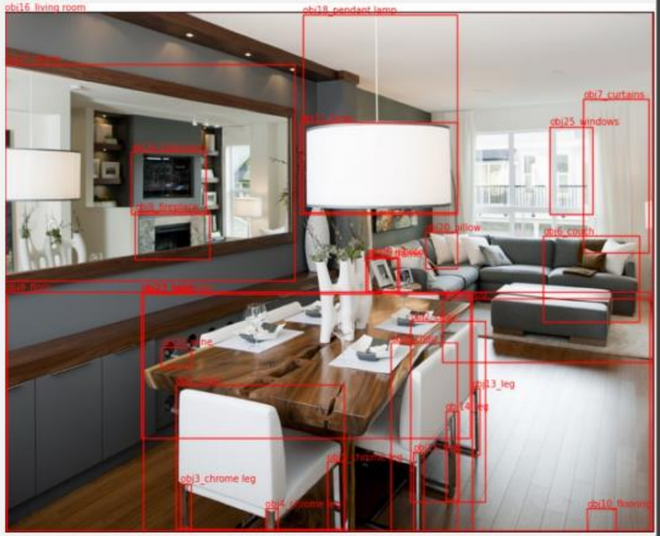


Ferret: Refer and Ground Anything Anywhere at Any Granularity [Apple, ‘23]



Ferret: Refer and Ground Anything Anywhere at Any Granularity [You et al '23]

Objects
Object 0: chair at [0.596, 0.637, 0.698, 0.997].
...
Relationships
Object 11 : frame → with → Object 9 : photo
...
Region Descriptions
Region Description at [0.560, 0.466, 0.600, 0.529] :
a white picture frame with a black and white photo on it.
...
Global Caption
White chairs sit around a polished wood dining table while
a sectional soft sits in the background.
...



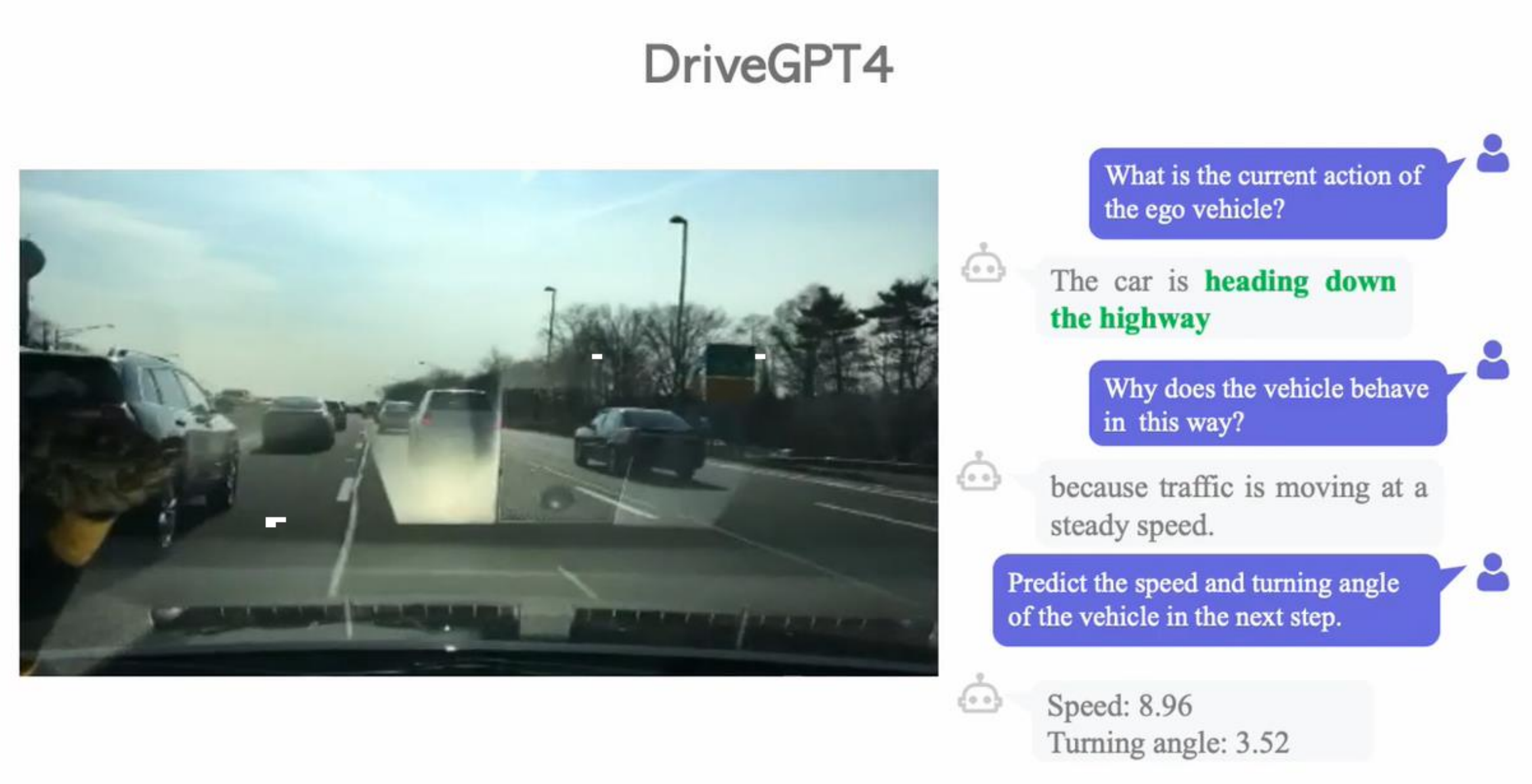
Conversation
Question:
What is the object [0.212, 0.539, 0.718, 0.820] made of?
===
Answer:
The dining table over there is crafted from a tree trunk.
===
Question:
Where would be a good place to have my dinner? Point out the locations in image.
===
Answer:
I suggest having dinner at the table [0.214, 0.541, 0.720, 0.997], possibly sitting on the white chair [0.264, 0.717, 0.524, 0.995]. Also, enjoy the wine [0.242, 0.644, 0.288, 0.682].
===
Question:
In terms of comfort for resting, which object is better - the one [0.830, 0.431, 0.978, 0.596] or the one [0.264, 0.717, 0.524, 0.995]?
===
Answer:
Among the two, the couch located [0.830, 0.431, 0.978, 0.596] would be more comfortable for resting. The object [0.264, 0.717, 0.524, 0.995] is a dining chair, which is typically less comfortable for extended relaxation.
....

Natural Language Processing

- World-grounded Extension: Ground natural language to world or virtual world, enabling interaction physically
  - Text2action & Action2text

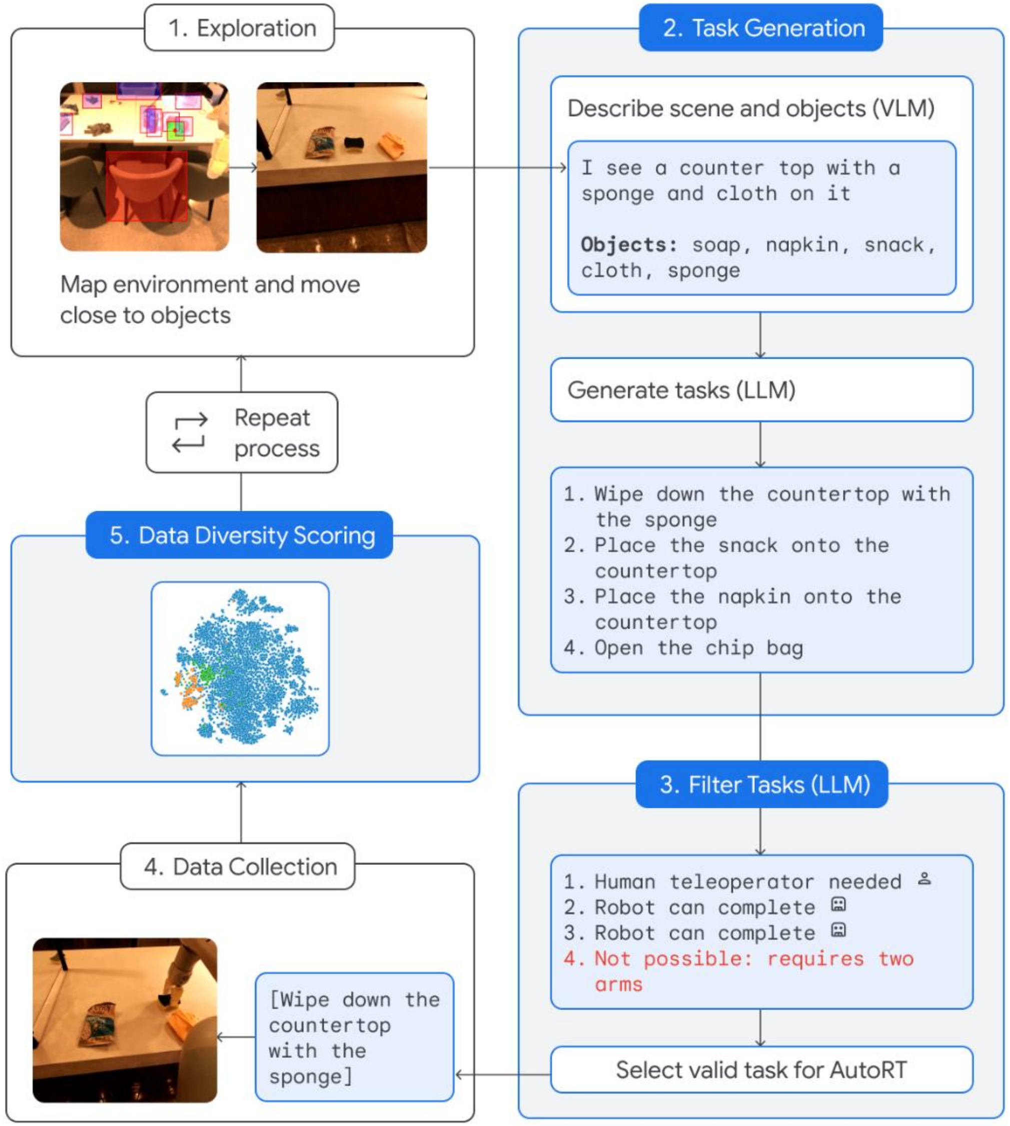
Large Language Model

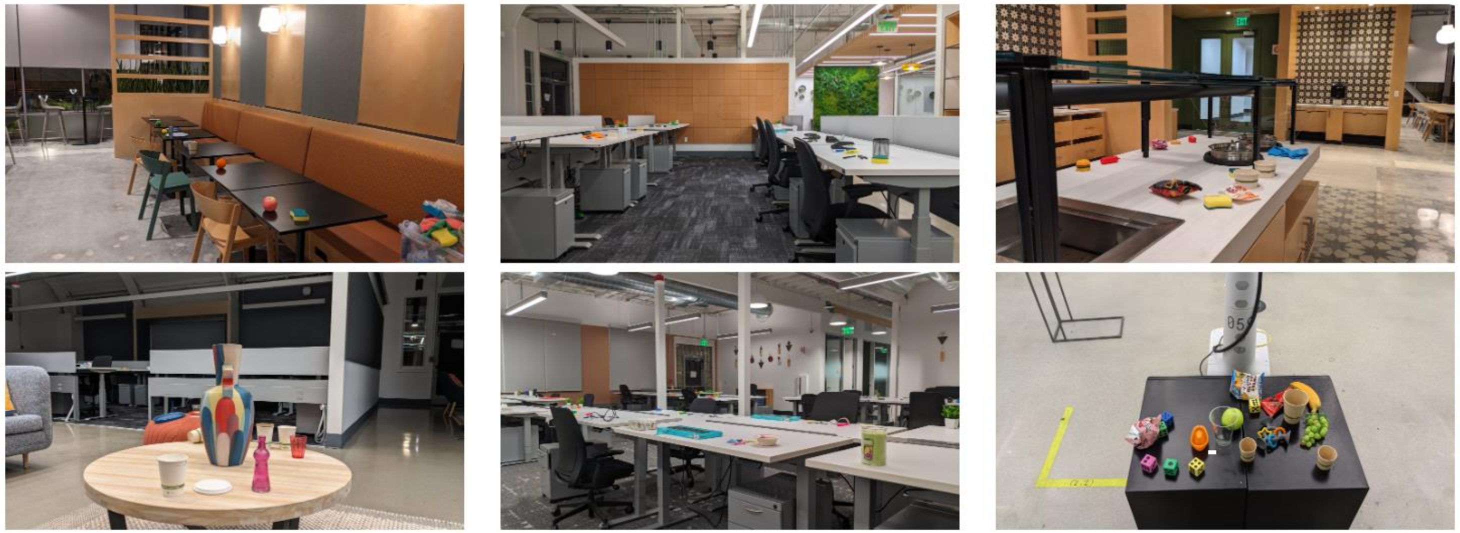
Natural Language Processing



https://tonyxuqaq.github.io/projects/DriveGPT4/

AutoRT: Embodied Foundation Models for Large Scale Orchestration of Robotic Agents [Ann et al '24]





Example environments where AutoRT was run

https://auto-rt.github.io/

AutoRT: Embodied Foundation Models for Large Scale Orchestration of Robotic Agents [Ann et al ‘24]

arrange the cups into a circle

count the objects on the table

fluff the pillows on the couch

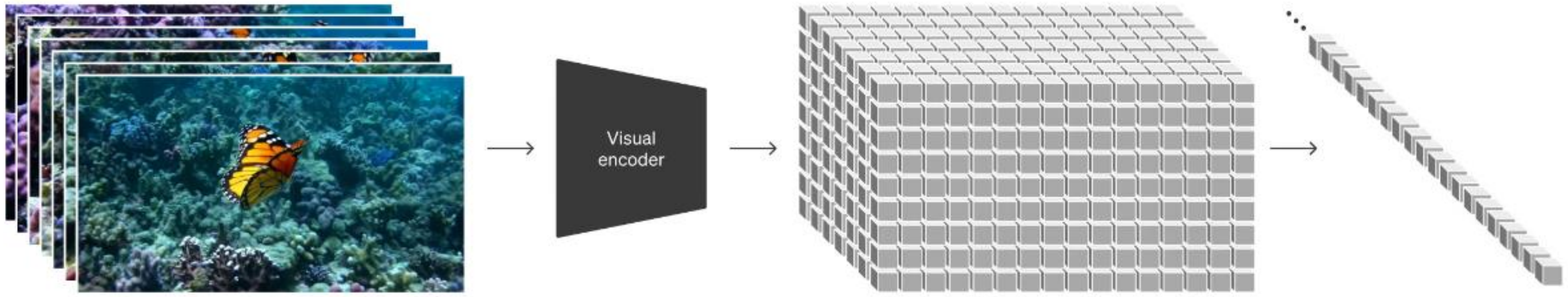
stack the boxes on top of each other

https://auto-rt.github.io/

Sora [OpenAI, 2024]: Video generation models as world simulators

- Sora is a diffusion *transformer*.

Turning visual data into patches



Scaling transformers for video generation

https://openai.com/research/video-generation-models-as-world-simulators

Sora [OpenAI, 2024]

- **Language understanding**

<u>**a woman**</u> wearing
<u>**purple overalls and cowboy boots**</u>
taking a pleasant stroll in <u>**Mumbai, India**</u>
during <u>**a winter storm**</u>

Sora [OpenAI, 2024]

- **Prompting with images and videos**

A Shiba Inu dog wearing a beret and black turtleneck.

An image of a realistic cloud that spells “SORA”.

In an ornate, historical hall, a massive tidal wave peaks and begins to crash. Two surfers, seizing the moment, skillfully navigate the face of the wave.

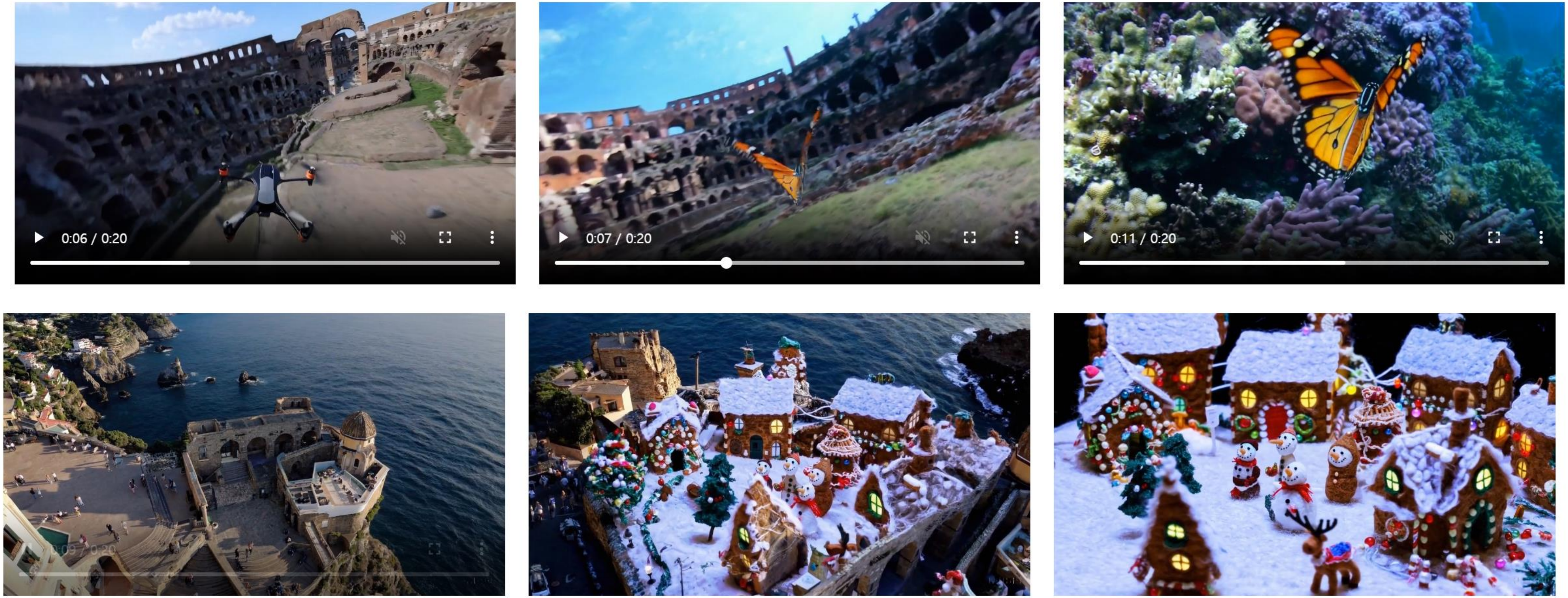
- **Video-to-video editing**

Input video | change the setting to be in a lush jungle

https://openai.com/research/video-generation-models-as-world-simulators

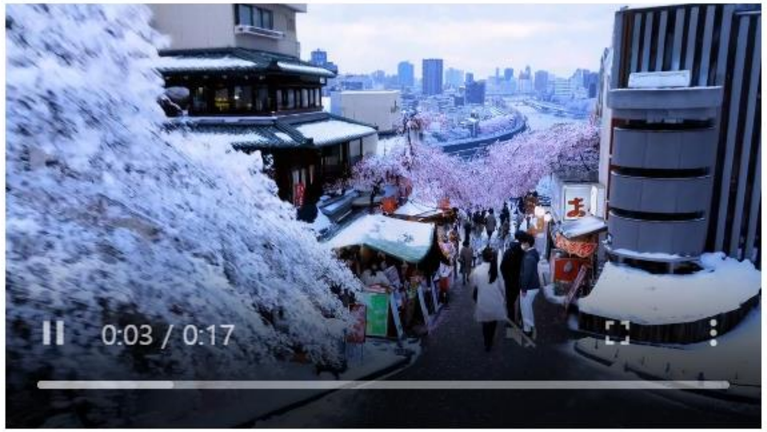
Sora [OpenAI, 2024]

- **Connecting videos**



Sora [OpenAI, 2024]

- **Emerging simulation capabilities**
  - **3D consistency**



  - **Long-range coherence and object permanence**

Sora [OpenAI, 2024]

- **Emerging simulation capabilities**
  - **Interacting with the world**

  - **Simulating digital worlds**

Natural Language Processing: Related disciplines

- **Artificial Intelligence**
  - The capacity for language is one of the central features of human intelligence, and is therefore a prerequisite or artificial intelligence
  - Natural language processing is a potential solution to the “knowledge bottleneck”, by acquiring knowledge from texts, and perhaps also from conversations.
  - Natural language understanding cannot be achieved in isolation from knowledge and reasoning
  - Like LLM (large language model), language models trained from large text corpus are actually providing commonsense knowledge and domain knowledge which are universally applied across various tasks.
- **Machine Learning**
  - Natural language provides the uniquely established problems in ML, being different from other areas, in the aspects:
    - Discreteness: Unlike images or audio, text data is fundamentally discrete, with meaning created by combinatorial arrangements of symbolic units.
    - Concept creativity: New words are always being created
    - Compositionality: Language is compositional: units such as words can combine to create phrases, which can combine by the very same principles to create larger phrases.
  - Classically, natural language has been studied in machine learning, strongly affecting the area of ``Structured Prediction'', which is a supervised machine learning, aiming to predicting structured objects

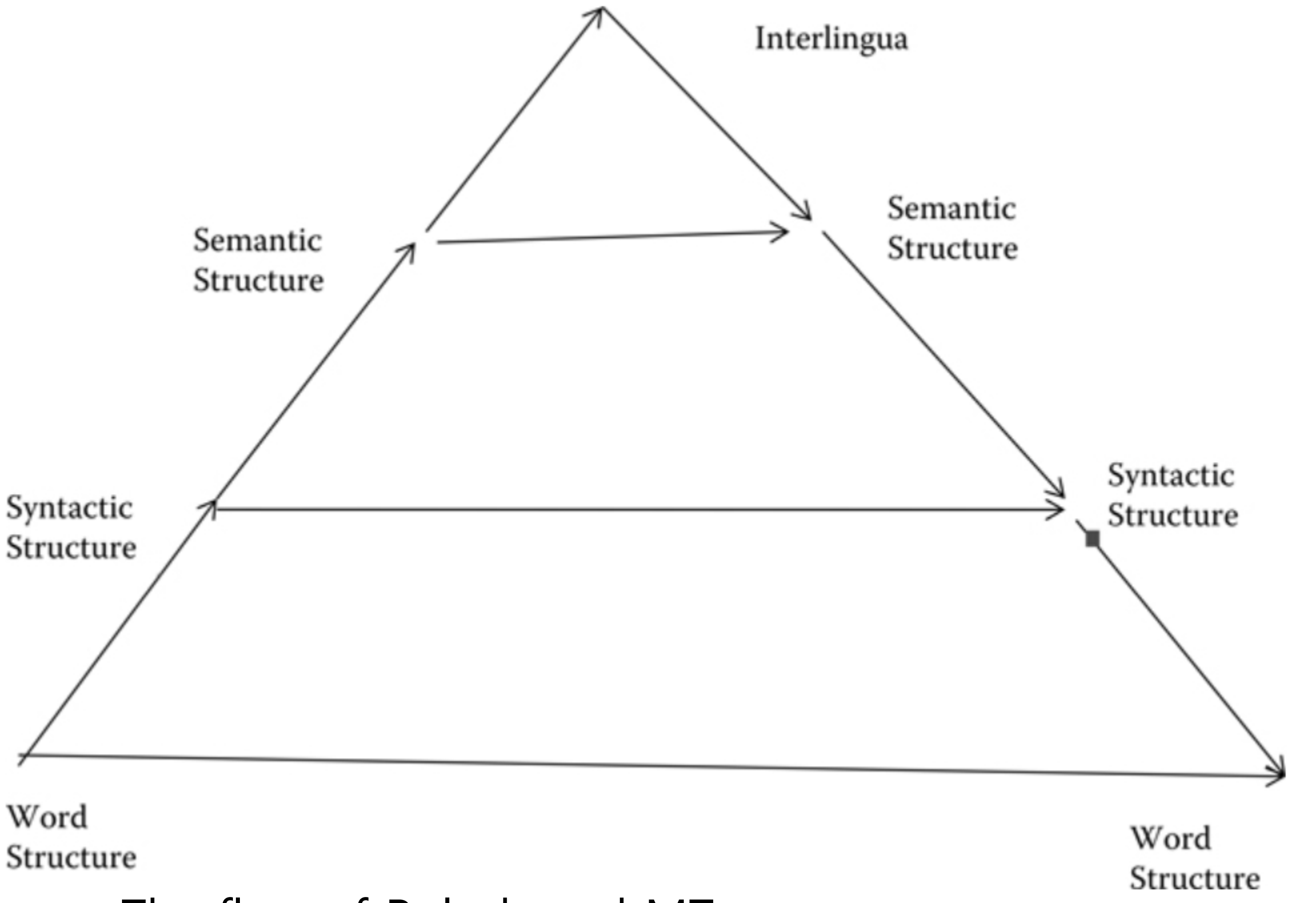
Main parts refer to https://github.com/jacobeisenstein/gt-nlp-class/blob/master/notes/eisenstein-nlp-notes.pdf

Natural Language Processing: Related disciplines

- **Information retrieval**
  - Typically, a information retrieval is regarded as a “semantic matching” between a query and a text.
    - Requires to understand the underlying meaning of a query and a text, to search semantically relevant texts
  - Previously, IR has been advanced by language modeling approaches for IR
    - The relevance score of a document is computed in a way of being proportional to the generative probability of a query from a document
      - ✓ P(q|doc) or P(doc|q)
    - While traditionally it is mostly based on unigram language models, bigram or proximal language models have been developed
    - Now, this kind of generative models has renewed as a generative IR and a cross-attention model using sequence-to-sequence model in the neural IR
  - Development
    - Classical strands: Classification model (BM25) → Generative model (LM4IR)
    - Modern neural approaches: Dense retrieval, Cross-encoder-based matching, generative retrieval

Natural Language Processing: Related disciplines

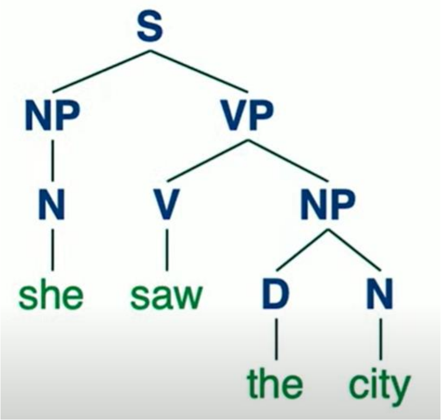
- **Linguistics**
  - The fundamental role in the rule-based NLP and statistical NLP



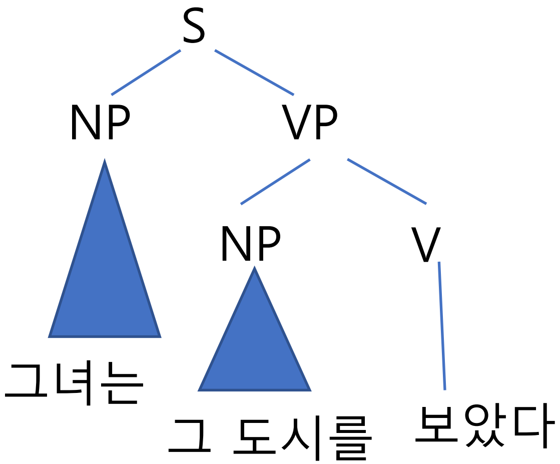
The flow of Rule-based MT

Natural Language Processing: Related disciplines

- **Linguistics**
  - Syntax-based transfer in the rule-based MT
    - 1) Analysis: A source sentence is analyzed to a syntactic structure
      - ✓ This syntactic structure is an instance of a linguistically well-founded grammar
        - Phrase structure grammar, Lexical functional grammar, Minimalist program-based grammar
    - 2) Transfer: A source syntactic structure is converted to a target one
      - ✓ E.g.) English (SVO) → Korean (SOV)
    - 3) Generation: A target syntactic structure is realized into a final target sentence



https://wikidocs.net/123085



Natural Language Processing: Related disciplines

- **Linguistics**
  - Morphology
    - Morphemes: The smallest units in a language with some independent meaning
    - Part-of-speech **(POS)**
      - ✓ a category of words (or, more generally, of lexical items) that have similar grammatical properties.
    - Inflection
      - ✓ Declensions: Nominal inflectional paradigms
      - ✓ Conjugations: Verbal inflectional paradigms.
  - Semantic
  - Pragmatics
  - ….

Natural Language Processing: Related disciplines

- **Linguistics**
  - Many NLP tasks correspond to structural subfields of linguistics

Subfields of linguistics	NLP Tasks
Phonetics Phonology	Speech recognition
Morphology Syntax	Word segmentation POS tagging Parsing
Semantics	Word sense disambiguation Semantic role labeling Semantic parsing
Pragmatics	Named entity recognition/disambiguation Reading comprehension

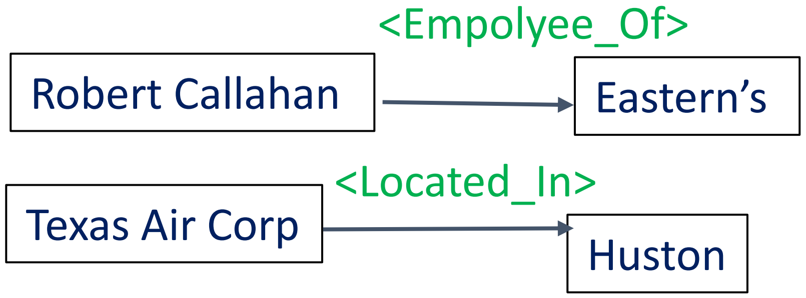
Natural Language Processing: Tasks - Information Extraction

According to Robert Callahan, president of Eastern’s flight attendants union, the past practice of Eastern’s parent, Houston-based Texas Air Corp., has involved ultimatums to unions to accept the carrier’s terms

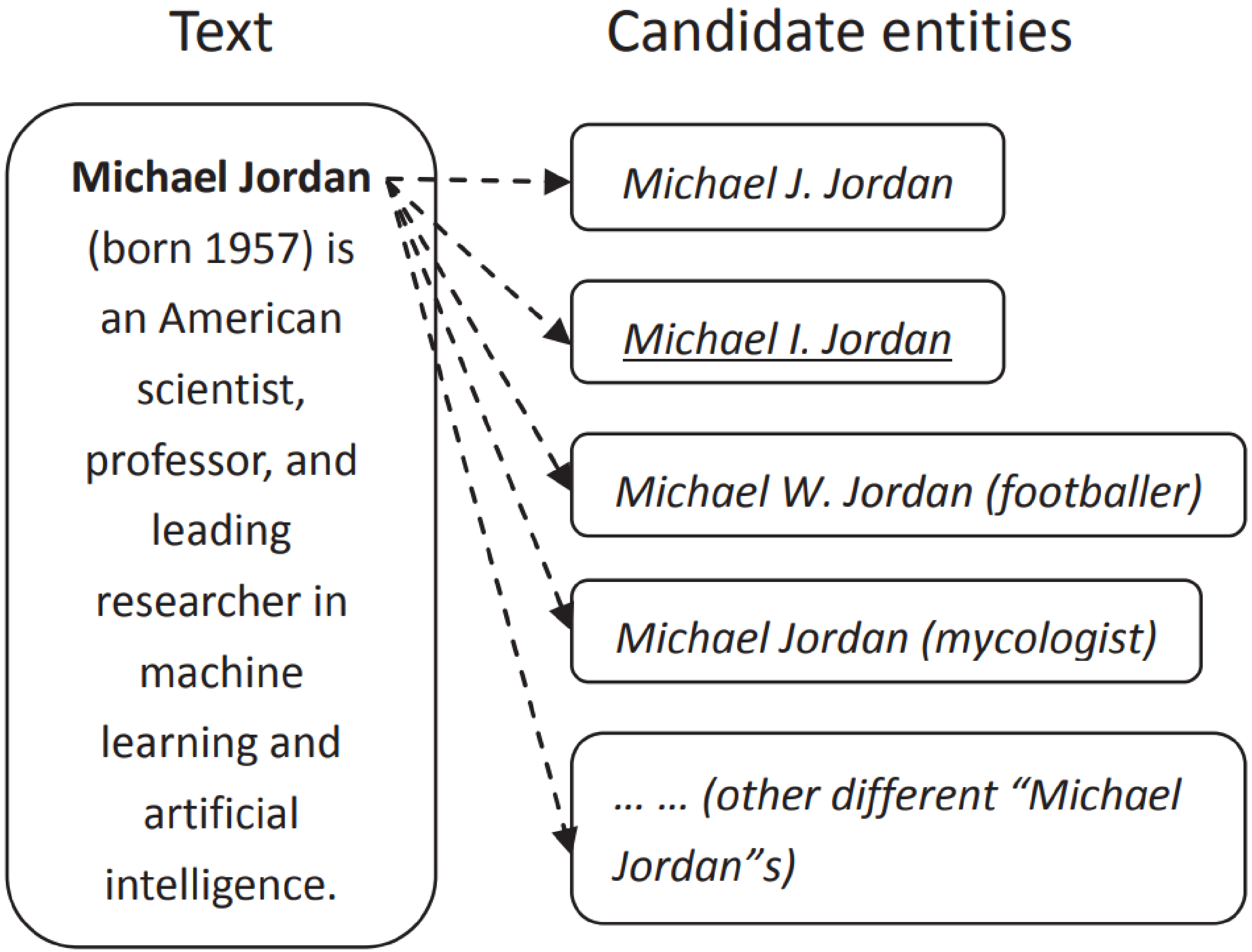
Entity extraction

According to <Per> Robert Callahan </Per>, president of <Org> Eastern’s </Org> flight attendants union, the past practice of <Org> Eastern’s </Org> parent, <Loc> Houston </Loc> -based <Org> Texas Air Corp. </Org>, has involved ultimatums to unions to accept the carrier’s terms

Relation extraction



Natural Language Processing: Tasks - Entity Linking



Natural Language Processing: Tasks - Machine Reading Comprehension

- SQuAD / KorQuAD

In meteorology, precipitation is any product of the condensation of atmospheric water vapor that falls under **gravity**. The main forms of precipitation include drizzle, rain, sleet, snow, **graupel** and hail... Precipitation forms as smaller droplets coalesce via collision with other rain drops or ice crystals **within a cloud**. Short, intense periods of rain in scattered locations are called "showers".

What causes precipitation to fall?
gravity

What is another main form of precipitation besides drizzle, rain, snow, sleet and hail?
graupel

Where do water droplets collide with ice crystals to form precipitation?
within a cloud

Natural Language Processing: Tasks - Chatbot

Machine: *hi*
Human: *hello*
Machine: *could you please let me know what are the operating systems you are using ?*
Human: *i am using linux*
Machine: *ok it was in the ssl vpn on which you stated it was broken*
Machine: *please check if you are logged in to corp network*
Machine: *what is the error that you are running please*
Human: *i am seeing an error related to vpn*
Machine: *what is the error message that you are getting when connecting to vpn using network connect ?*
Human: *connection refused or something like that*
Machine: *may i know the version of network connect you connect ?*
Human: *i am not sure i know that*
Machine: *is the network connect prompting for any user certificate*
Machine: *?*
Human: *yes*
Machine: *are you at home or at work at the moment ?*
Machine: *office ?*
Human: *i am at home*
Machine: *try this*
Machine: *goto <URL> page and enter your name and password*
Human: *done*
Machine: *try logging in with <NAME> and check if its working fine now*
Human: *yes , now it works !*
Machine: *great*
Machine: *anything else that i can help*
Human: *no , that was very useful*
Machine: *you are other than <NAME> :)*
Machine: *have a nice night*

Natural Language Processing: Tasks – Conversational Question Answering

Jessica went to sit in her rocking chair. Today was her birthday and she was turning 80. Her granddaughter Annie was coming over in the afternoon and Jessica was very excited to see her. Her daughter Melanie and Melanie's husband Josh were coming as well. Jessica had . . .

Q_1: Who had a birthday?
A_1: Jessica
R_1: Jessica went to sit in her rocking chair. Today was her birthday and she was turning 80.

Q_2: How old would she be?
A_2: 80
R_2: she was turning 80

Q_3: Did she plan to have any visitors?
A_3: Yes
R_3: Her granddaughter Annie was coming over

Q_4: How many?
A_4: Three
R_4: Her granddaughter Annie was coming over in the afternoon and Jessica was very excited to see her. Her daughter Melanie and Melanie's husband Josh were coming as well.

Q_5: Who?
A_5: Annie, Melanie and Josh
R_5: Her granddaughter Annie was coming over in the afternoon and Jessica was very excited to see her. Her daughter Melanie and Melanie's husband Josh were coming as well.

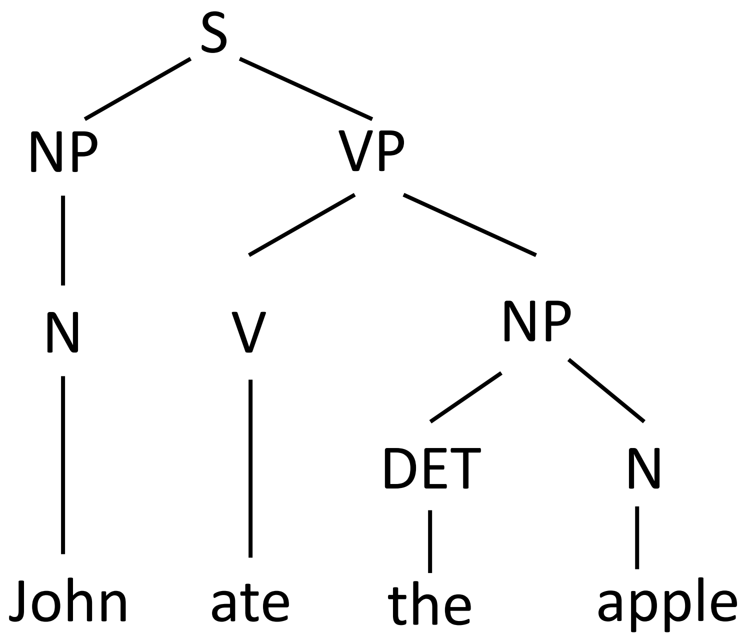
Natural Language Processing: Tasks – POS Tagging

- Input: Plays well with others
- Ambiguity: NNS/VBZ UH/JJ/NN/RB IN NNS
- Output: Plays/VBZ well/RB with/IN others/NNS

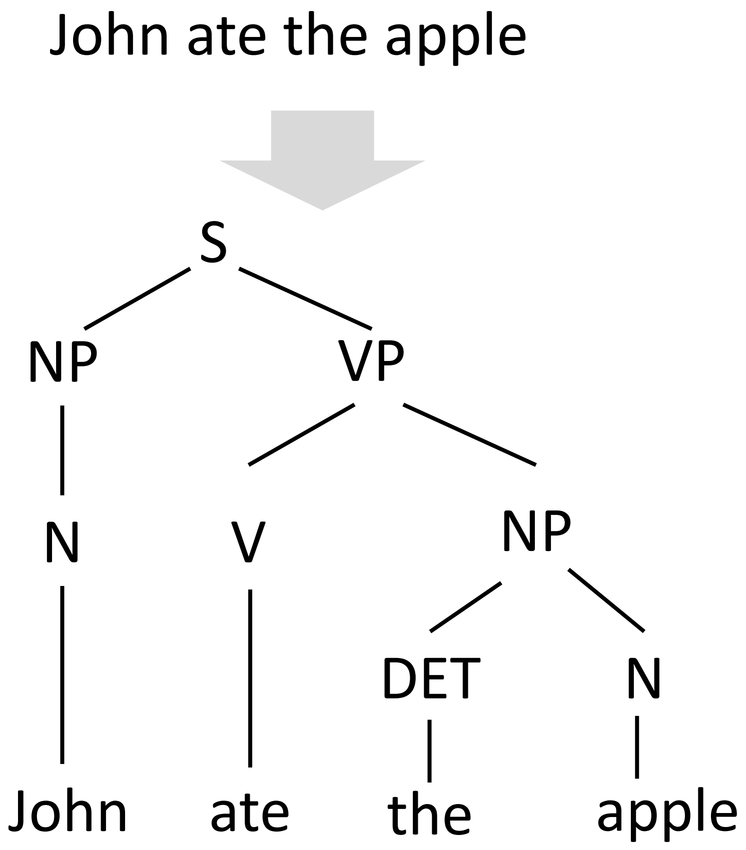
Natural Language Processing: Tasks – Parsing

- Sentence: “John ate the apple”
- Parse tree (PSG tree)

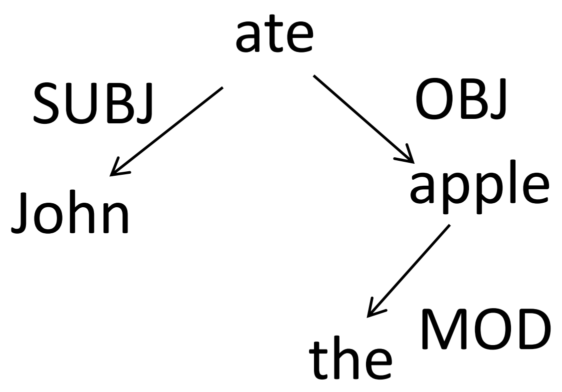
S → NP VP
NP → N
NP → DET N
VP → V NP
N → John
V → ate
DET → the
N → apple



Natural Language Processing: Tasks - Dependency Parsing



PSG tree



Dependency tree

Natural Language Processing: Tasks - Semantic Role Labeling

Jim gave the book to the professor

[Agent Jim] gave [Patient the book] [Goal to the professor.]

Semantic roles	Description
Agent	Initiator of action, capable of volition
Patient	Affected by action, undergoes change of state
Theme	Entity moving, or being "located"
Experiencer	Perceives action but not in control

Beneficiary Instrument Location

Source Goal

Natural Language Processing: Tasks - Coreference Resolution

[A man named Lionel Gaedi] went to [the Port-au-Prince morgue]2 in search of [[his] brother], [Josef], but was unable to find [[his] body] among [the piles of corpses that had been left [there]].

[A man named Lionel Gaedi]1 went to [the Port-au-Prince morgue]2 in search of [[his]1 brother]3, [Josef]3, but was unable to find [[his]3 body]4 among [the piles of corpses that had been left [there]2]5.

Contents

- Introduction
  - Goal & Tasks & Applications
  - Related disciplines
    - Artificial intelligence, Machine learning, Linguistics, etc.

- Natural language processing: Methods
  - Methods
    - Rule-based approach: ELIZA, RBMT
    - Statistical method: HMM, PCFG
    - Deep learning: Word2vec, RNN, LSTM, Transformer
    - Pretrained language model: BERT
    - Large language model: GPT3
    - Human-aligned LLM: ChatGPT
    - LLM for X: Multimodal & Robotics, Others

- Course schedule

Natural language processing: Methods

- Rule-based approaches [50s~80s]: ELIZA & RBMT
  - Syntax & Grammar-driven, Knowledge-based & symbolic approaches
  - Largely relied on linguistic theory
    - Transformational Generative Grammar, the Government and Binding Theory [Chomsky]
    - HPSG, Lexical functional grammar, Tree adjoining grammar
- Statistical approaches [~2000~2010 early]: HMM & PCFG
  - Corpus-based approaches
  - Train generative models from annotated corpus

  - Hidden Markov model ('86)
  - Probabilistic CFG ('80~'90)
  - IBM's Statistical machine translation ('90)
  - PennTreeBank ('93)
  - Statistical language learning [Charniak '94]
  - Head-driven statistical model for parsing [Collins '99]

Natural language processing: Methods

- Machine learning approaches [90s late~]: Structured prediction & Probabilistic graphical models
  - Structured discrimination, include many NLP tasks, such as sequential tagging, has been advanced to the structured prediction
  - Probabilistic graphical models have been extensively applied to NLP tasks

  - Support vector machine (Vapnik '95)
  - Structed perceptron (Collins '02)
  - Conditional random field (Lafferty '01)
  - Latent Dirichlet Allocation (Blei '03)

Natural language processing: Methods

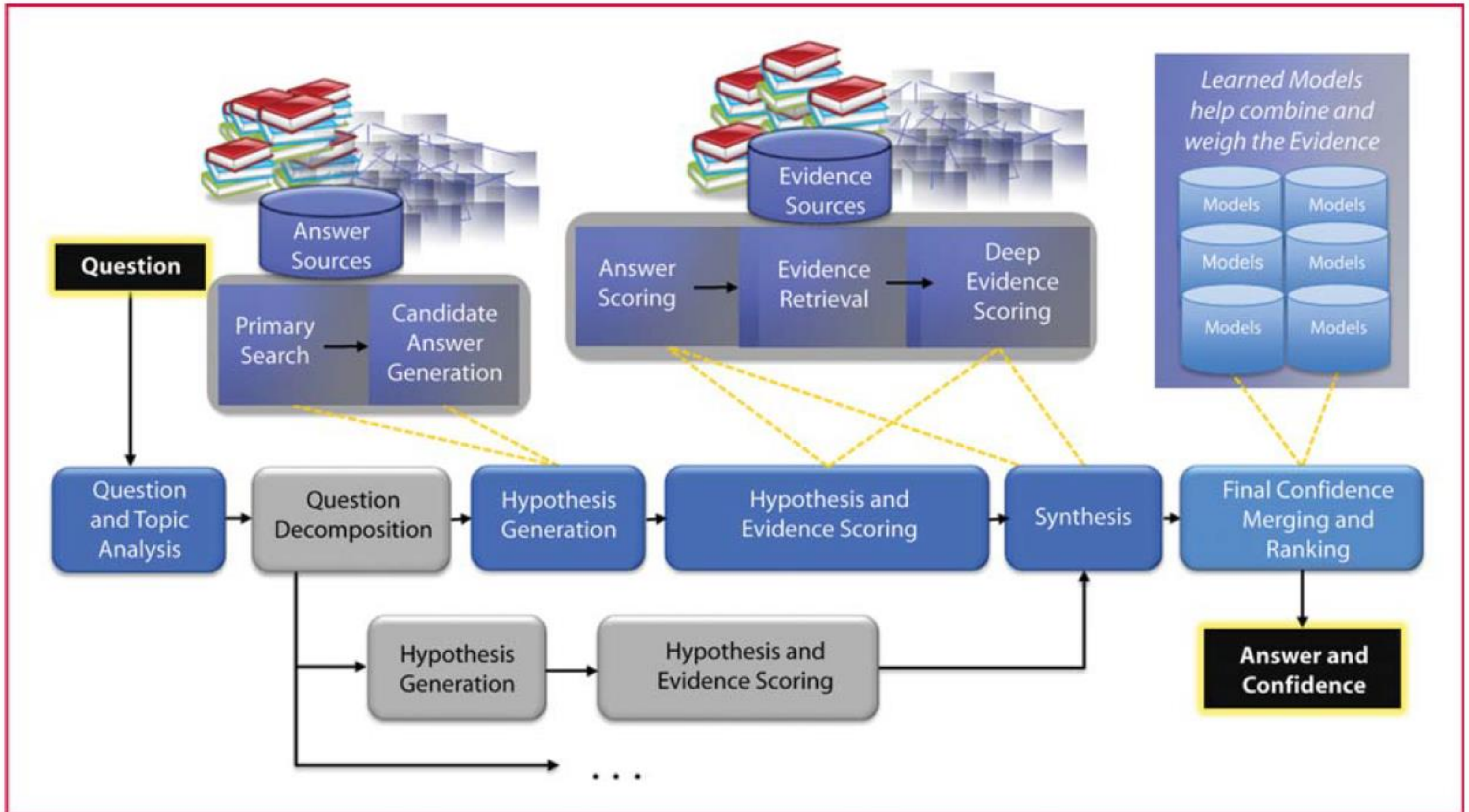
- Deep learning approaches [2010s late~]: Word embedding, Sequence-to-sequence, Transformer
  - Neural language model (Bengio et al. '03)
  - SENNA (Collobert et al. '11)
  - Recursive neural network (Socher et al. '12)
  - Neural machine translation (Cho et al '14)
    - Neural machine translation with attention mechanism (Bahdanau et al. '15)
  - Neural Turing machine (Grave '14)
  - Memory network (Weston et al '14)
  - Transformer (Vaswani et al '17)
- Pretrained language models [2017~]
  - Elmo (Peters et al '17]
  - GPT (Radford et al '17)
  - BERT (Devlin et al '17)

Natural language processing: Methods

- Language language models [2019~]
  - GPT3 (Brown et al '20)
  - InstructGPT (Ouwang et al '22)
  - ChatGPT (OpenAI 22)
  - Bard (Google 23)
- Multimodal LLMs [2023~]
  - GPT4 (OpenAI 23)
  - Gemini (Google 24)
  - Sora (OpenAI 24)
- World-grounded LLM ?

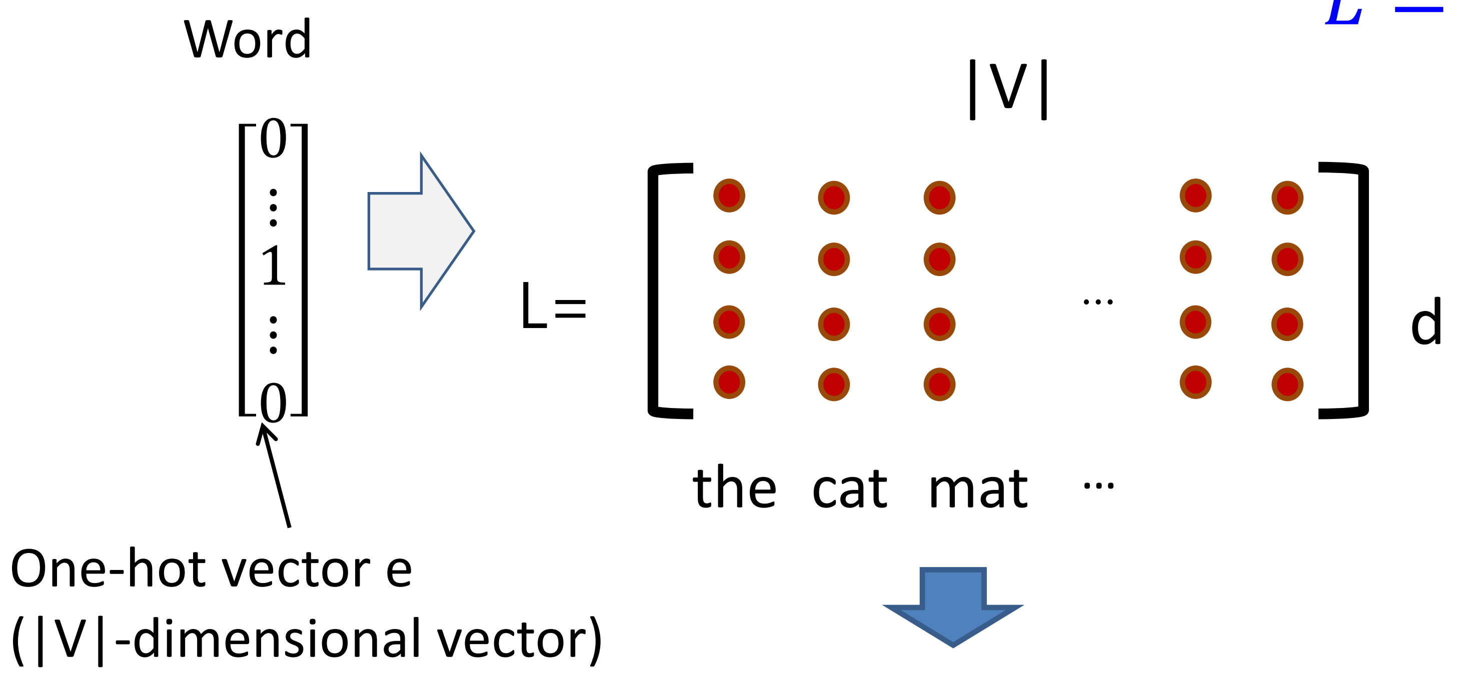
Machine learning-based NLP: IBM Watson System

- Open-domain question answering system (DeepQA)
  - ✓ In 2011, the Watson system won against Jeopardy! Challenge quiz show champions Brad Rutter and Ken Jennings



Word embedding: Lookup Table

$L = R^{d \times |V|}$



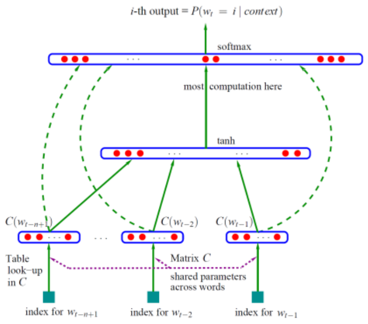
$x = L\,e$

Word vector *x* is obtained from one-hot vector e by referring to lookup table

Natural Language Processing using Word Embedding

Raw corpus

Learning word embedding matrix



Unsupervised

L= [...]

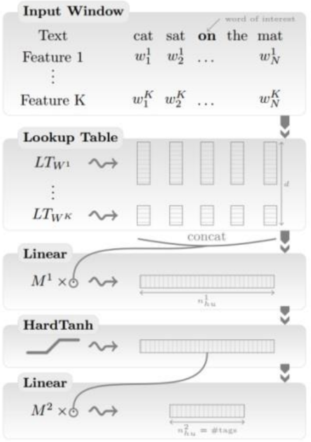
Application-specific NN

Initialize lookup table

Annotated corpus

Supervised

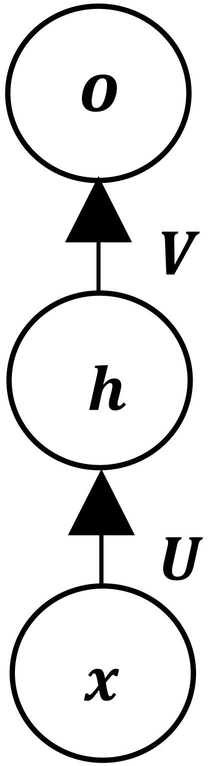
Application-specific neural network



➔ Lookup table is further fine-tuned

Recurrent Neural Networks

Feedforward NN



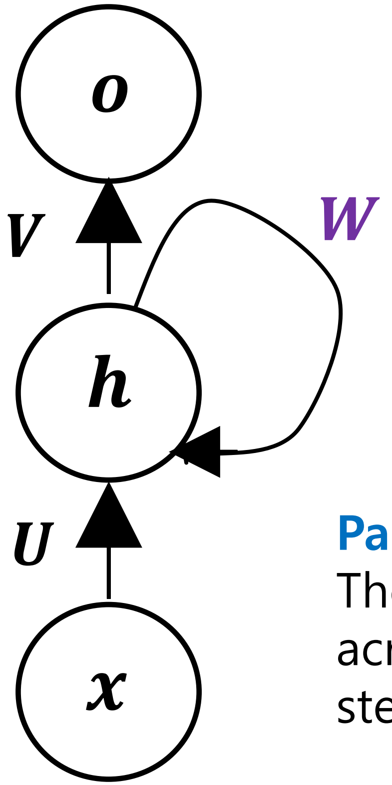
Output layer

Hidden layer

Input layer

$$\boldsymbol{h} = g(\boldsymbol{U}\boldsymbol{x})$$

Recurrent neural networks



Parameter sharing:
The same weights across several time steps

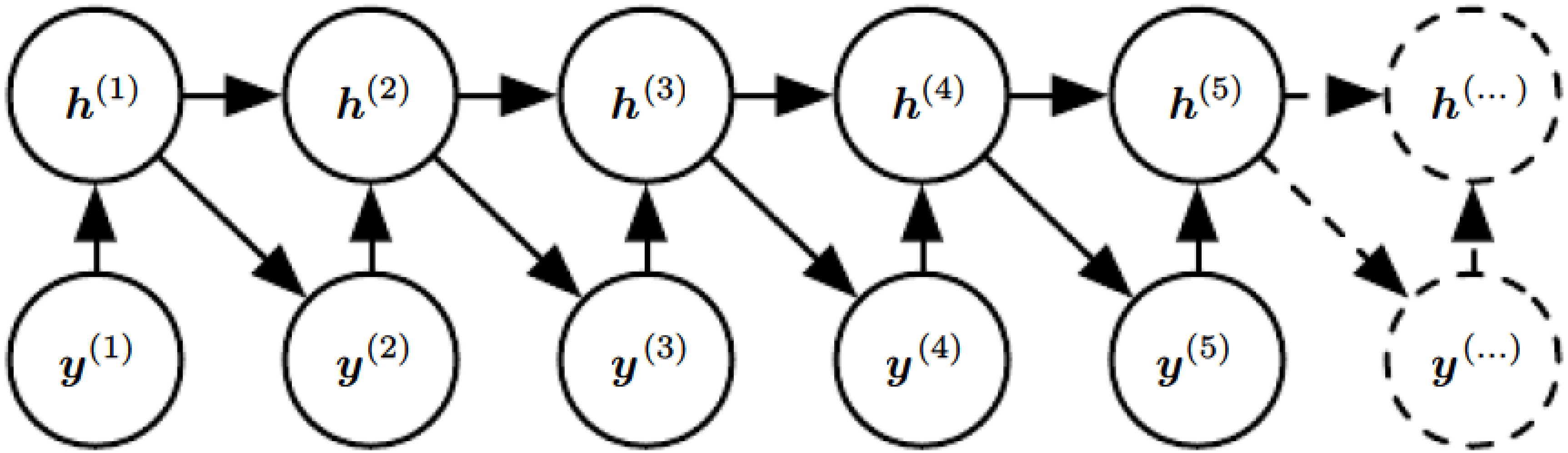
$$\boldsymbol{h}^{(t)} = g\left(\boldsymbol{W}\boldsymbol{h}^{(t-1)} + \boldsymbol{U}\boldsymbol{x}^{(t)}\right)$$

Recurrent Language Model

$$P(\mathbb{Y}) = P(\mathbf{y}^{(1)},\dots,\mathbf{y}^{(\tau)}) = \prod_{t=1}^{\tau} P(\mathbf{y}^{(t)} \mid \boxed{\mathbf{y}^{(t-1)},\mathbf{y}^{(t-2)},\dots,\mathbf{y}^{(1)}})$$

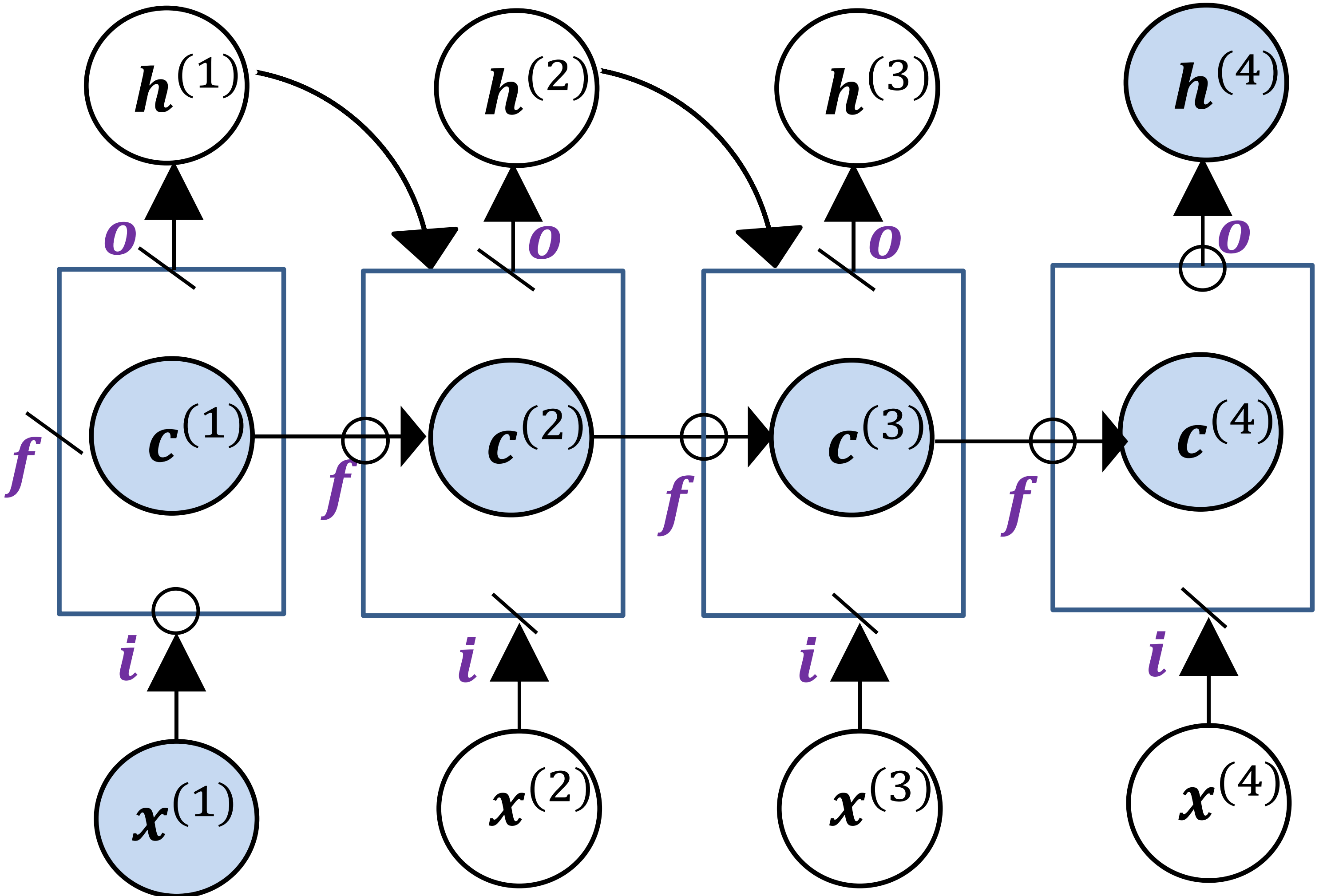
$\boldsymbol{h}^{(t-1)}$

- Introducing the state variable in the graphical model of the RNN



Long Short Term Memory (LSTM)

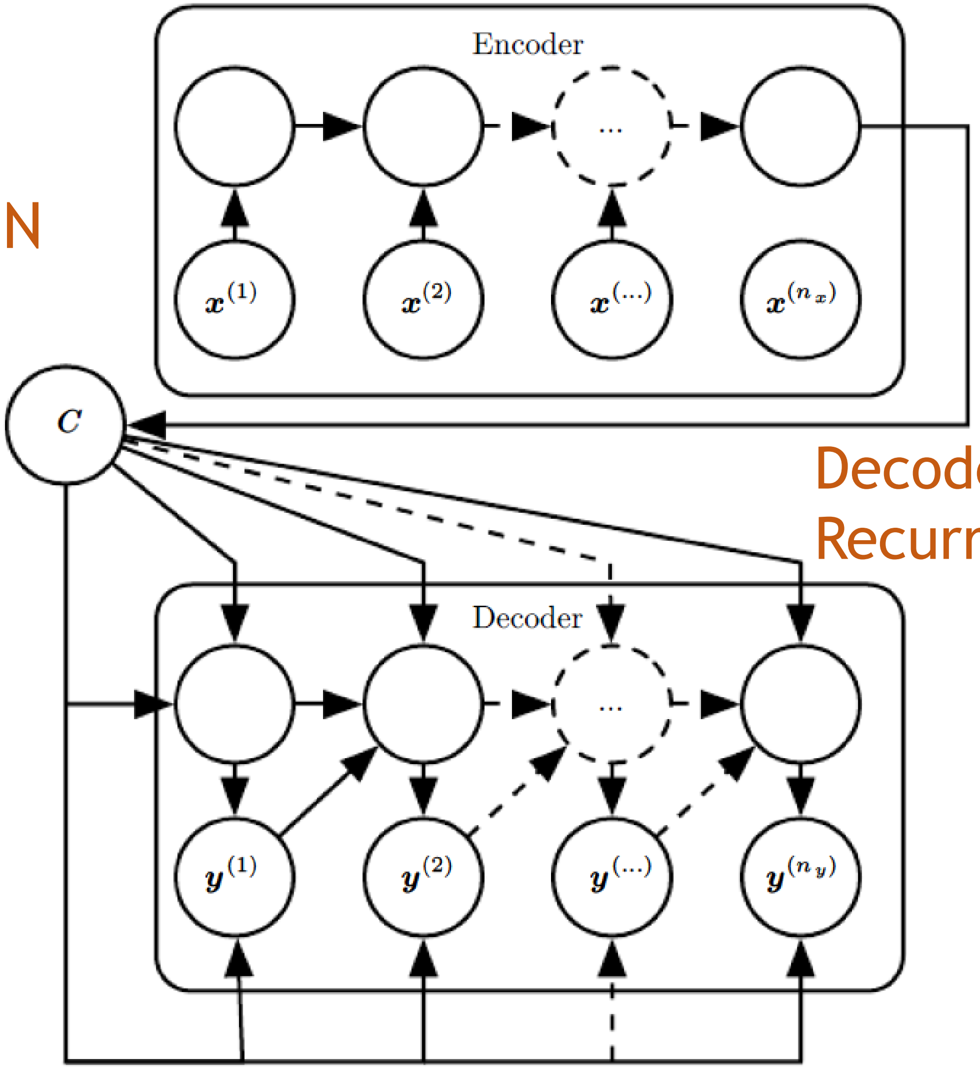
- Using **gated units** to resolve vanishing gradients of RNN



Neural Encoder-Decoder [Cho et al '14]

- Computing the log of translation probability $log\ P(y|x)$ by two RNNs

Encoder: RNN

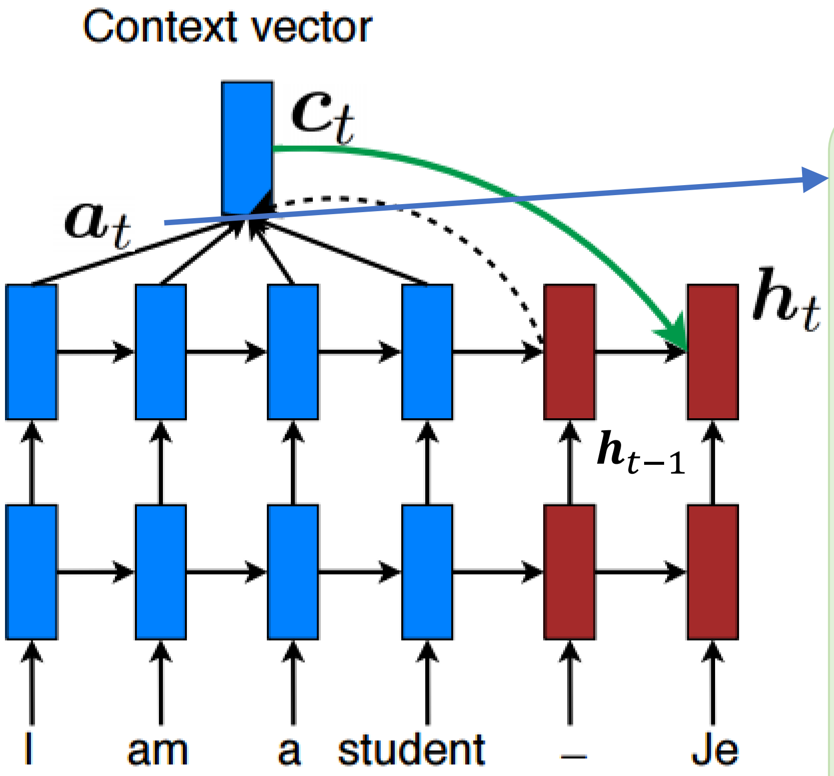


Decoder:
Recurrent language model

Attention Mechanism [Bahdanau et al 14]

Encoded representations

- Attention: $softmax(f_a(\boldsymbol{h}_{t-1}, \overline{\boldsymbol{H}}_s))$



$\overline{H}_S = [\bar{h}_1, \cdots, \bar{h}_n]$

Attention scoring function

$$score\left(\boldsymbol{h}_{t-1}, \overline{\boldsymbol{h}}_s\right)$$
$$= \boldsymbol{v}^T \tanh(\boldsymbol{W}\boldsymbol{h}_{t-1} + \boldsymbol{V}\overline{\boldsymbol{h}}_s)$$

Directly computes a soft alignment

$\Downarrow$ $softmax$

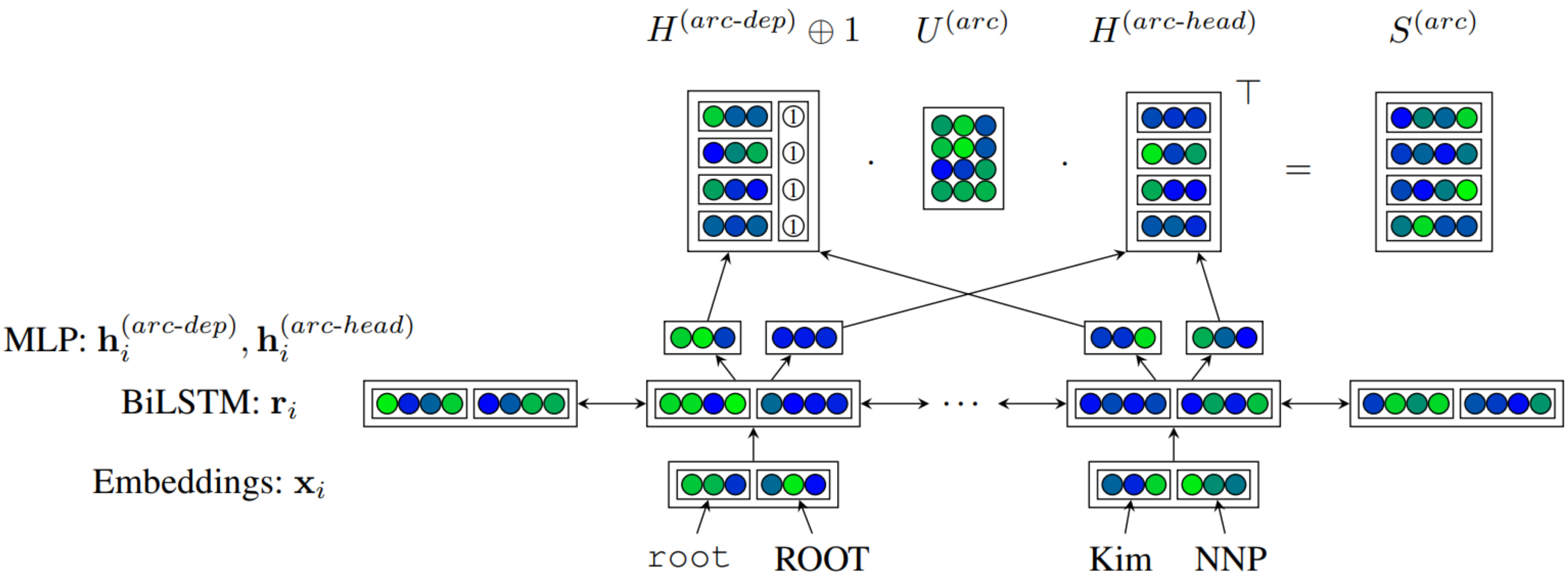
$$\boldsymbol{a}_t(s) = \frac{\exp(score(\boldsymbol{h}_{t-1}, \overline{\boldsymbol{h}}_s))}{\sum_{s'} \exp(score(\boldsymbol{h}_{t-1}, \overline{\boldsymbol{h}}_{s'}))}$$

Expected annotation

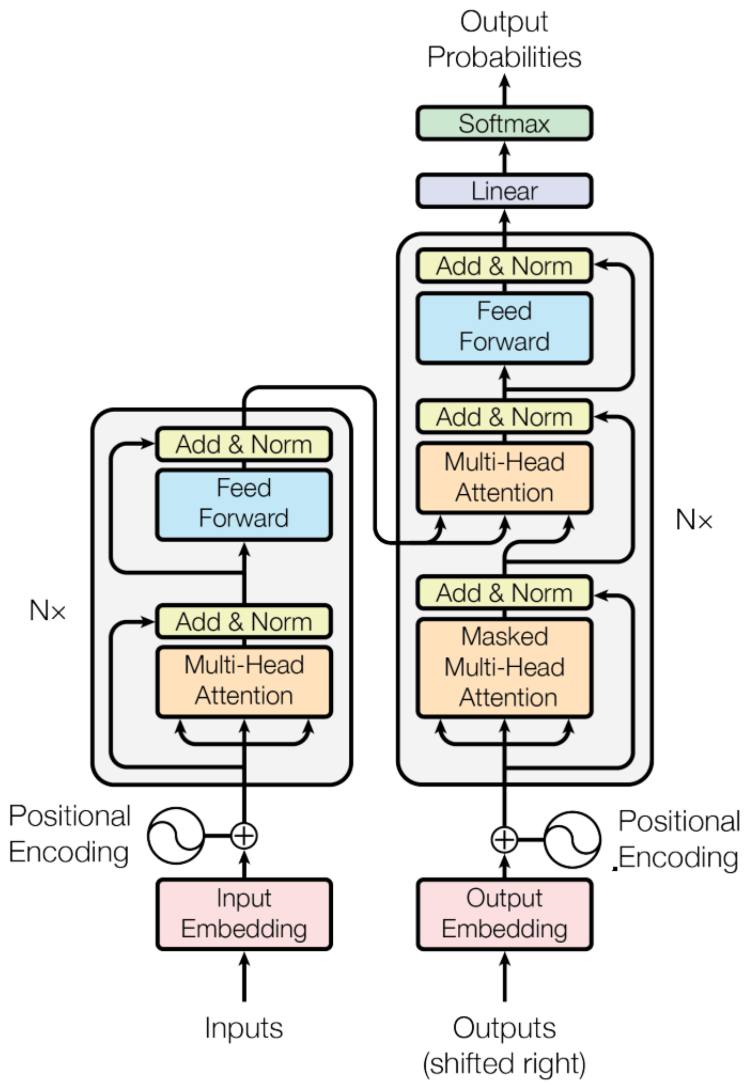
$\overline{\boldsymbol{h}}_s$: a source hidden state

Biaffine Attention [Dozat & Manning 16]

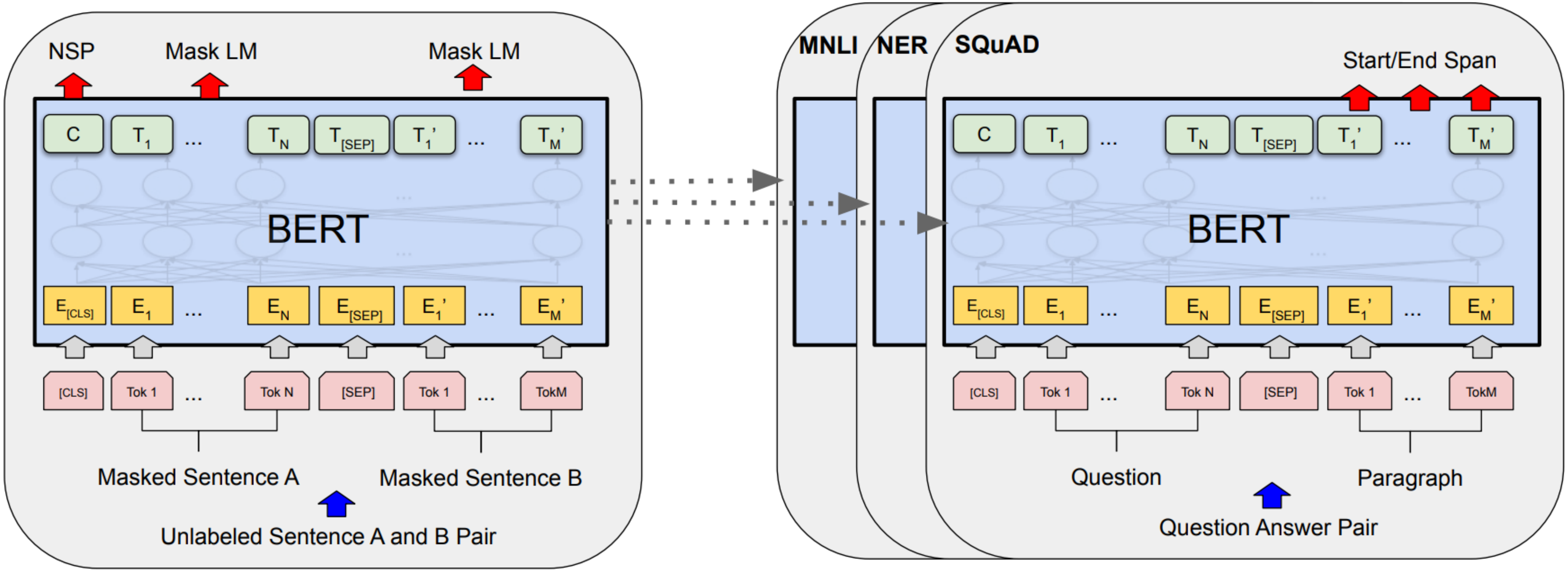
- <Query, Key>-based attention mechanism



Transformer [Vaswani et al '17]



Pretrained Language Models: GPT, BERT



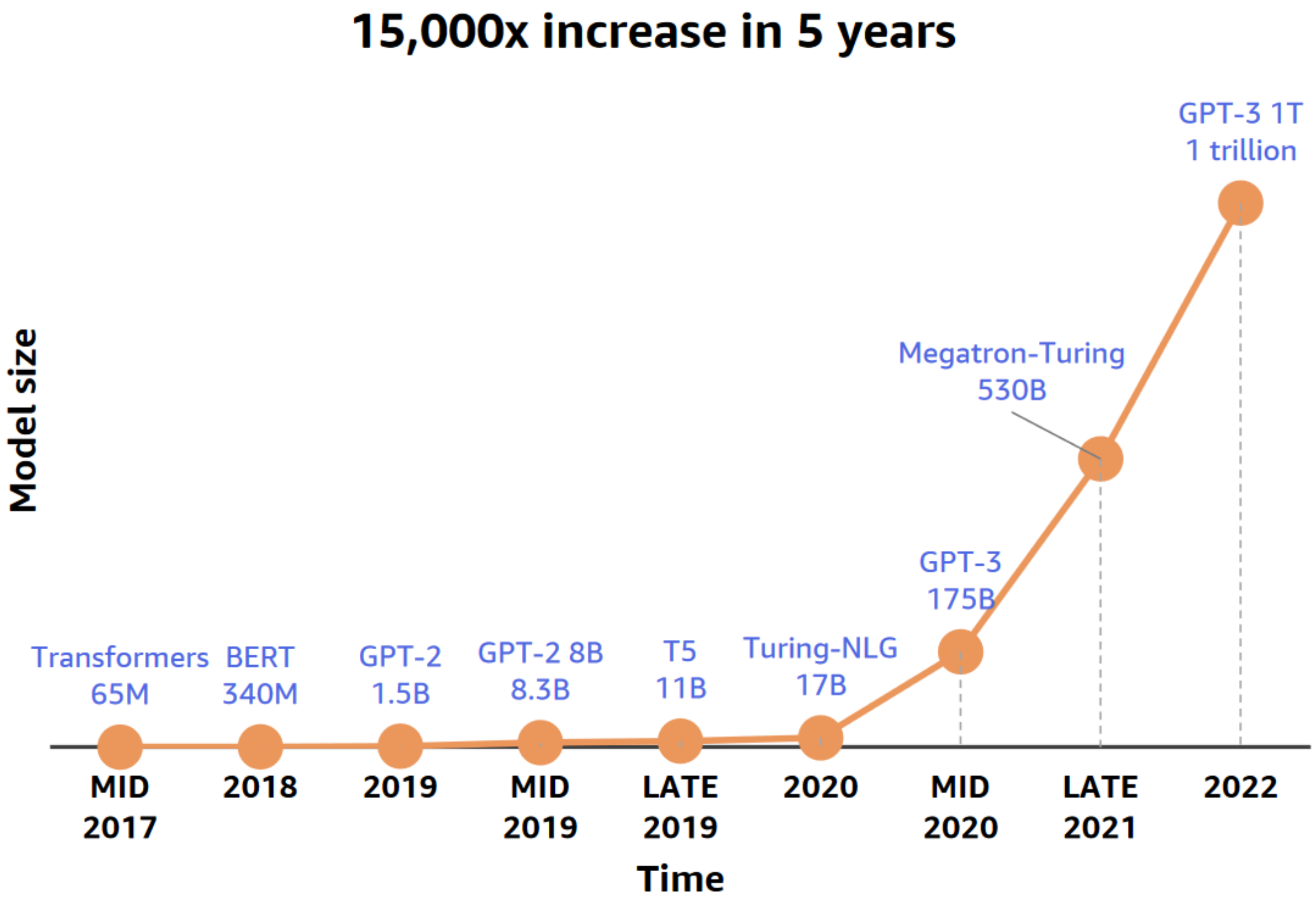
Pre-training

Fine-Tuning

Large Language Models

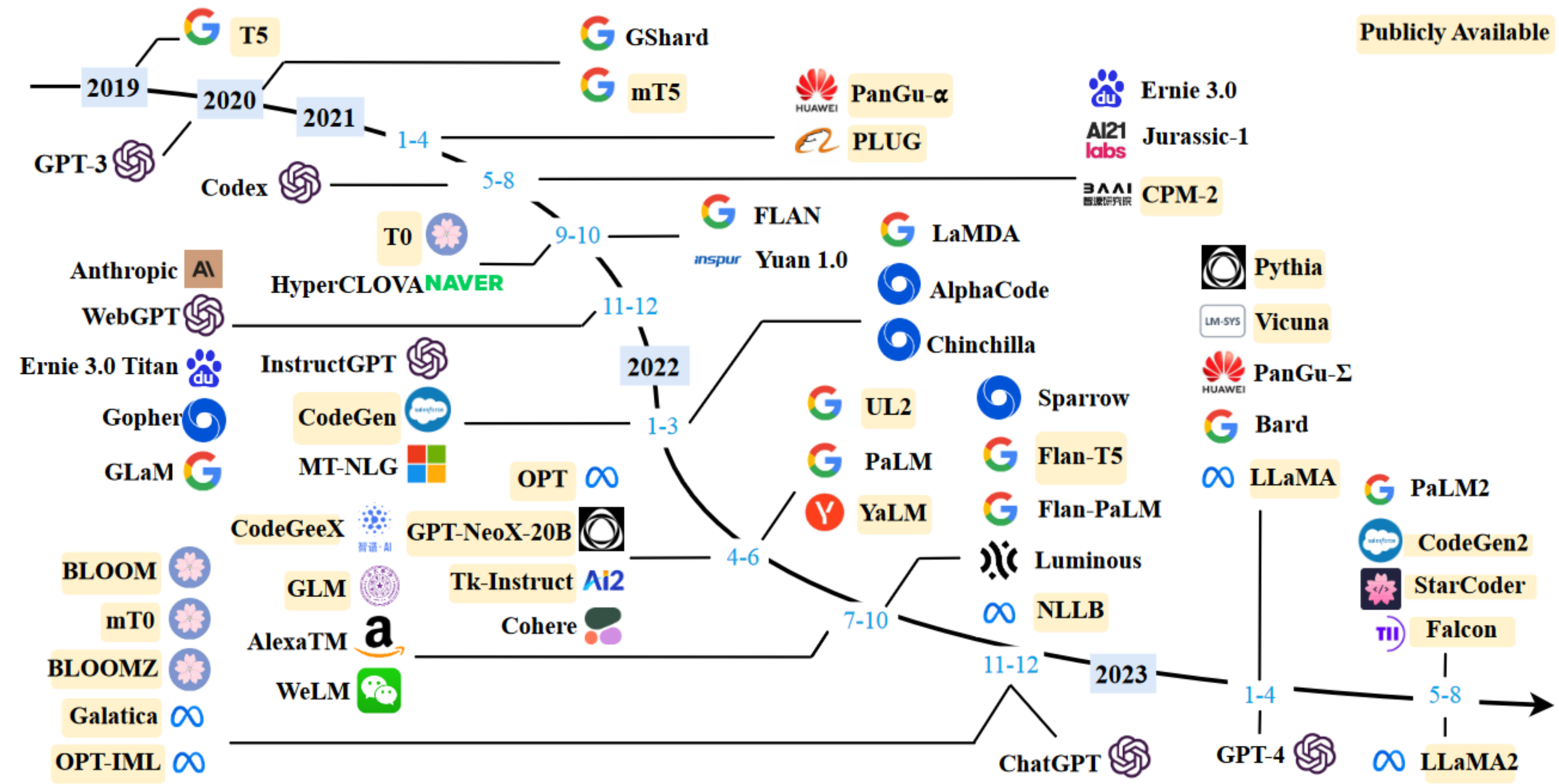
- The era of LLMs: GPT3, Hyperclova
  - Pretrained LMs (e.g., BERT) ➔ LLMs

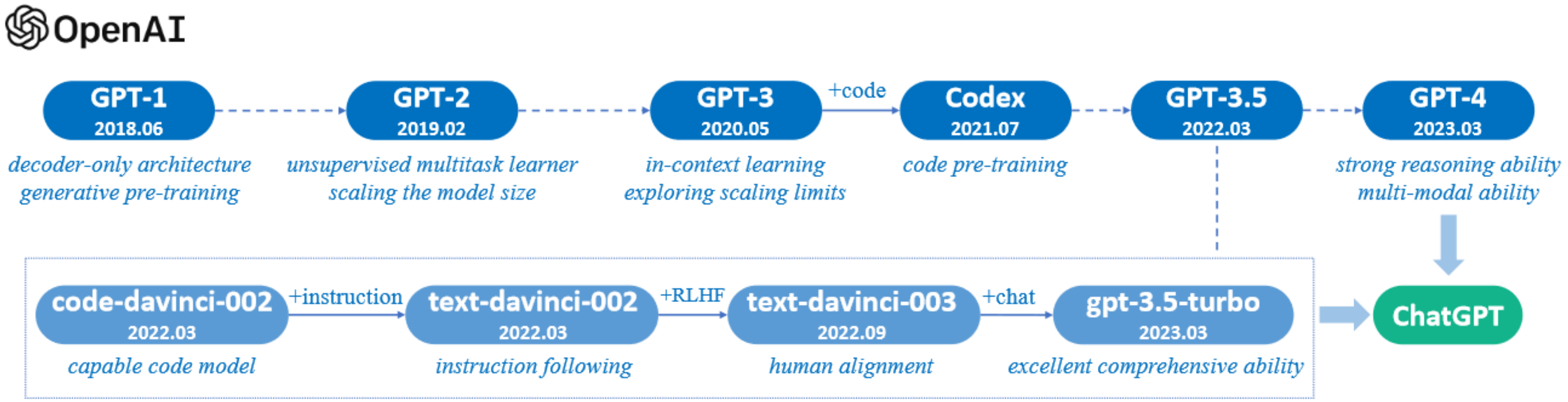
GPT4 > 1T



https://d1.awsstatic.com/events/Summits/reinvent2022/AIM405_Train-and-deploy-large-language-models-on-Amazon-SageMaker.pdf

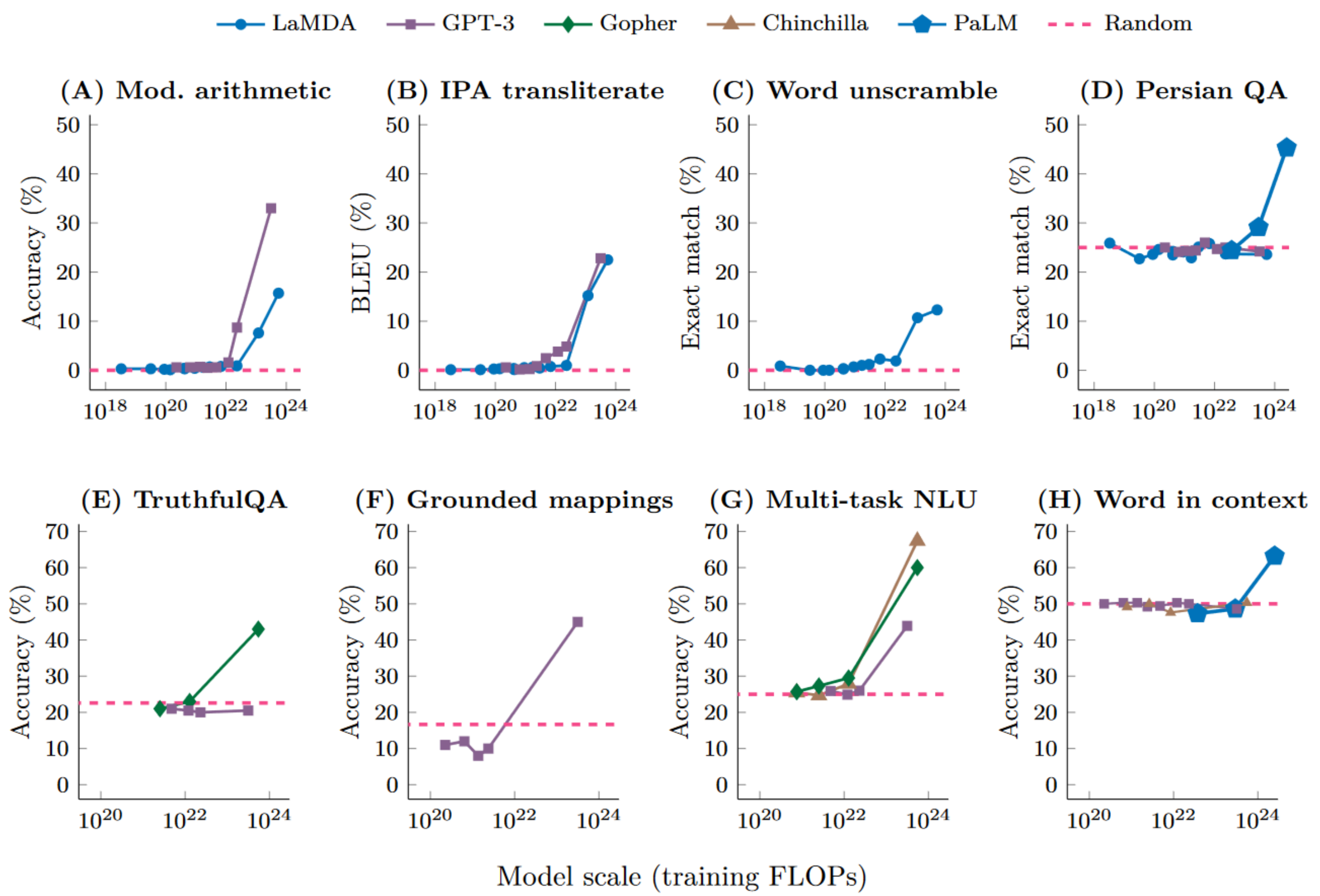
Large Language Models: Timeline





Large Language Models: Emergent Abilities

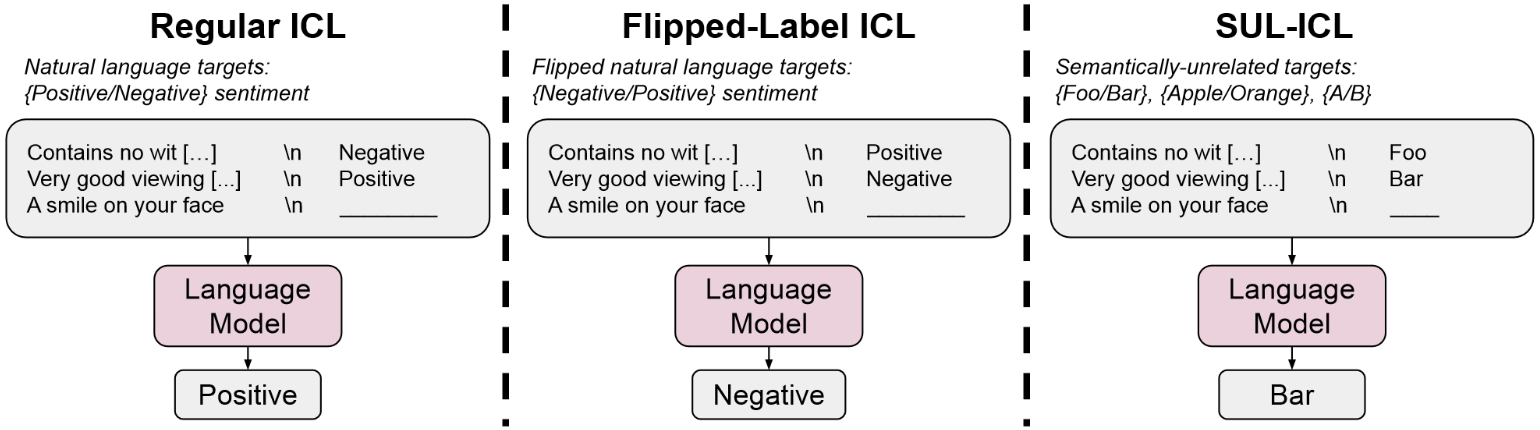
- Emergent Abilities of LLMs: In-context learning, Instruction following, Multi-step reasoning (CoT), etc.
  - Like the phenomenon of **phase transition** in physics

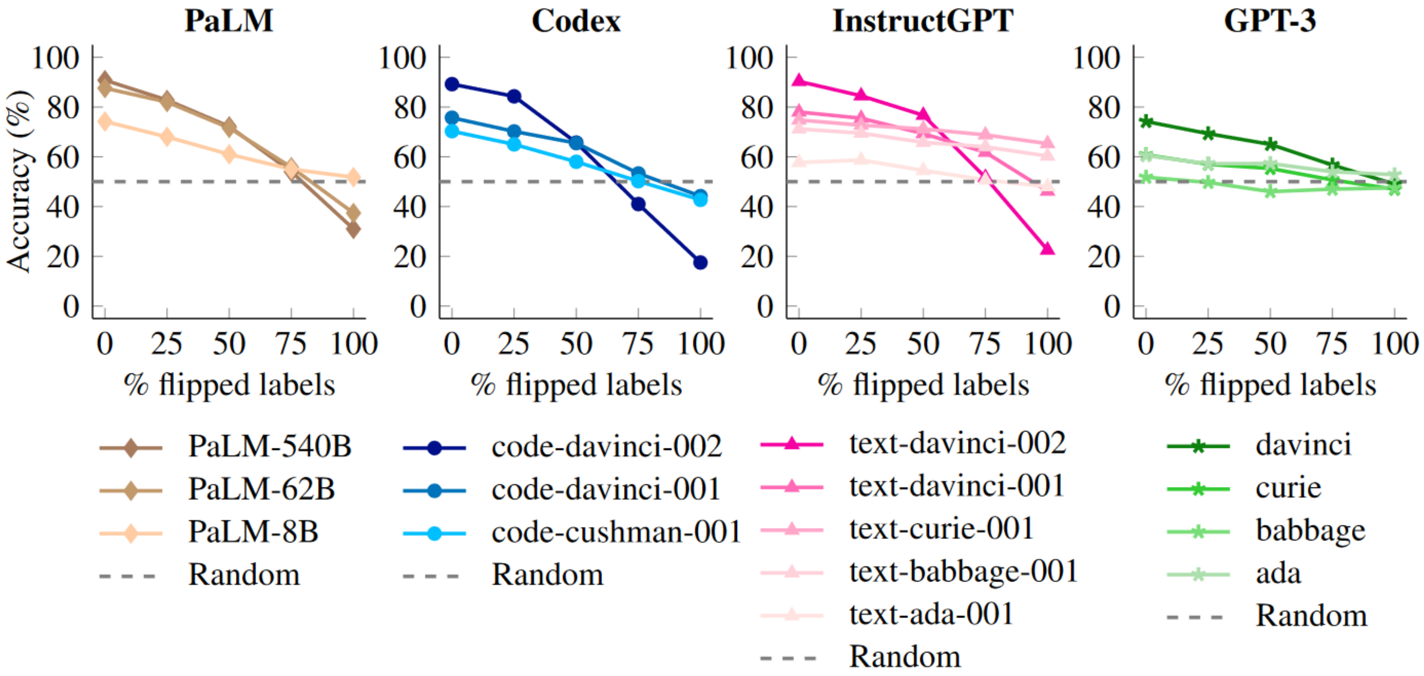


https://arxiv.org/pdf/2206.07682.pdf

Large Language Models: Emergent Abilities

- Larger language models do in-context learning differently [Wei et al '23]

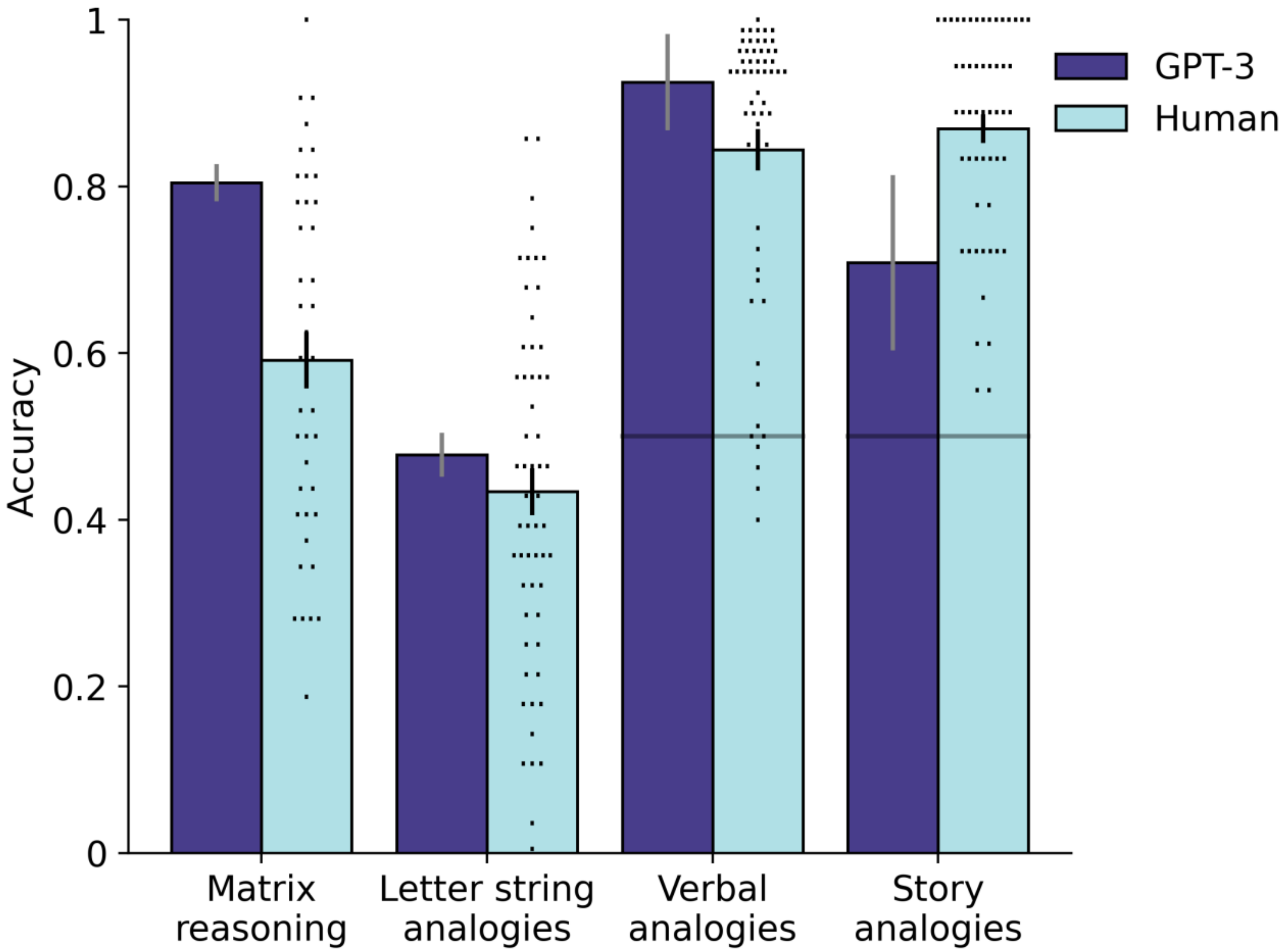




https://arxiv.org/pdf/2303.03846.pdf

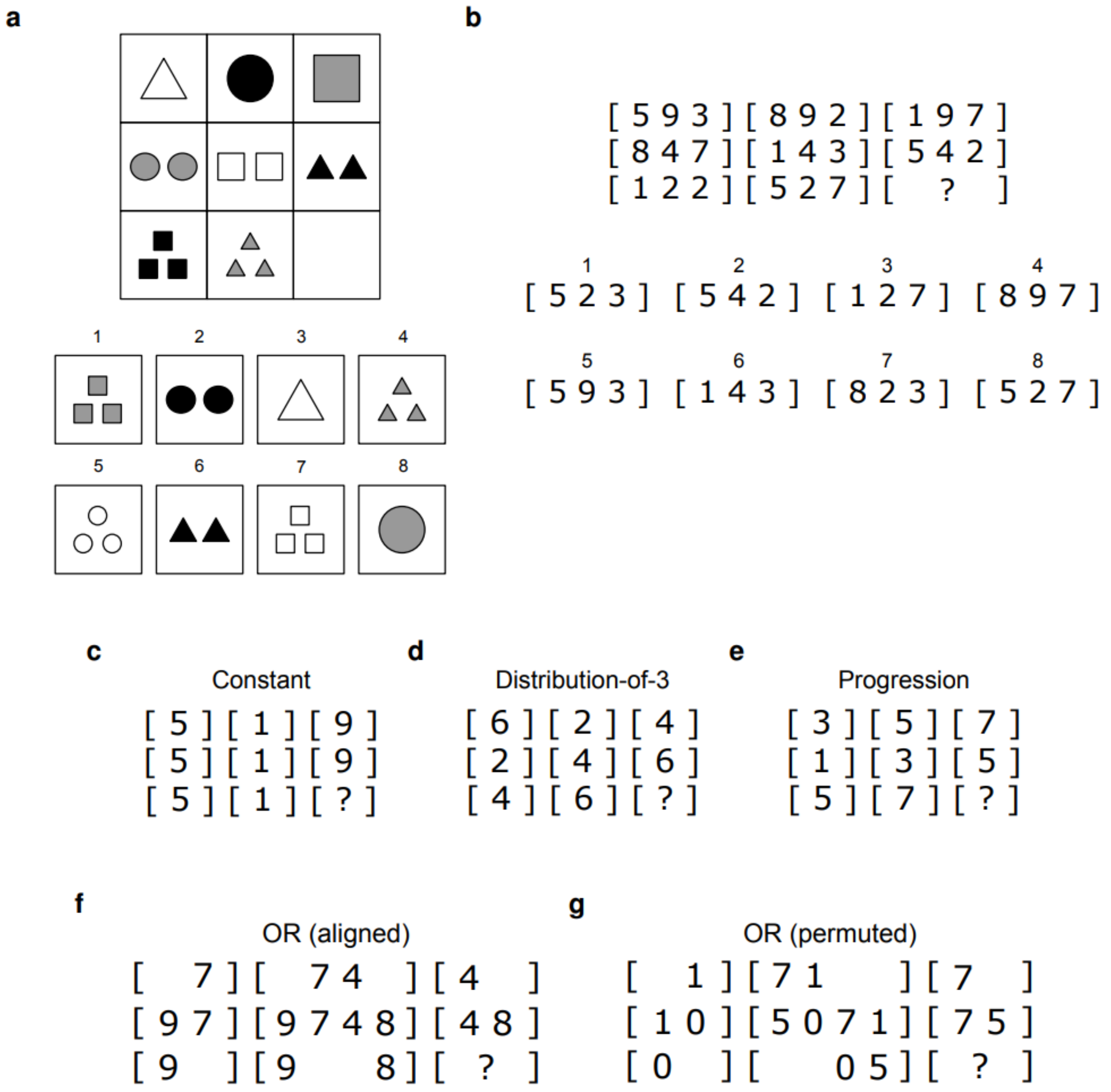
Large Language Models: Emergent Abilities

- Emergent analogical reasoning in large language models [Webb et al '23]



Large Language Models: Emergent Abilities

- Emergent analogical reasoning in large language models [Webb et al ‘23]



Matrix reasoning problems

Large Language Models: Emergent Abilities

- Emergent analogical reasoning in large language models [Webb et al ‘23]

Letter string analogy problems

a

a b c d → a b c e
i j k l → ?

b

a b c d → a b c e
x l x l x k x k x j x j x i x i → ?

c

a b c → a b c
cold cool warm → ?

d

Transformation types

Extend sequence
a b c d → a b c d e

Successor
a b c d → a b c e

Predecessor
b c d e → a c d e

Remove redundant letter
a b b c d e → a b c d e

Fix alphabetic sequence
a b c w e → a b c d e

Sort
a d c b e → a b c d e

e

Generalization types

Letter-to-number
a b c d → a b c e
1 2 3 4 → ?

Grouping
a b c d → a b c e
i i j j k k l l → ?

Longer target
a b c d → a b c e
i j k l m n o p → ?

Reversed order
a b c d → a b c e
l k j i → ?

Interleaved distractor
a b c d → a b c e
i x j x k x l x → ?

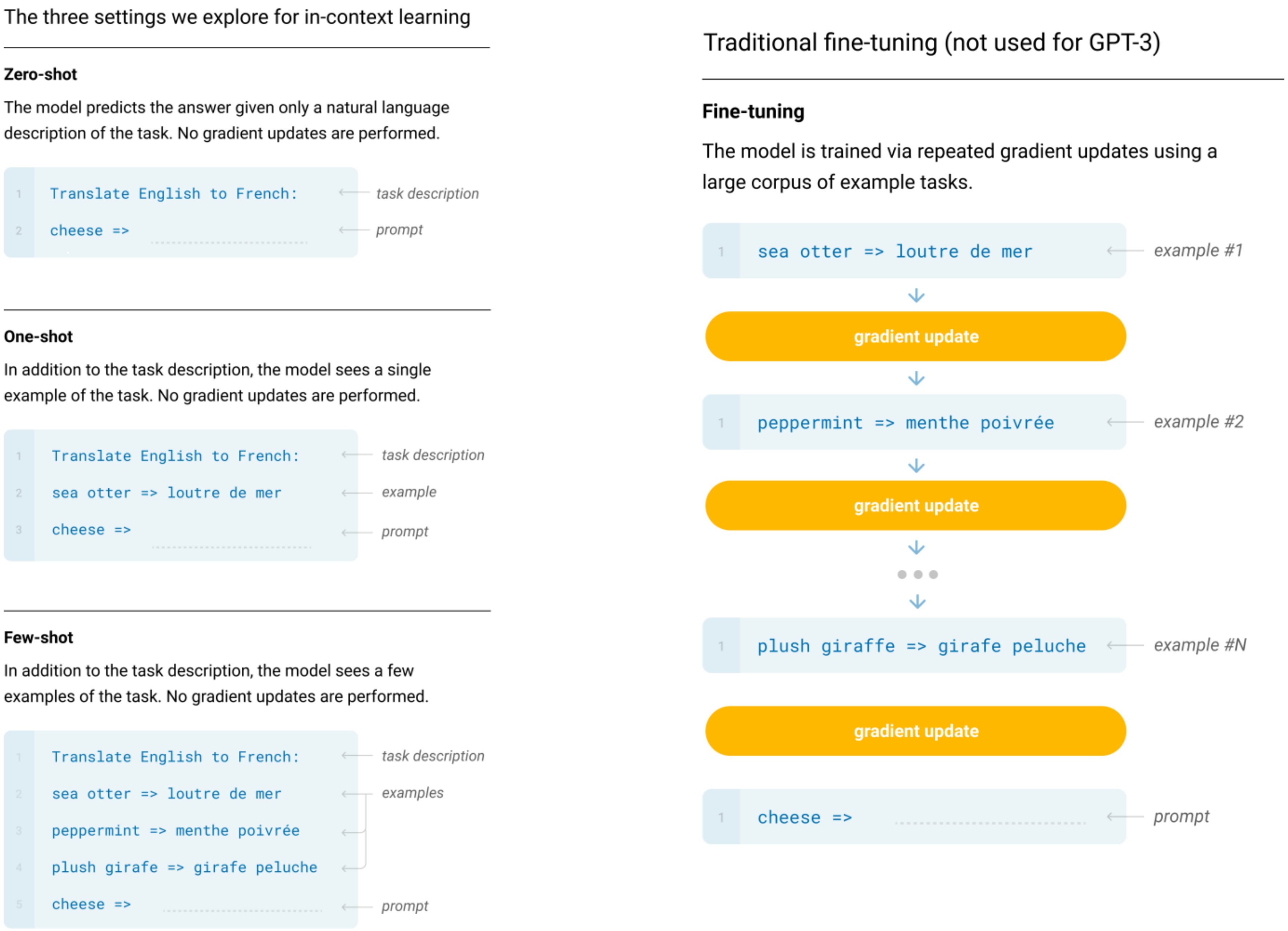
Larger interval
a b c d → a b c e
i k m o → ?

In-Context Learning [Brown et al ’20]

- Language Models are Few-Shot Learners [Brown et al ’20]

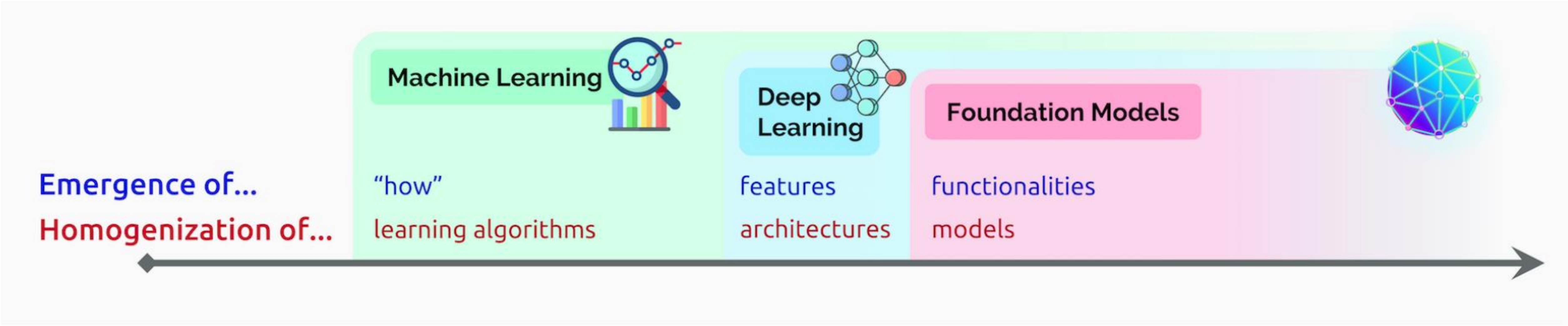
During unsupervised pre-training, a language model develops a broad set of skills and pattern recognition abilities. It then uses these abilities at inference time to rapidly adapt to or recognize the desired task. We use the term “in-context learning” to describe the inner loop of this process, which occurs within the forward-pass upon each sequence

In-Context Learning [Brown et al '20]

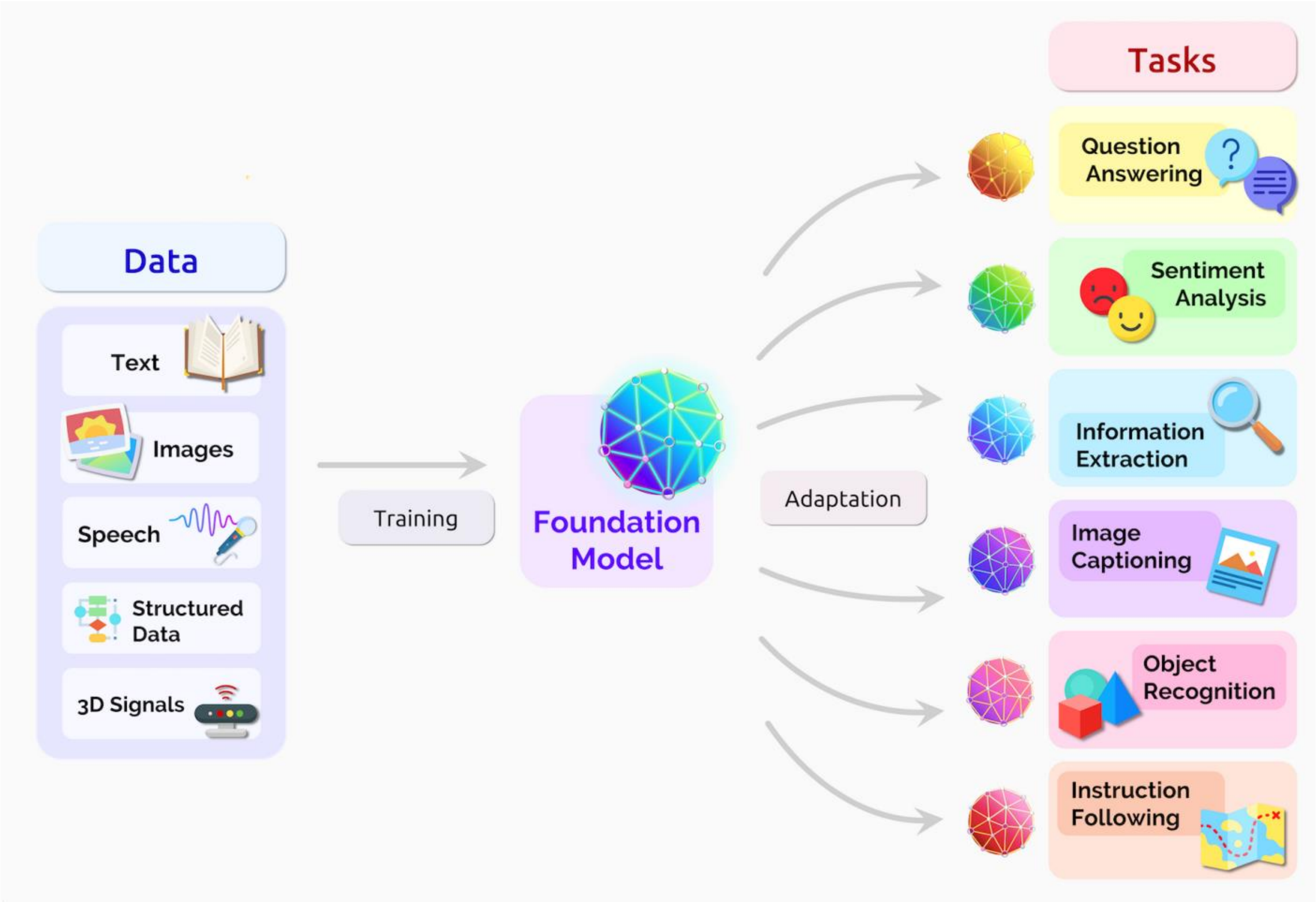


Foundation models [Bommasani et al 21]

- An emerging paradigm for building artificial intelligence (AI) systems based on a general class of models

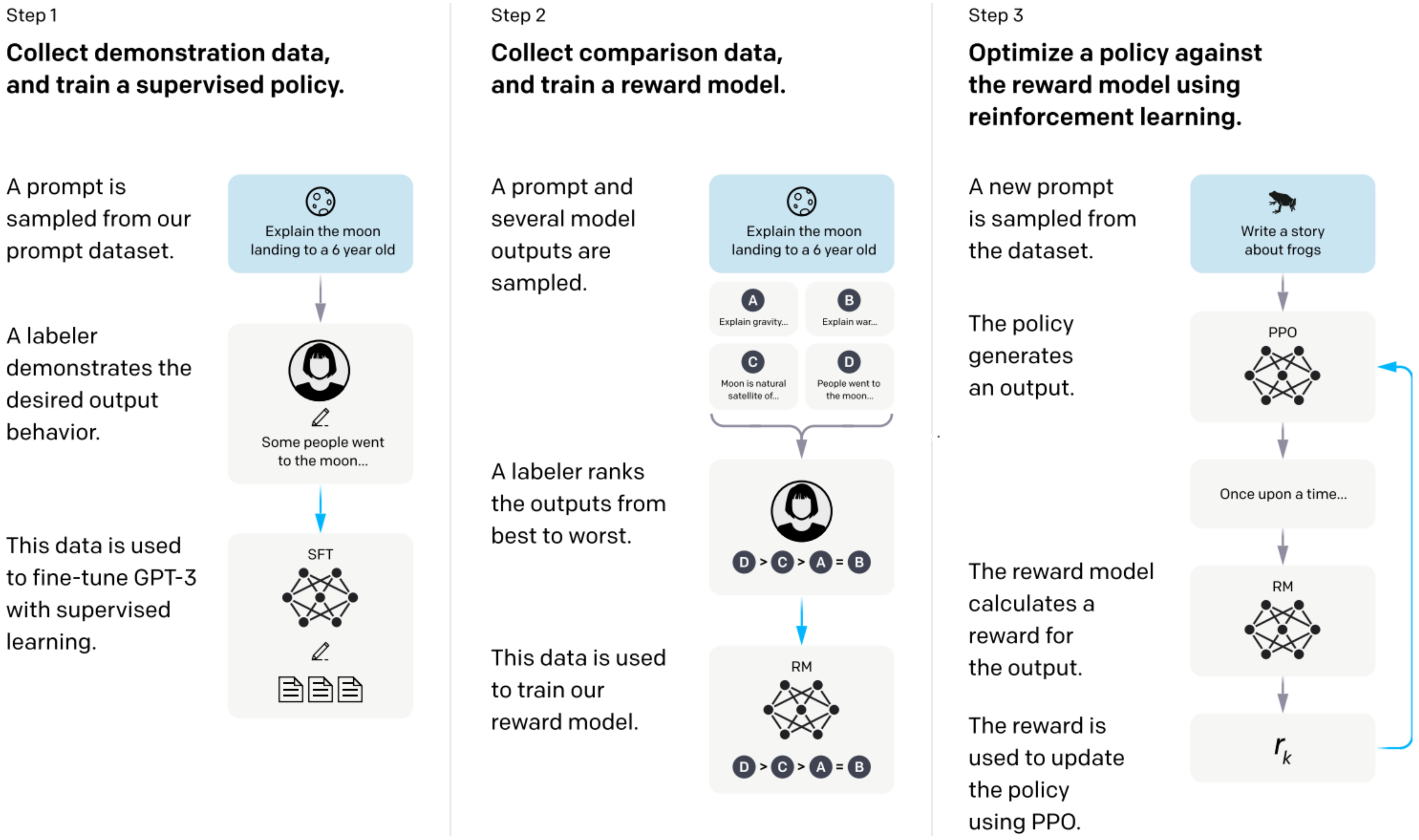


Foundation models [Bommasani et al 21]



A foundation model can centralize the information from all the data from various modalities. This one model can then be adapted to a wide range of downstream tasks.

Human-Aligned LLMs [Ouyang et al '22]



LLM: Challenges

- 1. Reduce and measure hallucinations
  2. Optimize context length and context construction
  3. Incorporate other data modalities
  4. Make LLMs faster and cheaper
  5. Design a new model architecture
  6. Develop GPU alternatives
  7. Make agents usable
  8. Improve learning from human preference
  9. Improve the efficiency of the chat interface
  10. Build LLMs for non-English languages

https://huyenchip.com/2023/08/16/llm-research-open-challenges.html

LLM: Challenges

⚠ Unfathomable Datasets

The size of modern pre-training datasets renders it impractical for any individual to read or conduct quality assessments on the encompassed documents thoroughly.

⚠ Unsustainable Loss Power-Law [256]

Performance increases through larger compute budgets but at a decreasing rate if the model or dataset size is fixed, reflecting a power law with diminishing returns.

⚠ Large Memory Requirements

Fine-tuning entire LLMs requires the same amount of memory as pre-training, rendering it infeasible for many practitioners.

⚠ Tokenizer-Reliance

Tokenizers introduce several challenges, e.g., computational overhead, language dependence, handling of novel words, fixed vocabulary size, information loss, and low human interpretability.

⚠ Overhead of Storing and Loading Fine-Tuned LLMs [213, 311]

When adapting an LLM via full-model fine-tuning, an individual copy of the model must be stored (consuming data storage) and loaded (expending memory allocation, etc.) for each task.

⚠ Full Matrix Multiplications

Parameter-efficient fine-tuning of LLMs still requires computing full forward/backward passes throughout the whole network.

https://arxiv.org/pdf/2307.10169.pdf

LLM: Challenges

High Inference Latency [431, 605]

LLM inference latencies remain high because of low parallelizability and large memory footprints.

Prompt Brittleness [675, 596, 342]

Variations of the prompt syntax, often occurring in ways unintuitive to humans, can result in dramatic output changes.

Misaligned Behavior

LLMs often generate outputs that are not well-aligned with human values or intentions, which can have unintended or negative consequences.

Limited Context Length

Limited context lengths are a barrier for handling long inputs well to facilitate applications like novel or textbook writing or summarizing.

Hallucination [293, 458, 241]

Generated text that is fluent and natural but unfaithful to the source content (intrinsic) and/or under-determined (extrinsic).

Isolated Model Updates without Side-Effects [205]

Updating isolated model behavior or factual knowledge can be expensive and untargeted, which might cause unintended side-effects.

https://arxiv.org/pdf/2307.10169.pdf

LLM: Challenges

⚠ Brittle Evaluations

Slight modifications of the benchmark prompt or evaluation protocol can give drastically different results.

⚠ Detecting LLM-generated Text

The difficulty in classifying whether a text is LLM-generated or written by a human.

⚠ Tasks Not Solvable By Scale

Tasks *seemingly* not solvable by further data/model scaling.

⚠ Reliance on Static, Human-Written Ground Truth

Static benchmarks become less useful over time due to changing capabilities while updating them often relies on human-written ground truth.

⚠ Paraphrasing Attacks

Another LLM can rewrite LLM-generated text to preserve approximately the same meaning but change the words or sentence structure.

⚠ Uncontrolled Experiments

Papers presenting novel LLMs often lack controlled experiments, likely due to the prohibitive costs of training enough models.

https://arxiv.org/pdf/2307.10169.pdf

LLM: Challenges

⚠ Curse of (Design) Dimensionality

Common design spaces of LLM experiments are high-dimensional.

⚠ Irrepeatable Training Runs

Parallelism strategies designed to distribute the training process across many accelerators are typically non-deterministic, rendering LLM training irreproducible.

⚠ Irreproducible API Inference

API-served models are often irreproducible.

⚠ Maintaining Coherence

Multi-turn interactions make Chatbots easily "forget" earlier parts of the conversation or repeat themselves [53, 451].

⚠ Transfer to Downstream Applications

The ultimate objective of protein language models is to deploy them in real-world projects such as drug design. Evaluations often target smaller and/or specialized datasets, not considering how the models could contribute to protein design in vitro or in vivo.

https://arxiv.org/pdf/2307.10169.pdf

LLM: Challenges

High Inference Latency

High inference latency (Sec. 2.5) hinders the user experience [397], especially in multi-turn interaction with chatbots.

Long-Range Dependencies [660, 504]

Long-range dependencies across a code repository usually cannot be regarded because of limited context lengths (Sec. 2.6).

Limited Context Window

The largest genomes have vastly longer DNA sequences [390] than existing genomic LLMs' context windows can handle, constraining the types of genomes that can be successfully modeled using these approaches.

Limited Context Window [368, 637]

The inability of current LLMs to keep the entire generated work within the context window currently constrains their long-form applications and generates the need for modular prompting (14).

https://arxiv.org/pdf/2307.10169.pdf

LLM: Challenges

Numerical Reasoning [436, 49]

LLMs have generally seen worse performance on quantitative tasks, potentially constraining their applications in knowledge work areas such as financial services or accounting.

Hallucination and Bias [538, 388, 511]

The safety-critical nature of the medical domain means the possibility of hallucinations significantly limits the current use cases. Further work is also needed to reduce the risk of LLMs perpetuating existing bias in clinical datasets.

Out of Date Information

Due to regularly updated laws and new precedents, the training/retrieval data become outdated frequently [195].

Sub-Human-Performance [562, 607]

Existing LLMs struggle to match human performance on reasoning benchmarks.

Single Modality [338, 14, 564]

While LLMs can help robots or agents understand instructions and add high-level planning capabilities, their inability to directly learn from image, audio or other sensor modalities constrain their applications.

ht

LLM: Challenges

⚠ Social Biases [12, 367]

Unbalanced views and opinions in the training data skew the LLMs towards biased human behaviors.

⚠ Hallucinated Distributions [506]

Using LLMs for fully synthetic data generation is currently constrained by our inability to verify whether the synthetic data generated is representative of the true distribution in the corresponding real-world data.

https://arxiv.org/pdf/2307.10169.pdf

Contents

- Introduction
  - Goal & Tasks & Applications
  - Related disciplines
    - Artificial intelligence, Machine learning, Linguistics, etc.

- Natural language processing: Methods
  - Methods
    - Rule-based approach: ELIZA
    - Statistical method: HMM, PCFG
    - Deep learning: Word2vec, RNN, LSTM, Transformer
    - Pretrained language model: BERT
    - Large language model: GPT3
    - Human-aligned LLM: ChatGPT
    - LLM for X: Multimodal & Robotics, Others

- Course schedule ◀

Course: Textbooks

Speech and Language Processing (

[Dan Jurafsky](#) and [James H. Martin](#)

Here's our Feb 3, 2024 release! We also expect to release Chapter 12 soo

Individual chapters and updated slides are below; here is a single pdf of all the chapters in the Feb 3, 2024

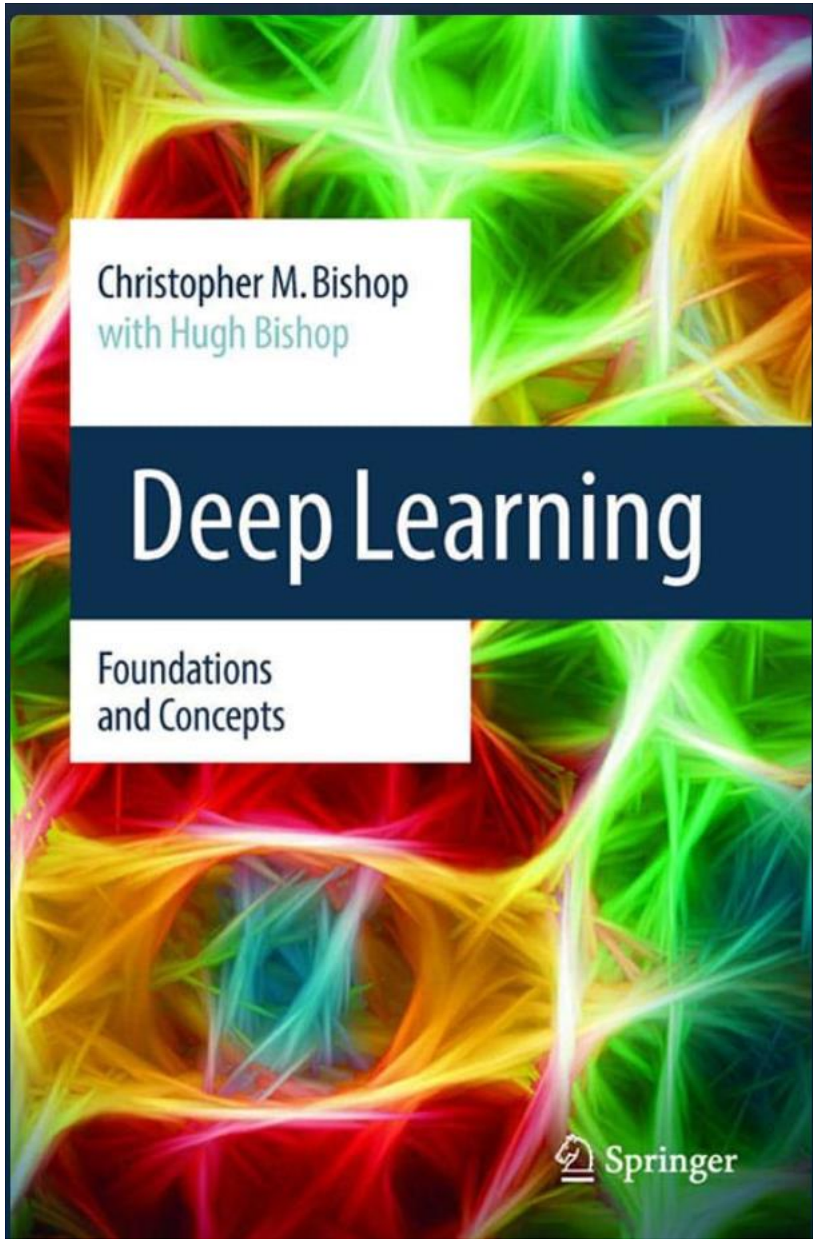
Feel free to use the draft chapters and slides in your classes, the resulting feedback we get from you make
As always, typos and comments very welcome (just email slp3edbugs@gmail.com and let us know the da
indvidual chapter pdfs, those are fixed in the full book draft)

We've put up a list here of the amazing people who have sent so many fantastic suggestions and bug-fi
without you!

When will the whole book be finished? Don't ask.

If you need last year's Jan 2023 draft chapters, they are here;

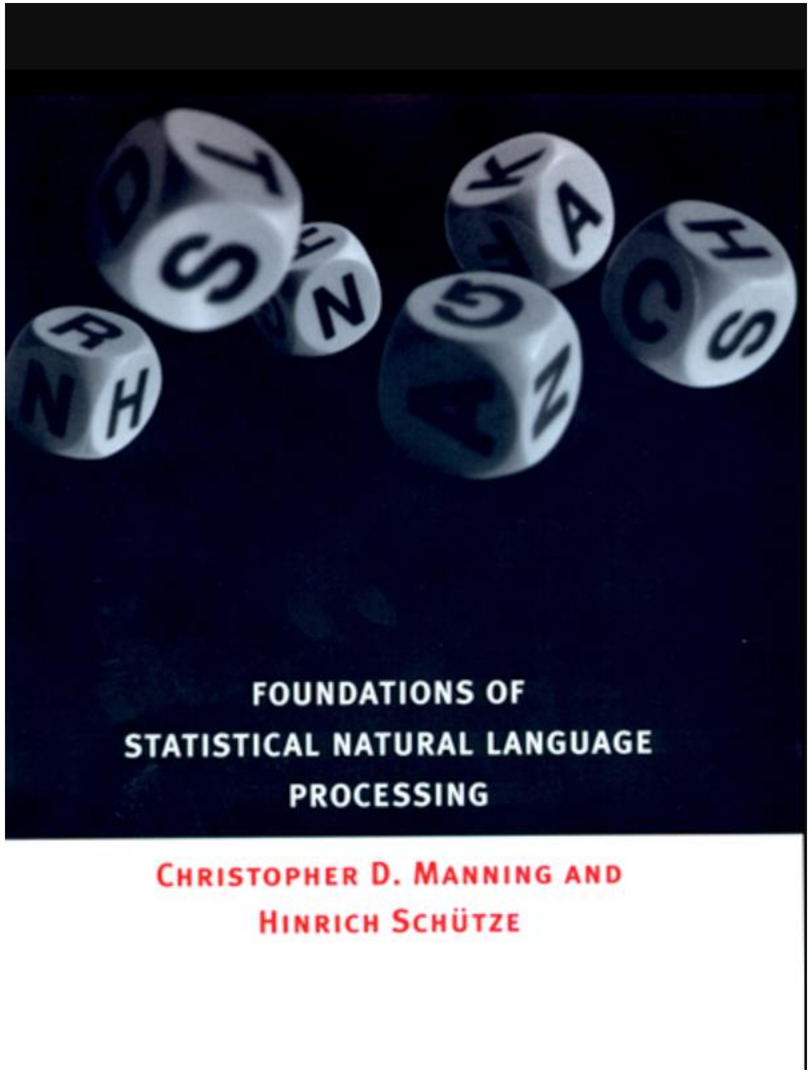
	Chapter	Slides
	Part I: Fundamental Algorithms	
1:	Introduction	
2:	Regular Expressions, Text Normalization, Edit Distance	2: Text Processing [pptx] [pdf] 2: Edit Distance [pptx] [pdf]
3:	N-gram Language Models	3: [pptx] [pdf]
4:	Naive Bayes, Text Classification, and Sentiment	4: [pptx] [pdf]
5:	Logistic Regression	5: [pptx] [pdf]
6:	Vector Semantics and Embeddings	6: [pptx] [pdf]
7:	Neural Networks and Neural Language Models	7: [pptx] [pdf]
8:	Sequence Labeling for Parts of Speech and Named Entities	8: (Intro only) [pptx] [pdf]
9:	RNNs and LSTMs	
10:	Transformers and Large Language Models	



https://web.stanford.edu/~jurafsky/slp3/

https://www.bishopbook.com/

Course: Textbooks



Natural Language Processing

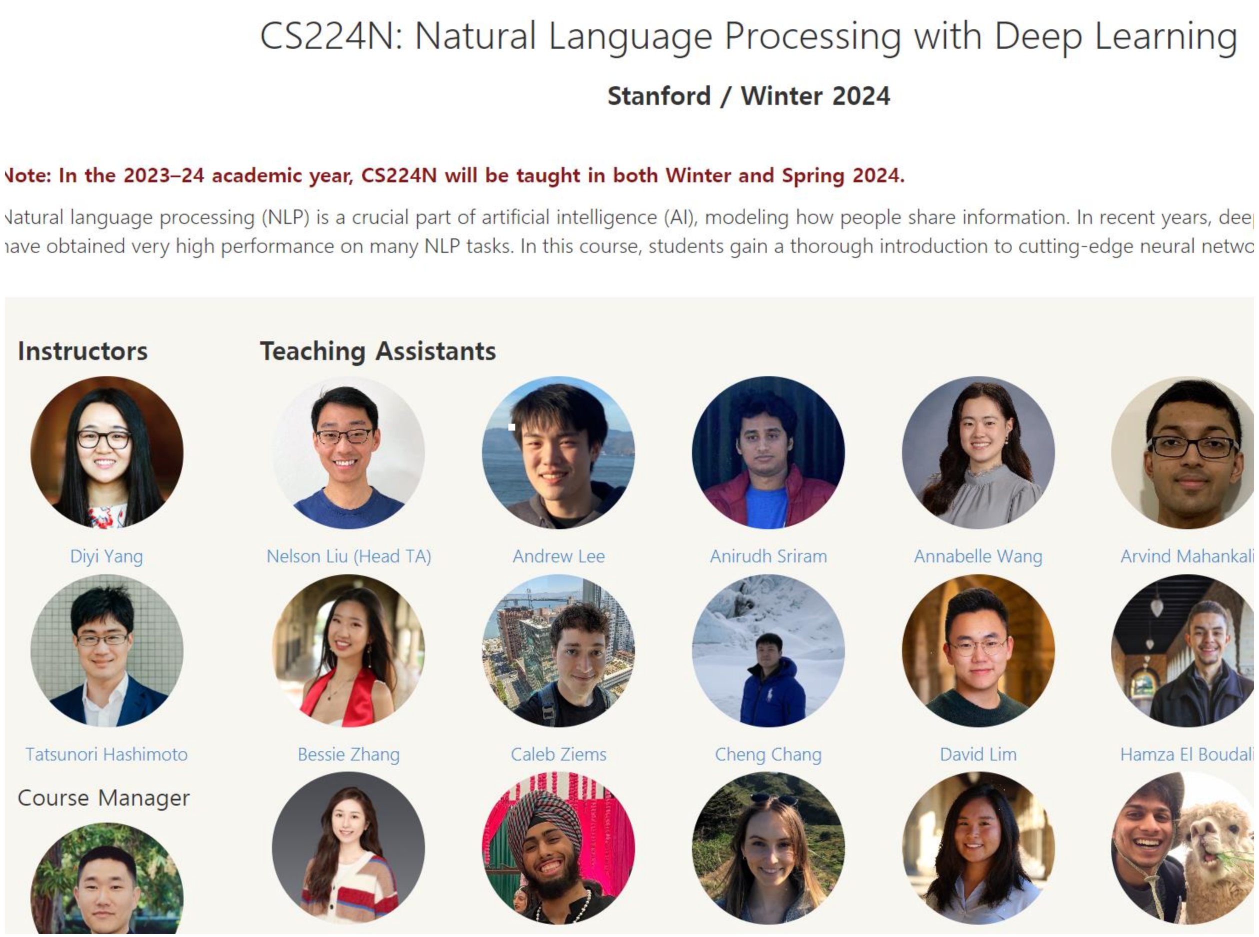
Jacob Eisenstein

November 13, 2018

https://nlp.stanford.edu/fsnlp/

https://github.com/jacobeisenstein/gt-nlp-class/blob/master/notes/eisenstein-nlp-notes.pdf

Course: Textbooks



https://web.stanford.edu/class/cs224n/

Course Schedule

- Introduction
- N-gram Language Models
- Markov Models
- Sequence Labeling for Parts of Speech and Named Entities
- Vector Semantics and Embeddings
- Neural Networks and Neural Language Models
- RNNs and LSTMs
- Sequence to Sequence Models and Machine Translation

Course Schedule

- Transformers
- Pretrained Language Models
- Large Language Models & In-Context Learning & Prompting
- Retrieval-augmented Language Models
- Multimodal Language Models
- Final term