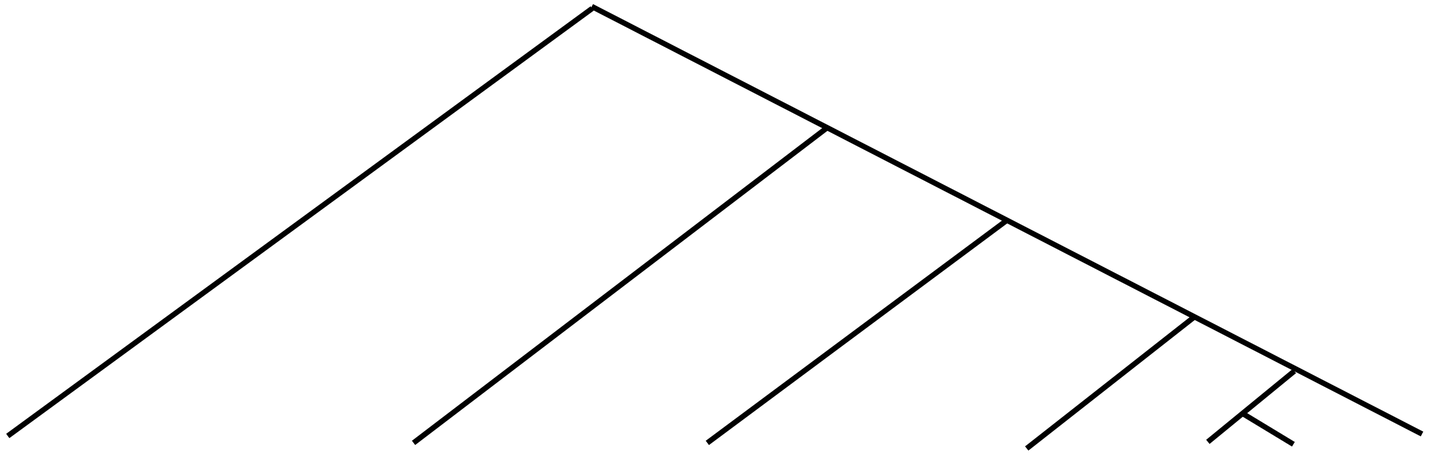


Intro to Intro to NLP

CS 505: Introduction to Natural Language Processing
Spring 2026
Boston University

Course Staff

Instructor: Aaron Mueller

Assistant Professor, CS and CDS

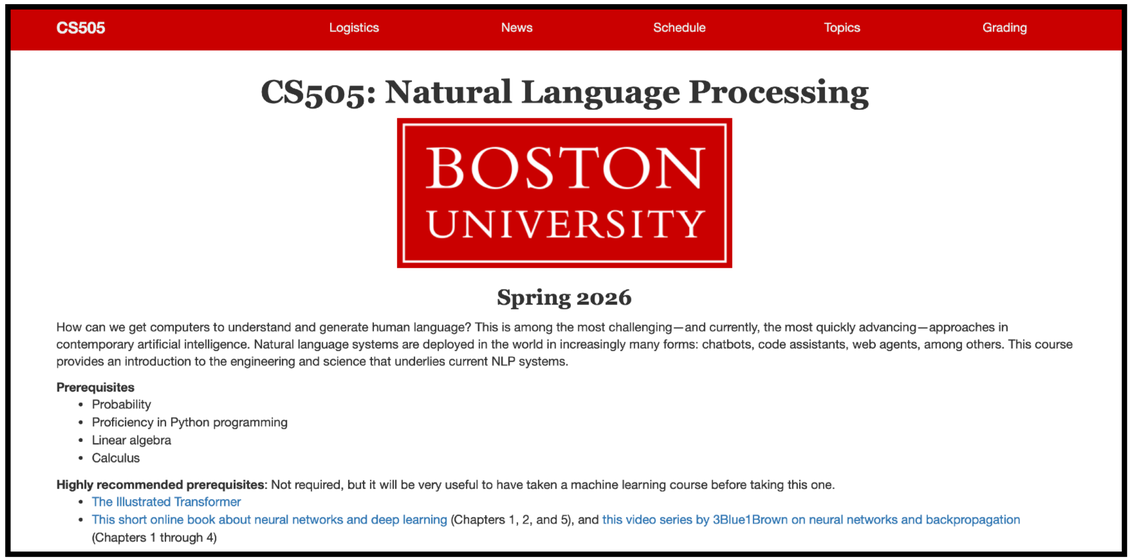
Teaching Fellow: Ge Gao

PhD student, CS

Syllabus & Communication

- **Communication:**
  - Announcements on webpage and in class
  - Questions? Use **Piazza**!
  - (Email is ok when needed, but we'll probably respond faster on Piazza.)

https://aaronmueller.github.io/teaching/cs505_spring26/home.html



- **Course Technology:**
  - Sites: Gradescope and Piazza
  - Some autograded coding assignments, manually graded written and coding assignments

Gradescope entry code: **WNNEKG**

Piazza: link on syllabus. Entry code: **5uzfqw5vwqd**

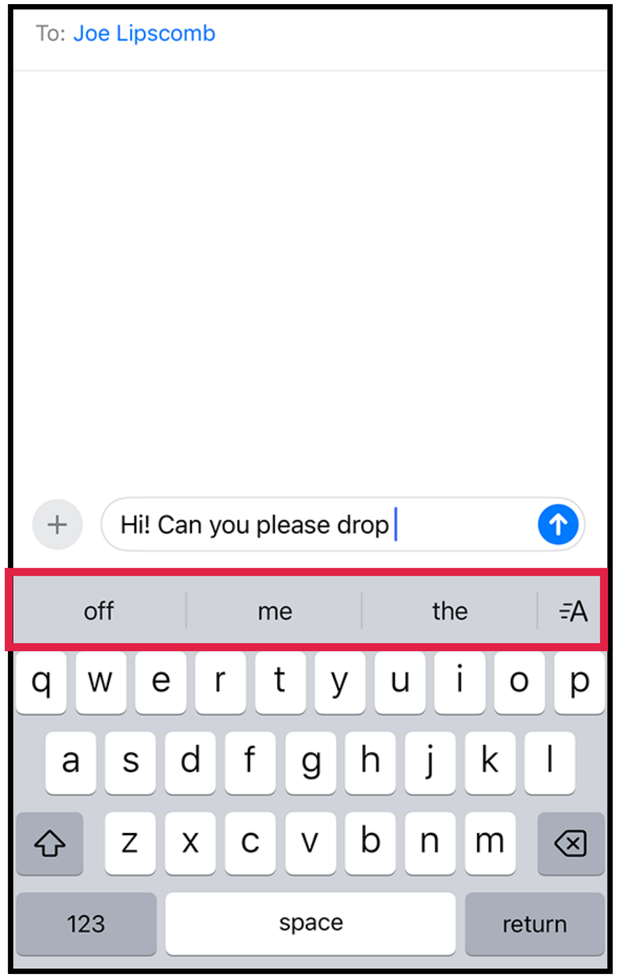
What can we do with NLP?

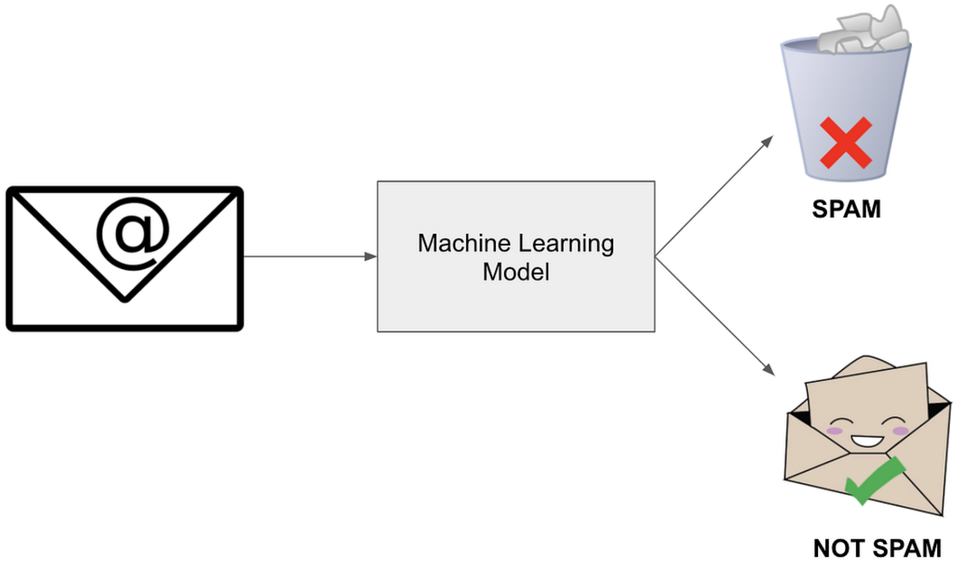
ChatGPT

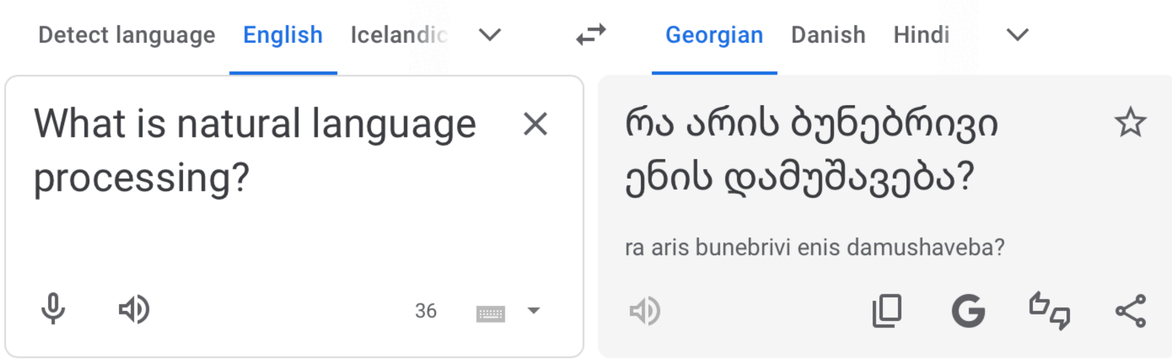
Claude

Gemini

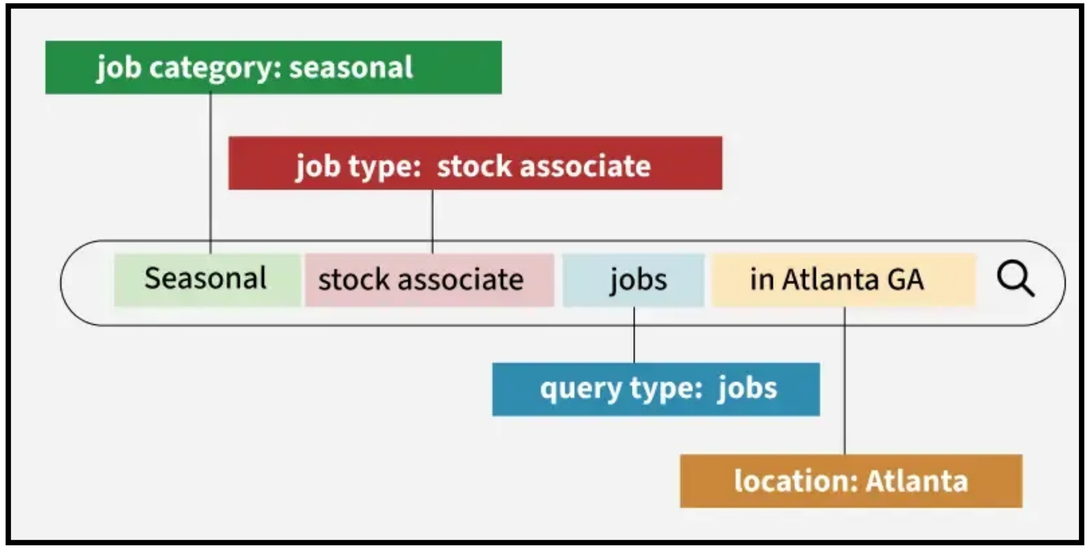
Language model: A system that assigns probabilities to sequences of words.

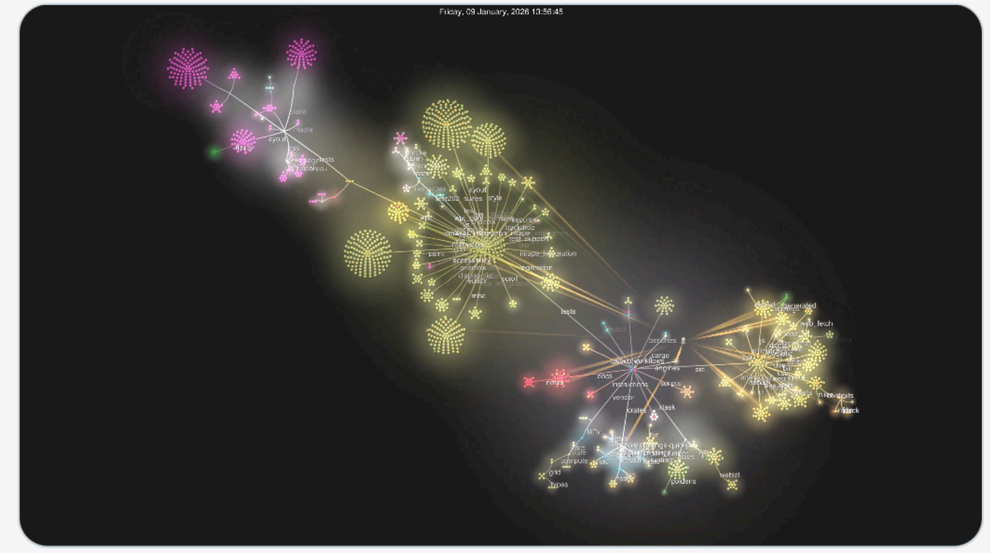






What can we do with NLP?





https://x.com/mntruell/status/2012825801381580880?s=20

home.html
1,192 lines
HTML

Can you summarize the important parts of this syllabus? It's too long tbh

Here's a quick breakdown of the key info:

Key Policies

- **6 free late days** to use as you want (no questions asked)
- AI tools allowed for homeworks and final project, but NOT for the exam
- For the final project: use AI as a tool (debugging, editing), not a crutch (generating entire reports/code)

Main Topics You'll cover language models (from n-grams to transformers to LLMs), training methods (pre-training, fine-tuning, RLHF), NLP tasks (translation, QA, classification), linguistics (syntax, semantics), and current topics like interpretability, bias, and multimodal NLP.

The textbook (Jurafsky & Manning) is free online and optional but helpful for exam prep.

What is NLP?

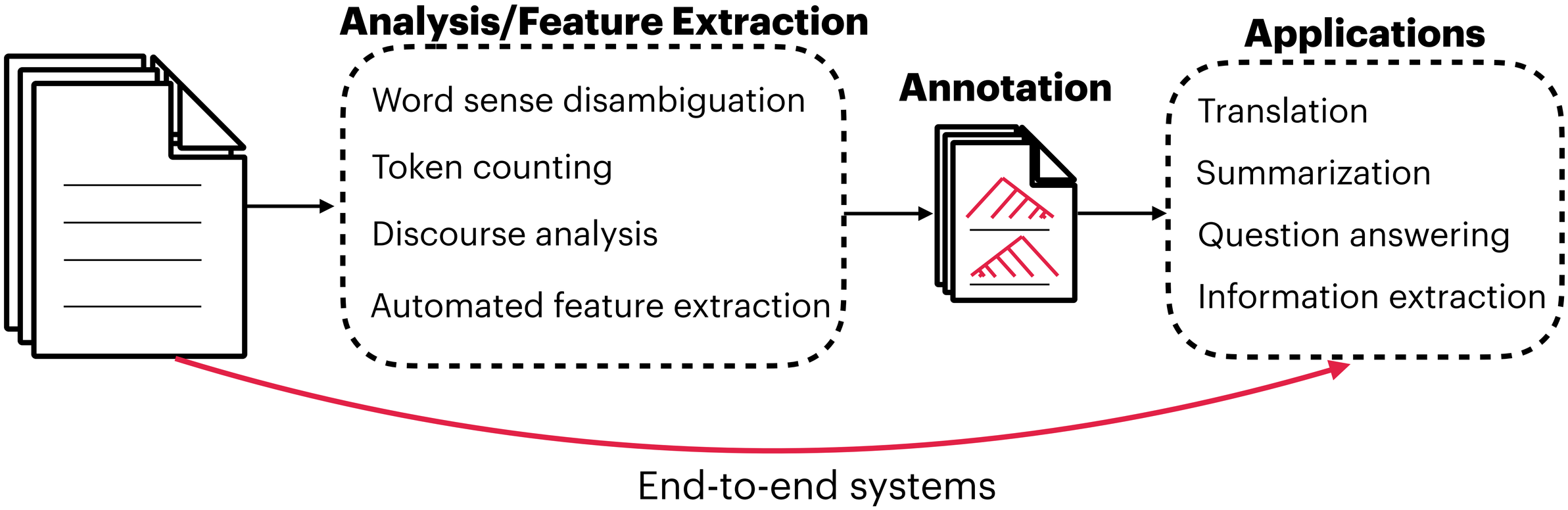
NLP refers to technology to handle human language (usually text) using computers.

The goal of natural language processing (NLP) systems is to solve problems that require **deep understanding** of language. This includes:

- Machine translation, text-to-speech (and vice versa), summarization
- Information extraction, question answering

- Another goal is to **aid communication and productivity:**
  - Spell checking, assisted writing, code generation

The NLP Pipeline



What makes NLP hard?

Language is fundamentally ambiguous.

The girl saw the boy with a telescope.

What makes NLP hard?

Language is fundamentally ambiguous.

The girl [saw [the boy] [with a telescope]].

What makes NLP hard?

Language is fundamentally ambiguous.

The girl [saw [the boy with a telescope]]

PREP

Time flies like an arrow.

VERB

Fruit flies like a banana.

INTJ

Like, why?

head of state *part of body*

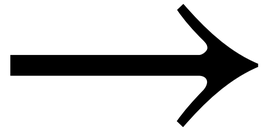
Iraqi Head Seeks Arms

weapons

part of body

What makes NLP hard?

Language is fundamentally ambiguous.

Il fait pas beau. 

It’s not very nice out.

The weather is not good.

It is not a beautiful day.

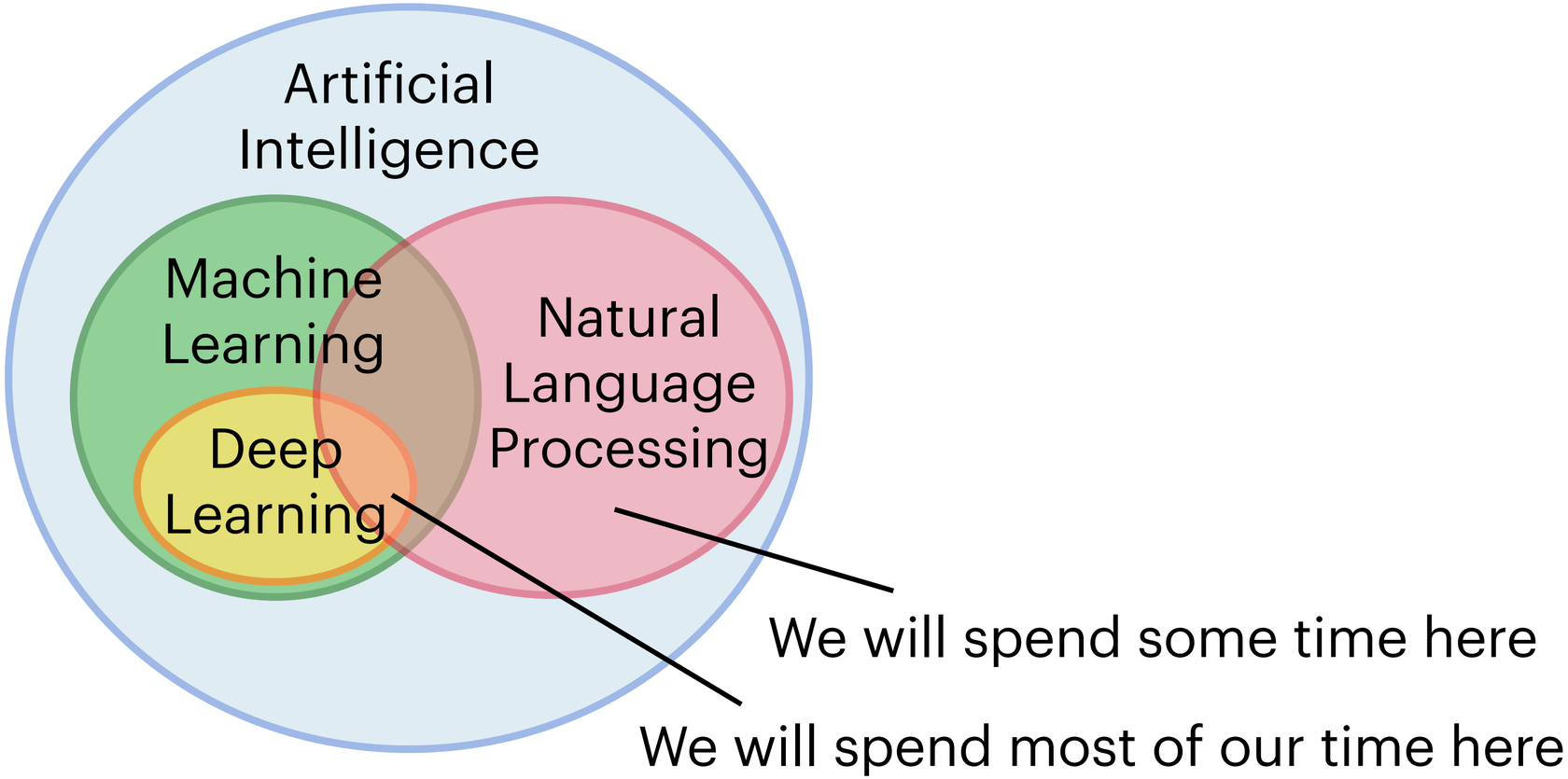
It makes not beautiful.

He does not make beautiful.

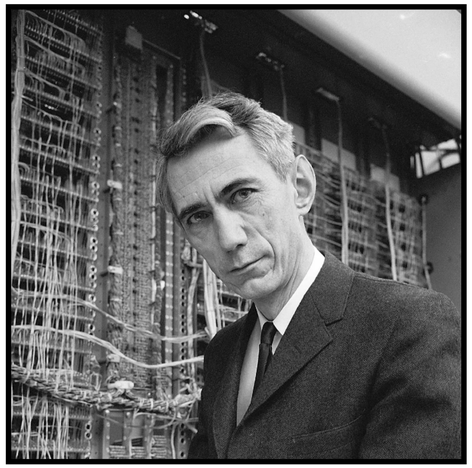
He fact not pretty.

What's the difference between NLP and machine learning?

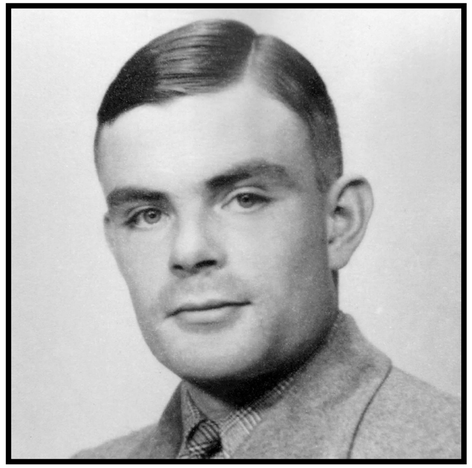
These days, NLP and machine learning are heavily intertwined.
However, they do not completely (or even mostly) overlap!



History



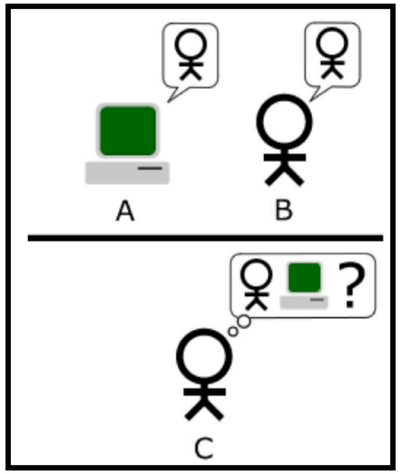
1948: Claude Shannon proposes the mathematical idea of language models



1950: Alan Turing proposes the Turing test

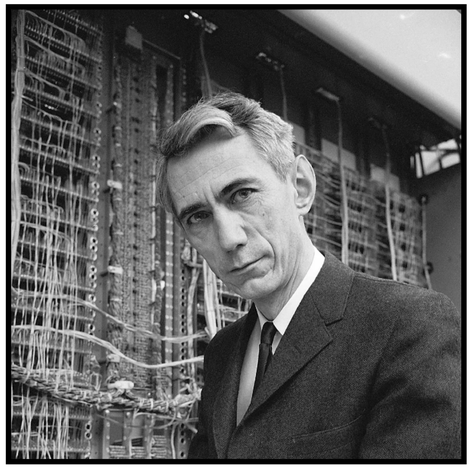
The **Turing test** was the first test of artificial intelligence.

Idea: there are two entities A and B behind a wall, one a computer and one a human. A human interrogator C asks questions of A or B, and if, after some reasonable amount of time, C can't distinguish A and B, then the machine may be considered intelligent.

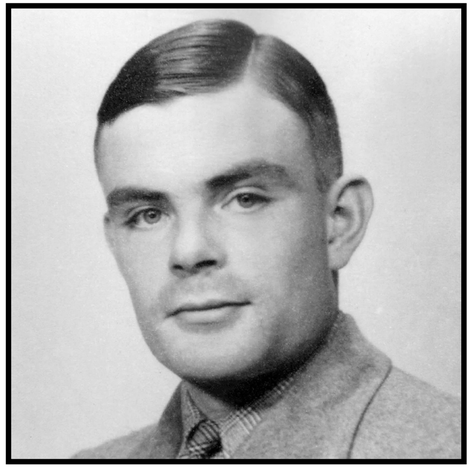


1950 1960 1970 1980 1990 2000 2010 2020

History



1948: Claude Shannon proposes the mathematical idea of language models



1950: Alan Turing proposes the Turing test

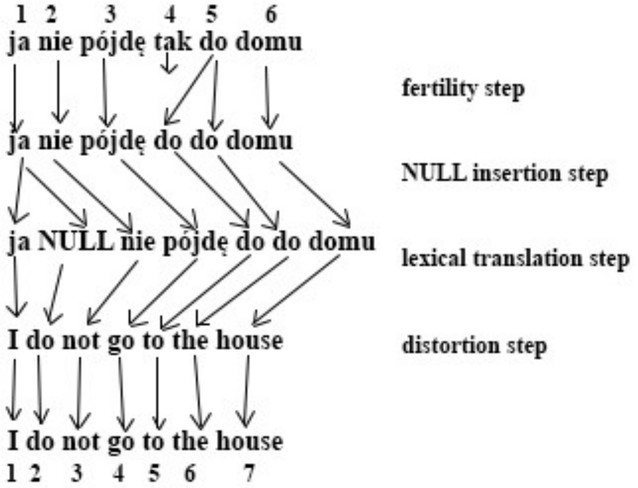
1966: ELIZA released

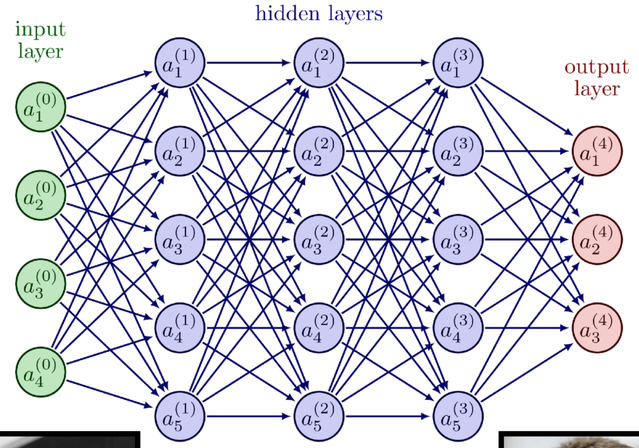
```
 Welcome to
                EEEEEE  LL      IIII   ZZZZZZ   AAAAA
                EE      LL       II       ZZ   AA   AA
                EEEEE   LL       II     ZZZ    AAAAAAA
                EE      LL       II    ZZ      AA   AA
                EEEEEE  LLLLLL  IIII ZZZZZZ    AA   AA

 Eliza is a mock Rogerian psychotherapist.
 The original program was described by Joseph Weizenbaum in 1966.
 This implementation by Norbert Landsteiner 2005.


ELIZA: Is something troubling you ?
YOU:   Men are all alike.
ELIZA: What is the connection, do you suppose ?
YOU:   They're always bugging us about something or other.
ELIZA: Can you think of a specific example ?
YOU:   Well, my boyfriend made me come here.
ELIZA: Is it important to you that your boyfriend made you come here ?
YOU:   He says I'm depressed much of the time.
ELIZA: I am sorry to hear that you are depressed.
YOU:   It's true. I am unhappy.
ELIZA: Can you explain what made you unhappy ?
YOU:
```

1950 1960 1970 1980 1990 2000 2010 2020

History

1986: Backpropagation published; the beginning of deep learning





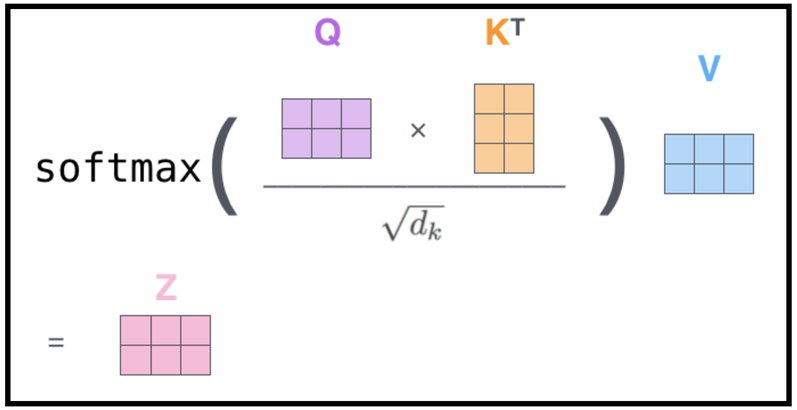
2003: The first neural network-based LM

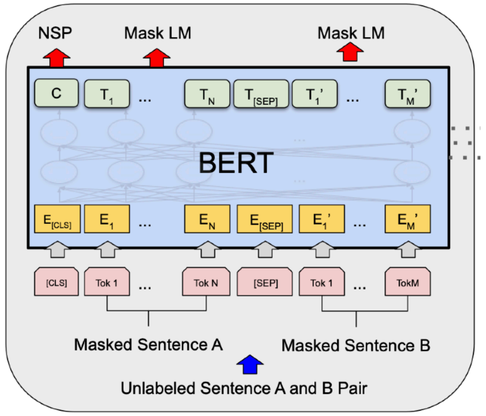
1966: ALPAC report released; an “AI winter” begins

1988-1993: IBM statistical alignment models

1950 1960 1970 1980 1990 2000 2010 2020

History





2014: “Attention” devised; recurrent neural network NLP method outperforms statistical methods for the first time

2017: Transformers

2018: Large-scale pretrained language models; BERT

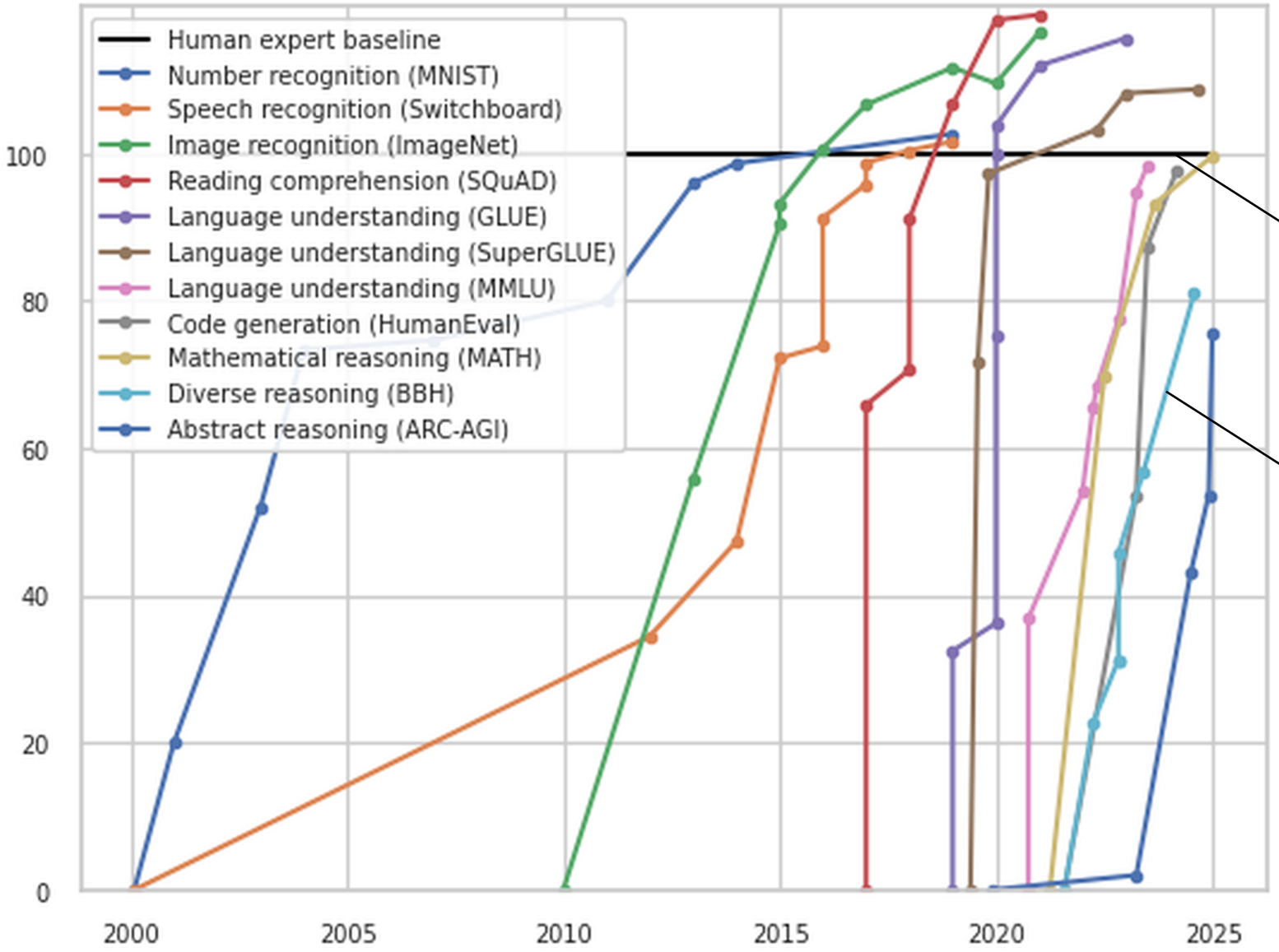
2022: ChatGPT

2025: DeepSeek

2011: Siri released

1950 1960 1970 1980 1990 2000 2010 2020

The Incredible Pace of NLP Progress



Human performance

State-of-the-art NLP system performance

[Kiela, 2024]

The New York Times

Account

Can A.I. Generate New Ideas?

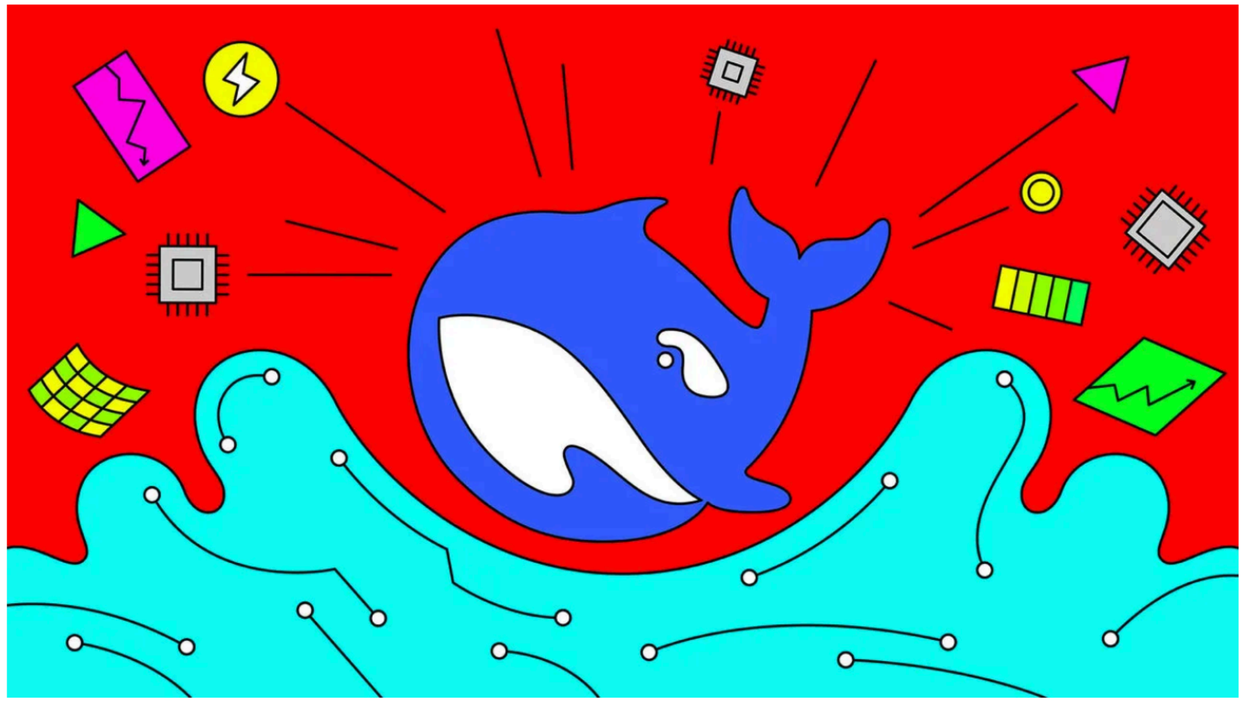
Systems like OpenAI's GPT-5 are accelerating research in math, biology and chemistry. But there is a debate over whether it can do that work on its own.

Business | Bubble trouble

DeepSeek sends a shockwave through markets

A cheap Chinese language model has investors in Silicon Valley asking questions

Share



NLP Engineering vs. NLP Scholarship

- **NLP engineering**: implementation, programming, substantial projects, emphasis on state-of-the-art approaches

- **NLP scholarship**: critical engagement with NLP and its impacts, underlying theory and ideas (e.g., linguistics, cognitive science, scaling laws).

We will engage with both throughout the course. If you are interested in knowing more about cutting-edge NLP and how to get involved in research in the field, consider taking Advanced NLP next semester!

Core Goals

- Hands-on experience with the full range of core components that underlie state-of-the-art NLP methods and systems.

- A guided exploration of an advanced NLP topic of your choice.

- **Main goal:** to make you a responsible, insightful, and resourceful practitioner and researcher of language technologies.

Course Topics

Overview

Part 1

1. Text classification and regression
2. Language modeling
    a. Tokenization
    b. N-grams
    c. Recurrent neural networks, LSTMs
    d. Transformers, attention
    e. In-context learning
    f. Fine-tuning
    g. Post-training

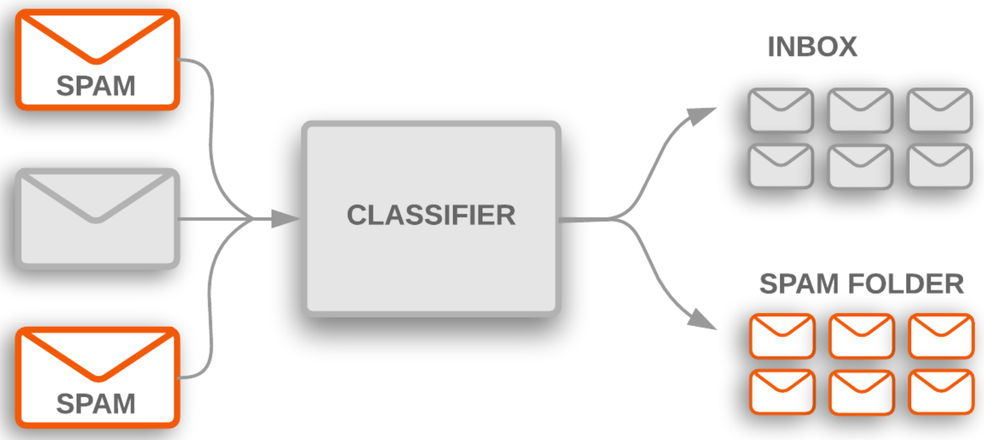
The exam will cover Topics 1 through 3.

Part 2

3. Syntax and morphology
    a. Word tagging
    b. Context-free grammars
    c. Parsing: constituencies and dependencies
4. Semantics
5. Pragmatics and discourse
6. NLP tasks and applications
7. A few hot topics
    a. Multilingual NLP
    b. Interpretability and evaluation

Text Classification

Spam filtering



Sentiment classification

“I love this movie! I’ve seen it many times and it’s still thrilling.”

👍

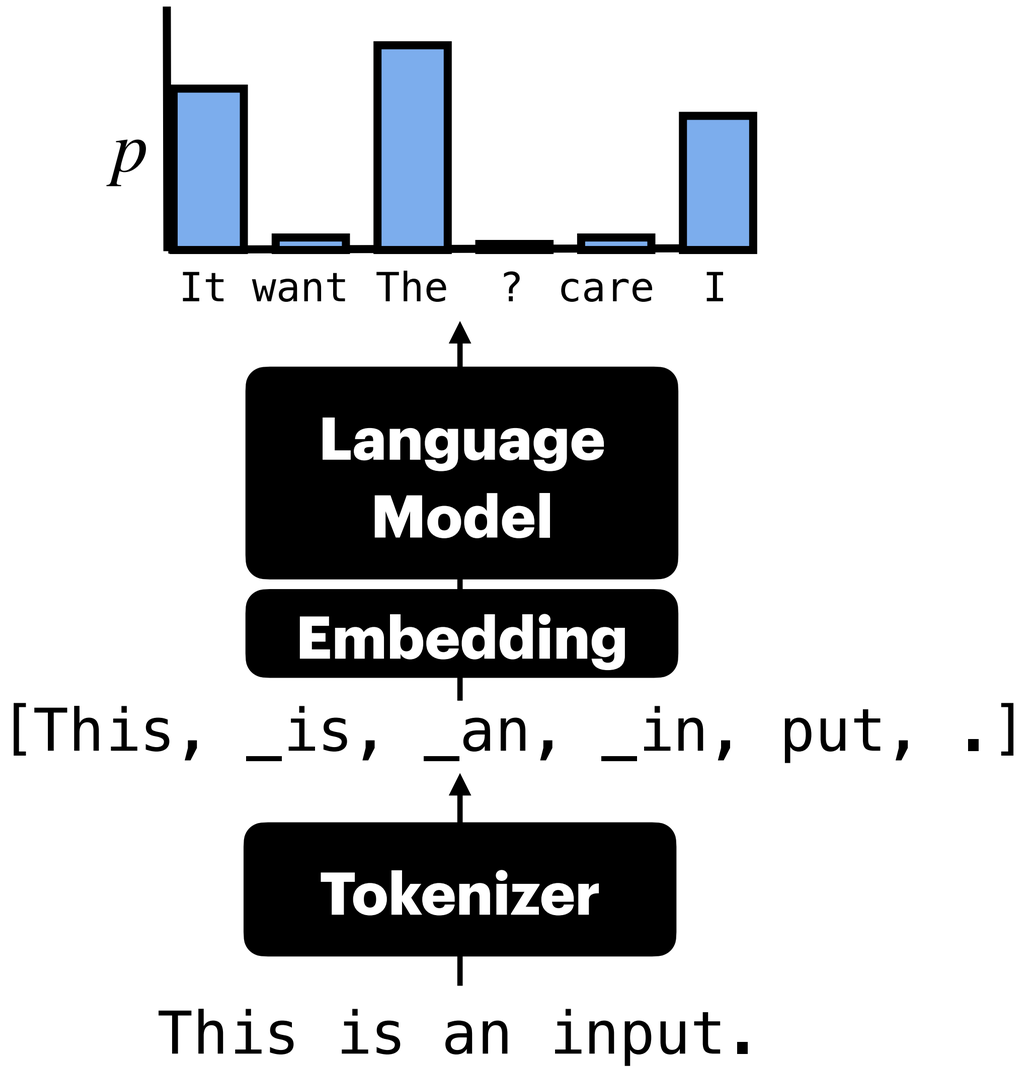
“The only thing more mediocre than this movie’s writing was the acting.”

👎

We’ll start simple with **bag-of-word models**.

We’ll build up to models that consider sequential information and more interesting representations of language.

Theme: Language Modeling



Training a language model:

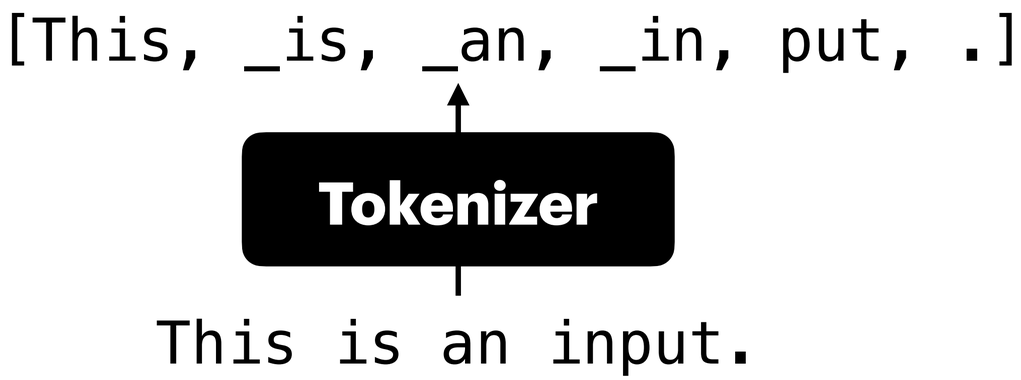
1. Train or define a tokenizer.
2. Collect training data.
3. Pretrain.
4. (*Optional*) Posttrain.
5. (*Optional*) Adapt model for downstream task(s).

Tokenization and N-grams

- What is the right unit to operate over? Words? Characters? Word chunks?
  - What is a “word”? Does it matter?

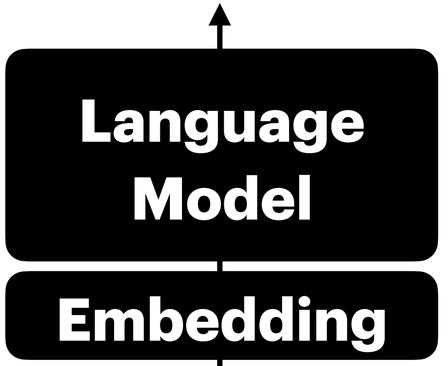
- **Tokenization**: breaking up strings into discrete units that a language model can operate over.

- **N-grams**: groups (of size *N*) of tokens

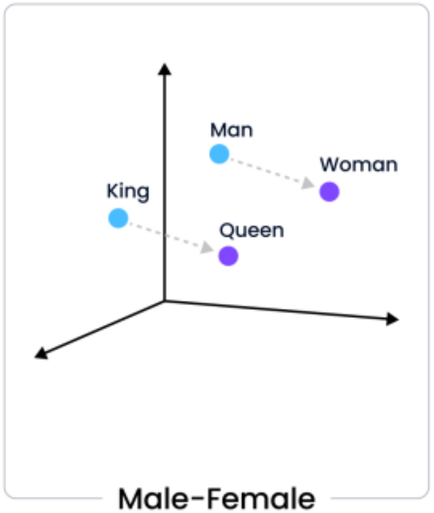


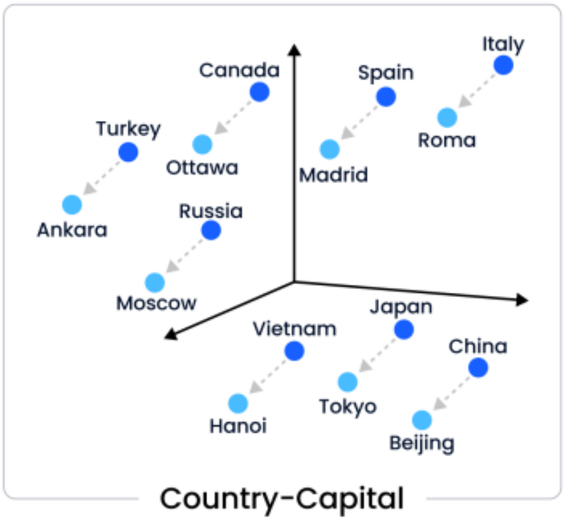
Embeddings

- **Embeddings**: Real-valued *vectors* that represent tokens.
- We'll start simple: one embedding per token index
- Later, we'll discuss *contextual* embeddings



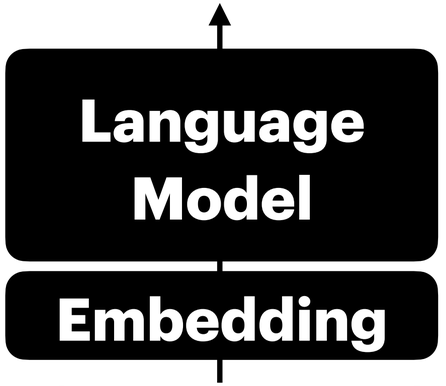
Embeddings lead naturally to **vector semantics**.





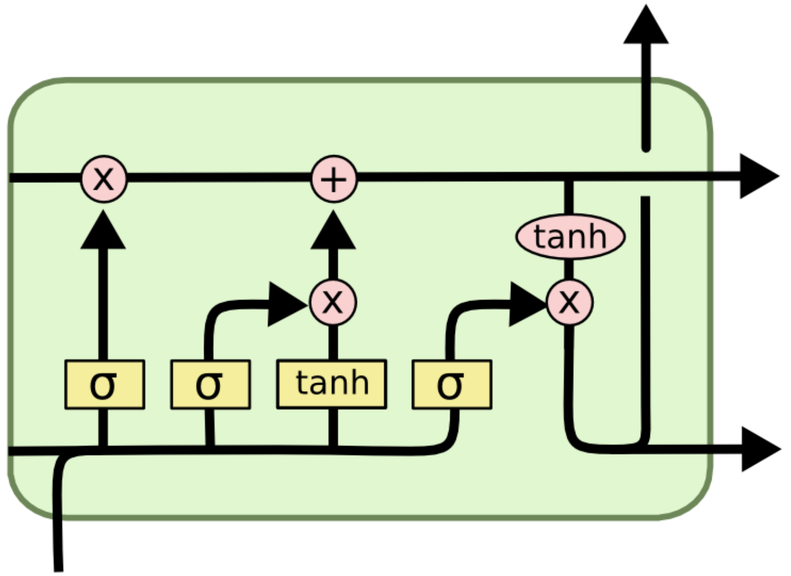
Neural Architectures

- What should the **Language Model** part of this diagram look like?



[This, _is, _an, _in, put, .]

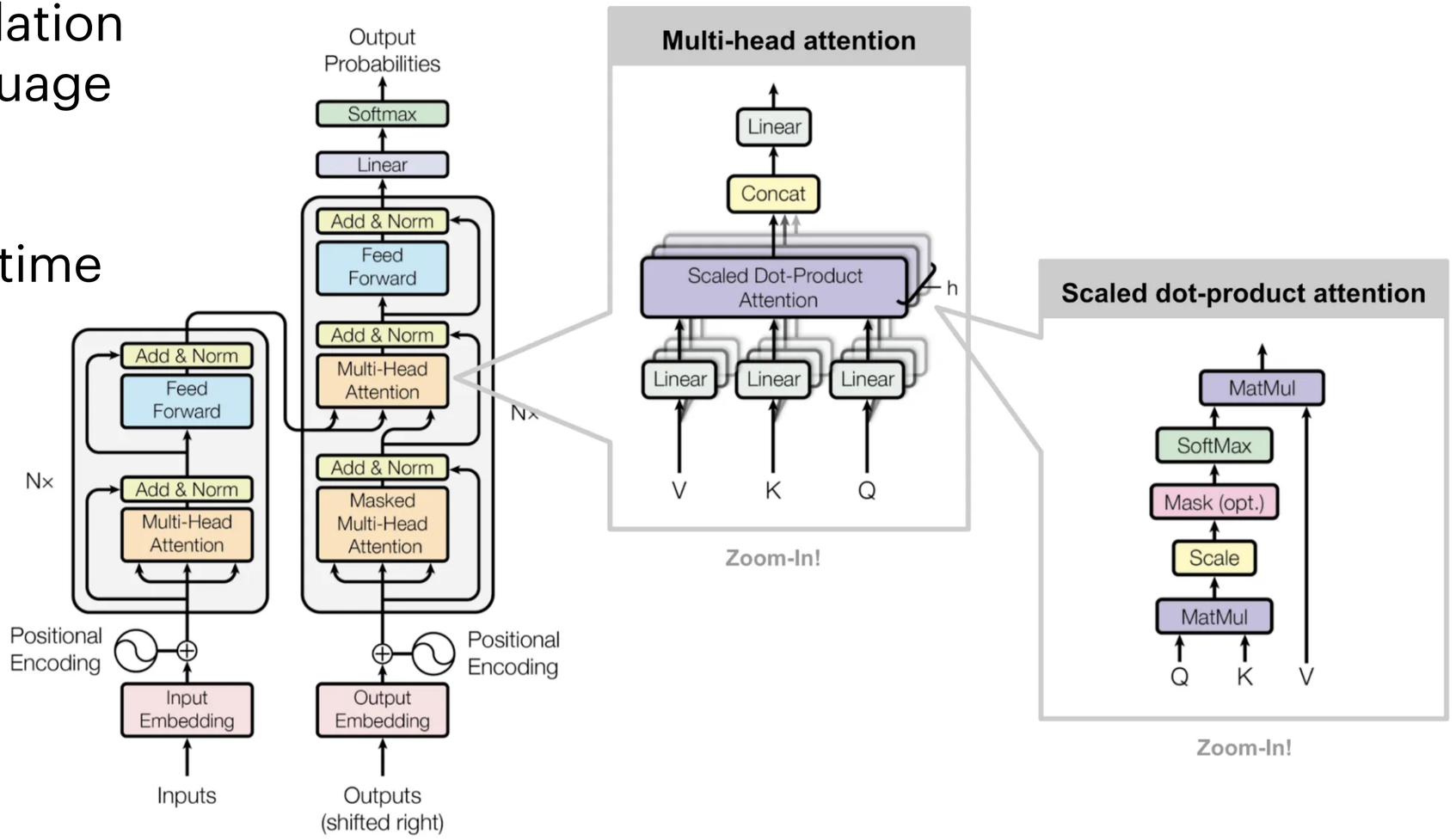
- We have many options. This class will focus mainly on the most successful ones—specifically, **neural network**-based approaches.
  - Recurrent neural networks & LSTMs
  - Transformers



Attention and Transformers

Transformers are the foundation of the most successful language models today.

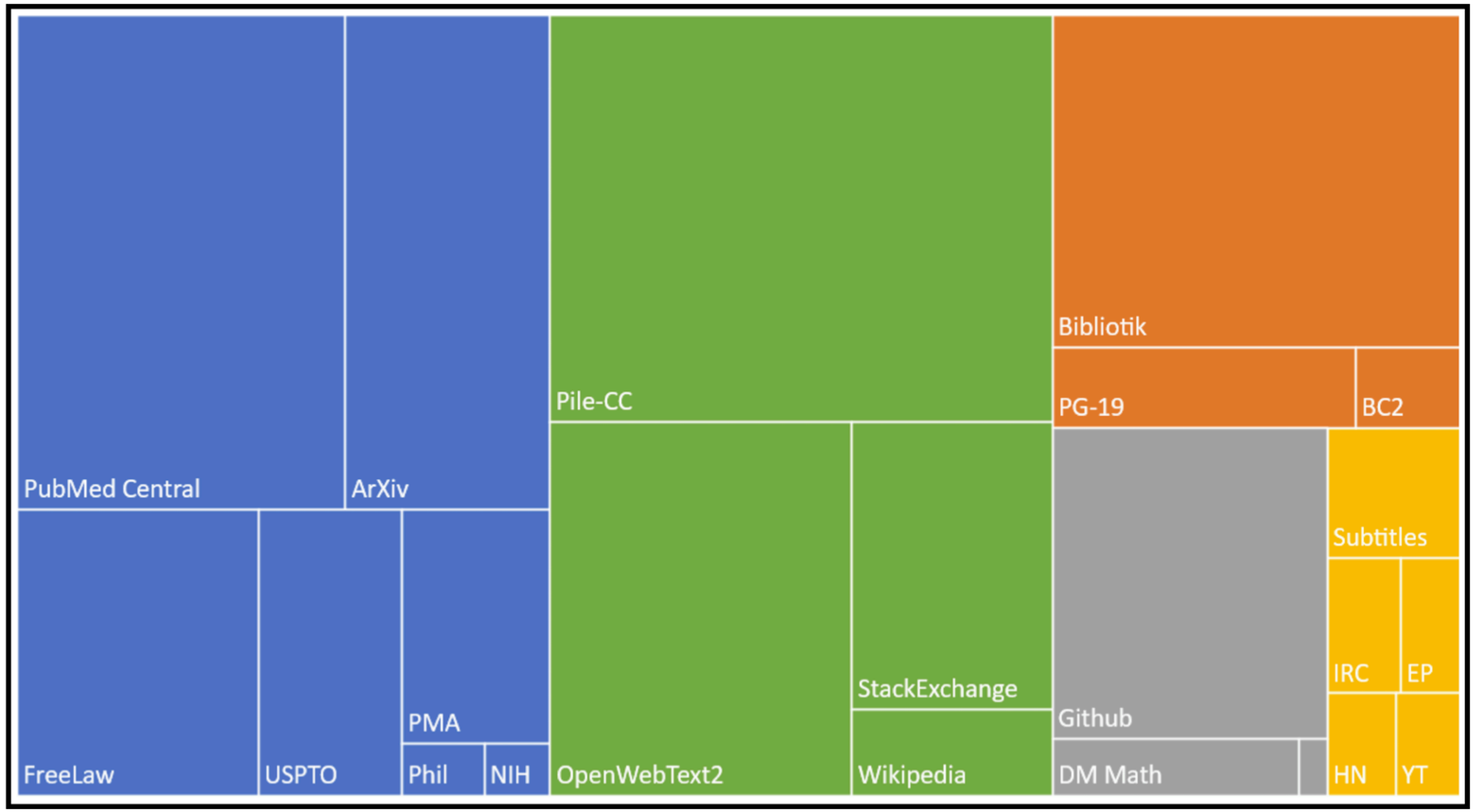
We will spend quite a bit of time learning how they work.



Pre-training

- How do we make our language model good? We need to **pre-train** it on a *lot* of text data.

[Gao et al., 2020]



- How does the model use this data to learn? We provide a **loss function** that the model must minimize:

- How can we train this in a reasonable amount of time? **Batching.**

Post-training

- How do we turn a language model (a probability machine) into a chatbot?

- How can we get the model to follow instructions and show us its reasoning without us having to ask for it?

Before

Prompt: Explain the moon landing to a six year old in a few sentences.
Output: Explain the theory of gravity to a 6 year old.

Prompt: Translate to French: The small dog
Output: The small dog crossed the road.

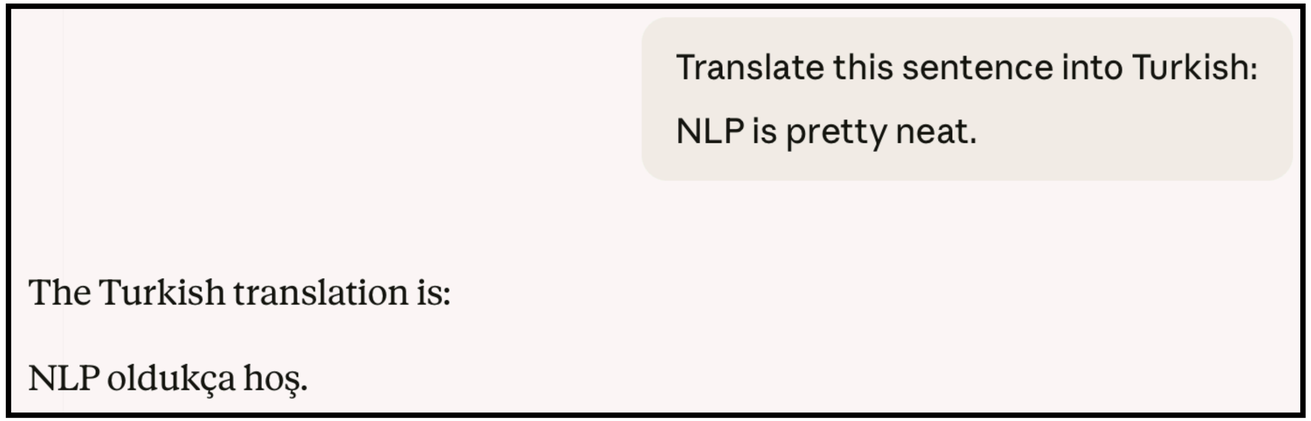
After

Prompt: Explain the moon landing to a six year old.
Output: Some astronauts went to space, and they...

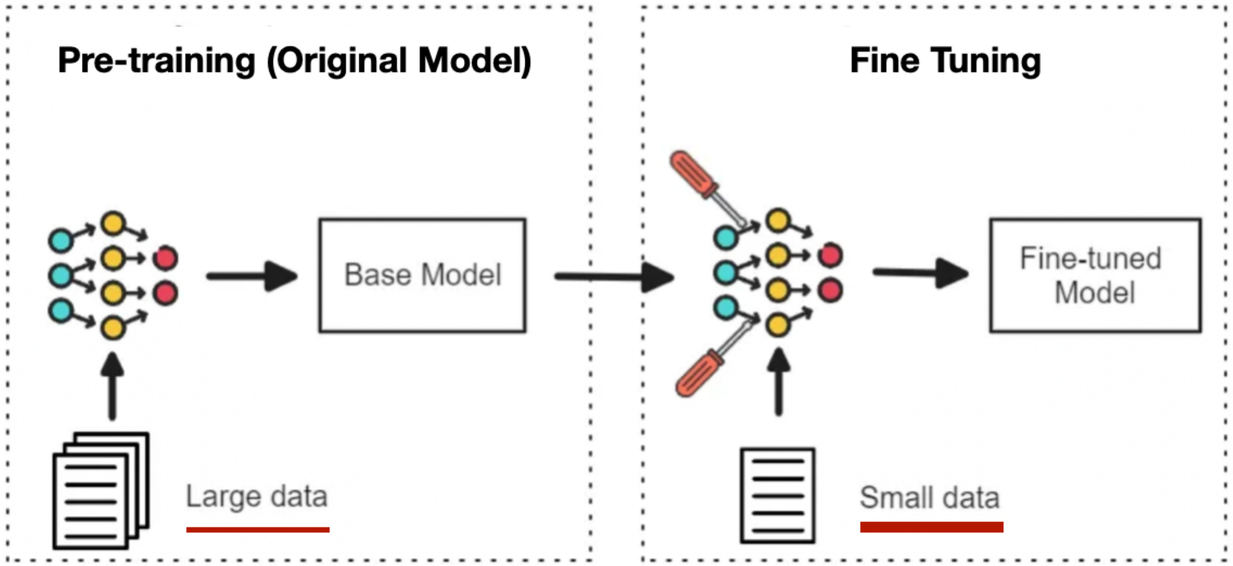
Prompt: Translate to French: The small dog
Output: Le petit chien

Adapting LLMs for Specific Tasks

1. Ask the model directly (prompting), or give it examples (*in-context learning*).



2. Fine-tuning: Update the language model's parameters to do the task.

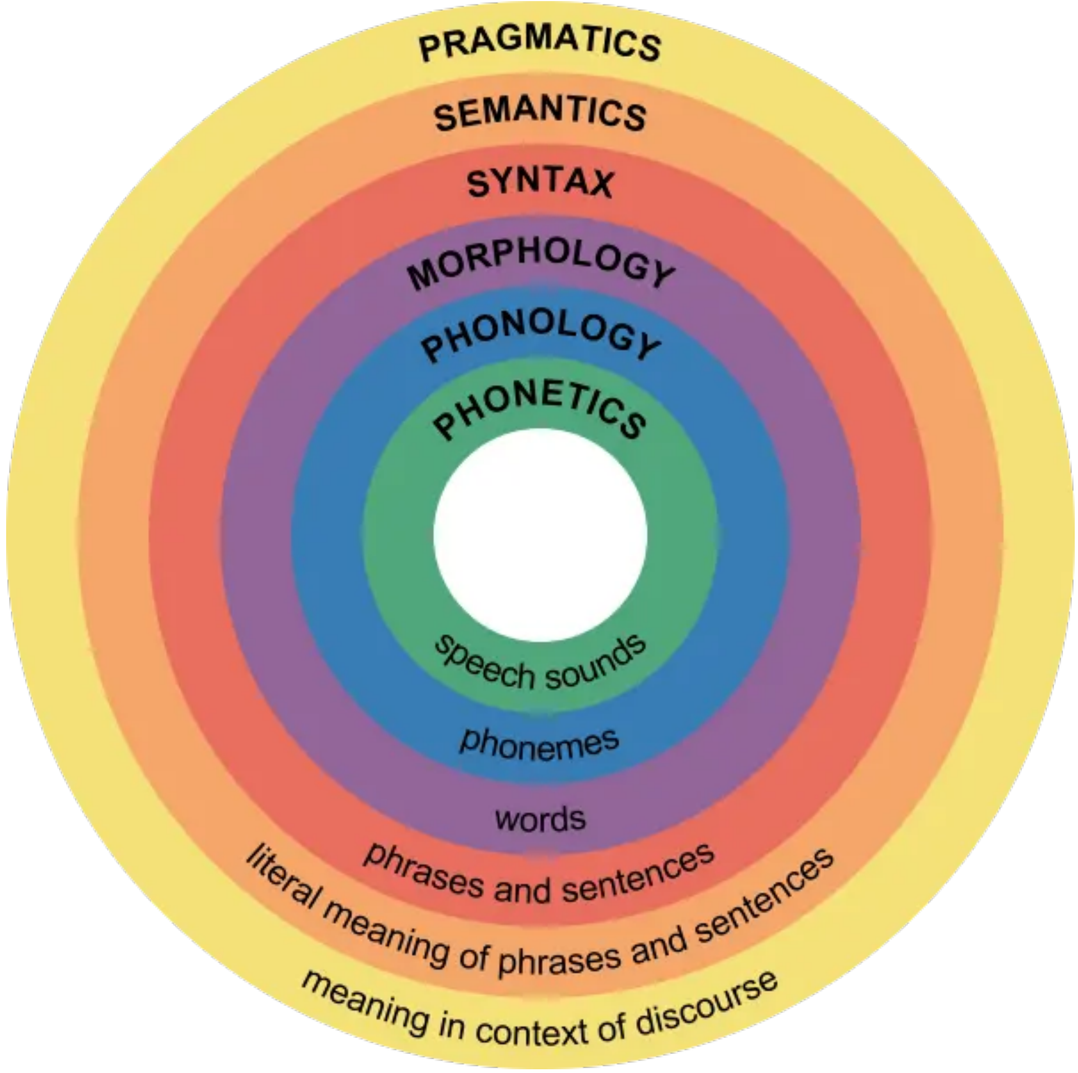


Theme: The Structure of Language

But what is language? How does it work?

- Part 1 of this course will treat language largely as a *statistical phenomenon*.

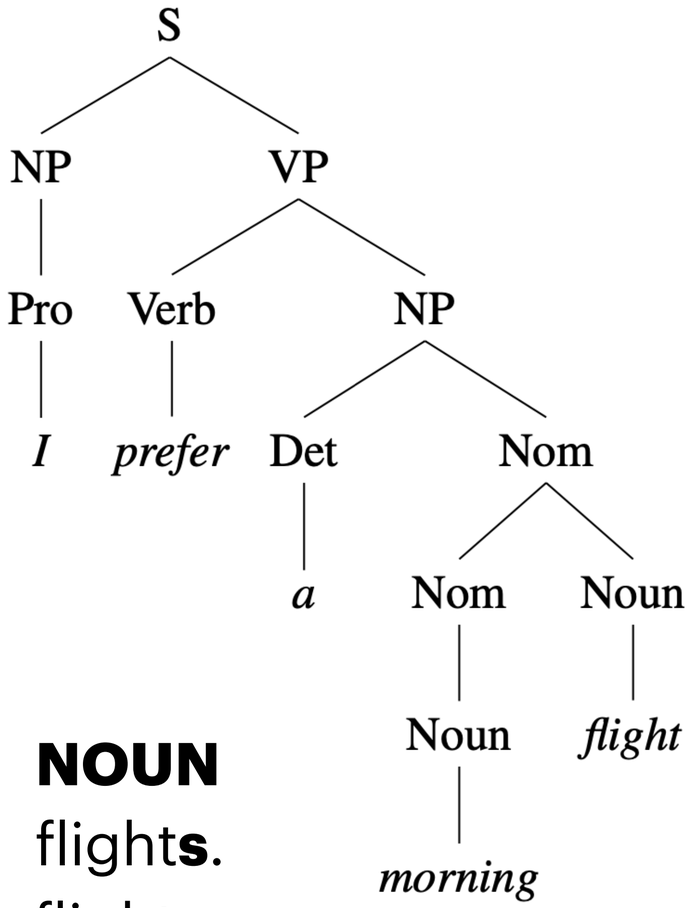
- In Part 2, we'll engage more deeply with questions about how language works, and how it's structured.



Syntax and Morphology

Syntax: The arrangement of words to form larger structures, like phrases and sentences.

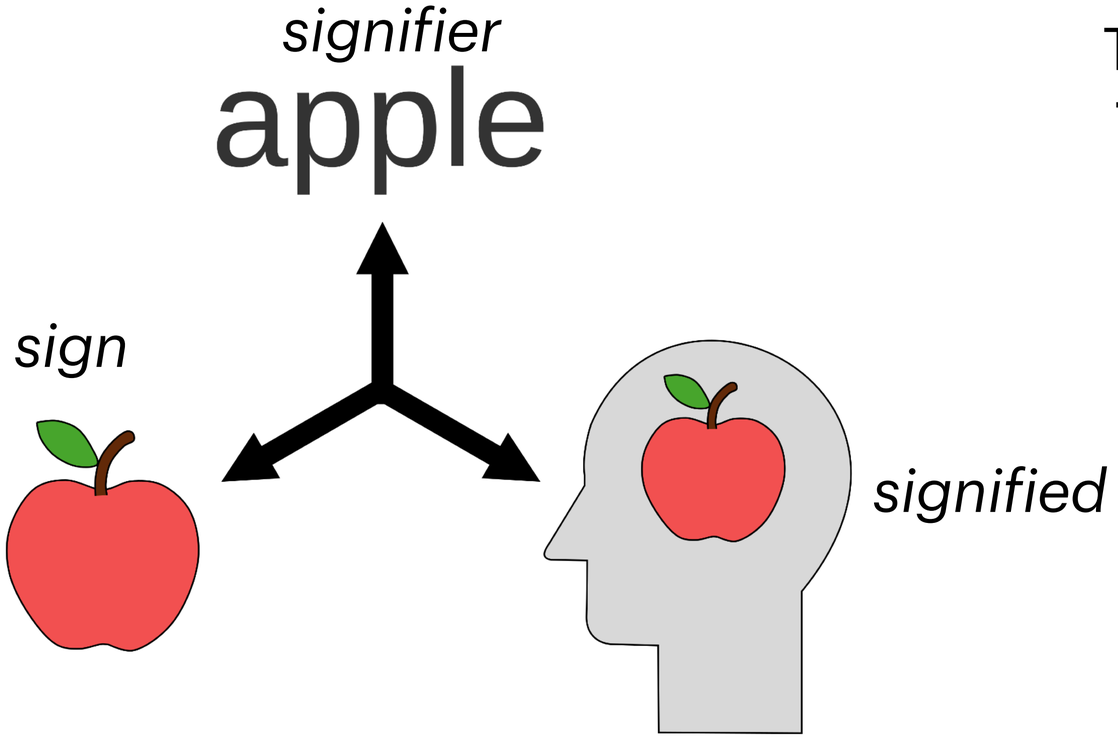
Morphology: The arrangement of smaller units of meaning to form words.



PRO	VERB	NOUN	NOUN
I	prefer	morning	flight**s**.
He	prefer**s**	a morning	flight.

Semantics

Semantics: The study of (literal) meaning.

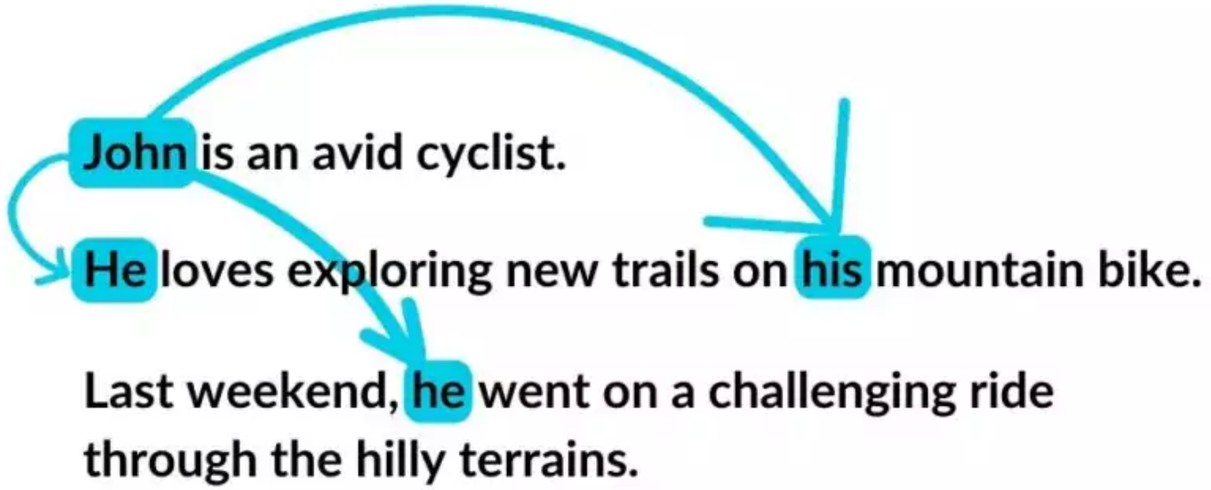


The manager	gave	a task	to	the employee.
Arg1 AGENT		Arg2 THEME		Arg3 GOAL

Discourse and Pragmatics

What makes for a coherent document or conversation?

- *Coreference*
- *Relations*
- *Entity linking and tracking*



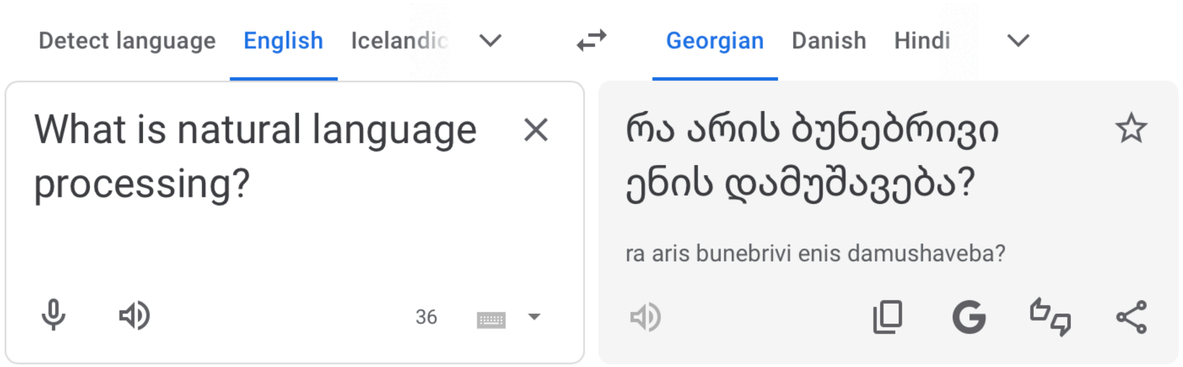
Pragmatics: How does meaning interact with context? What is a sentence's potentially non-literal meaning? How are these meaning inferred?

You're at home. The window is open. A friend says:

"It's a bit cold in here, isn't it?"

"Yeah." *Close the window.*

Theme: NLP Applications and Tasks



Machine translation

The language {ww | w ∈ (0 + 1)*} is:

A. not accepted by any Turing machine
B. accepted by some Turing machine, but by no pushdown automaton
C. Accepted by some pushdown automaton, but not context-free
D. Context-free, but not regular

(Multiple-choice) Question answering

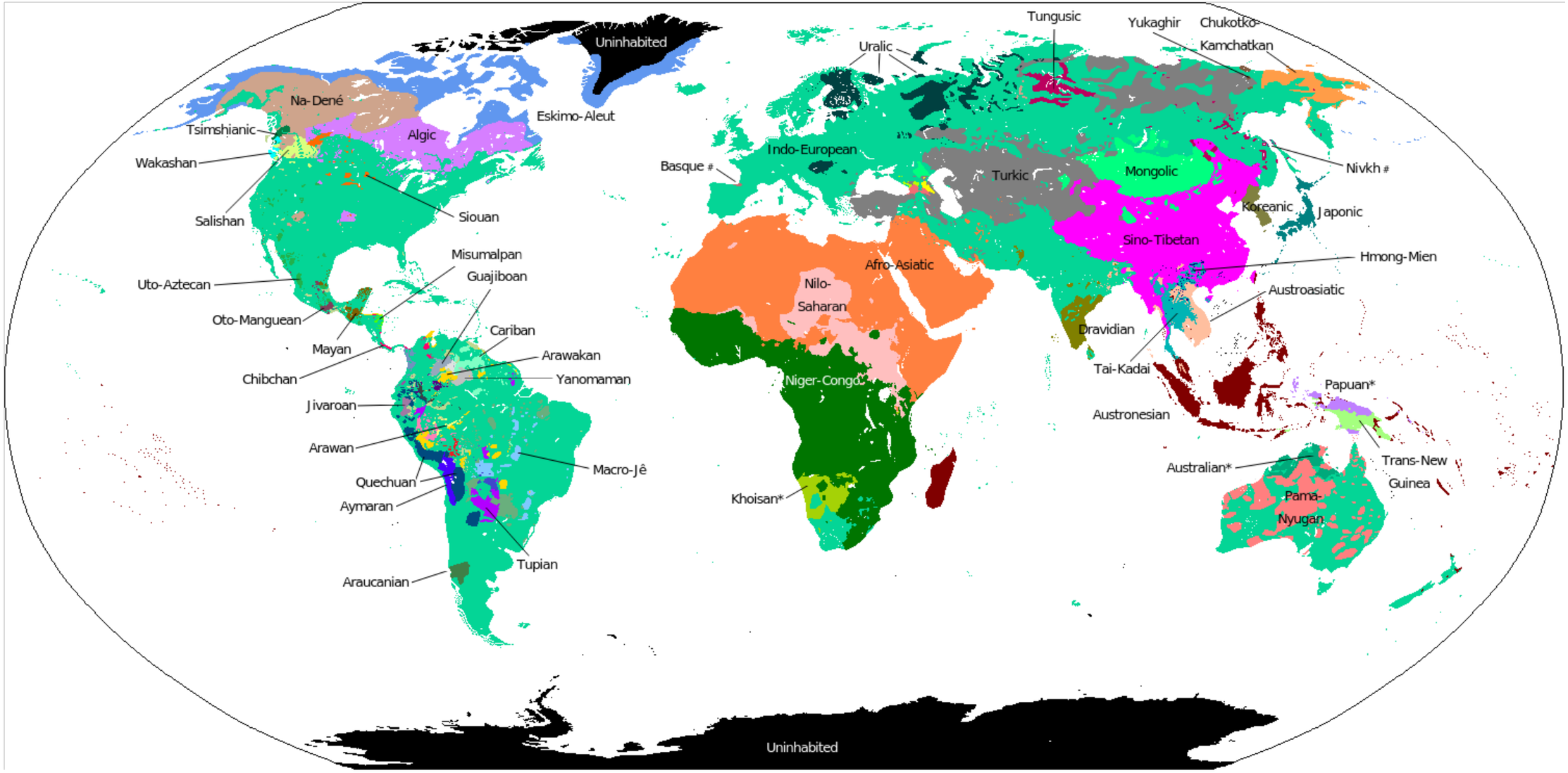
Who was the founder of the first Persian empire?

Answer: ...

(Abstractive) Question answering

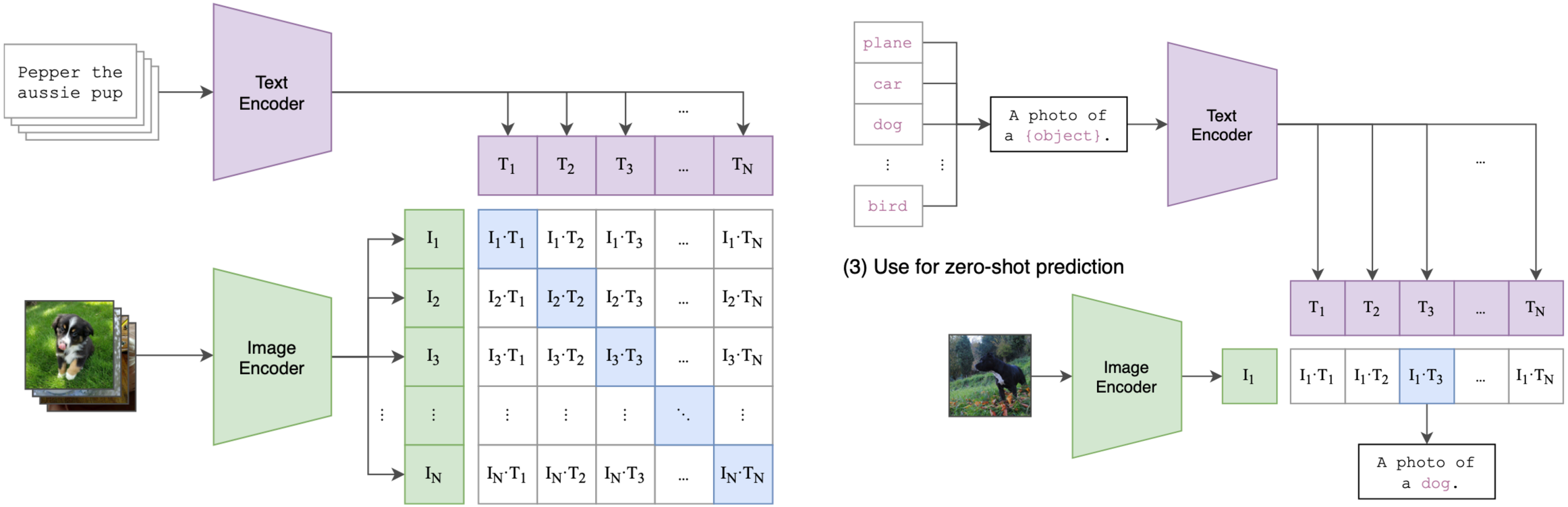
Some Hot Topics

Multilinguality and Translation



Multimodal NLP

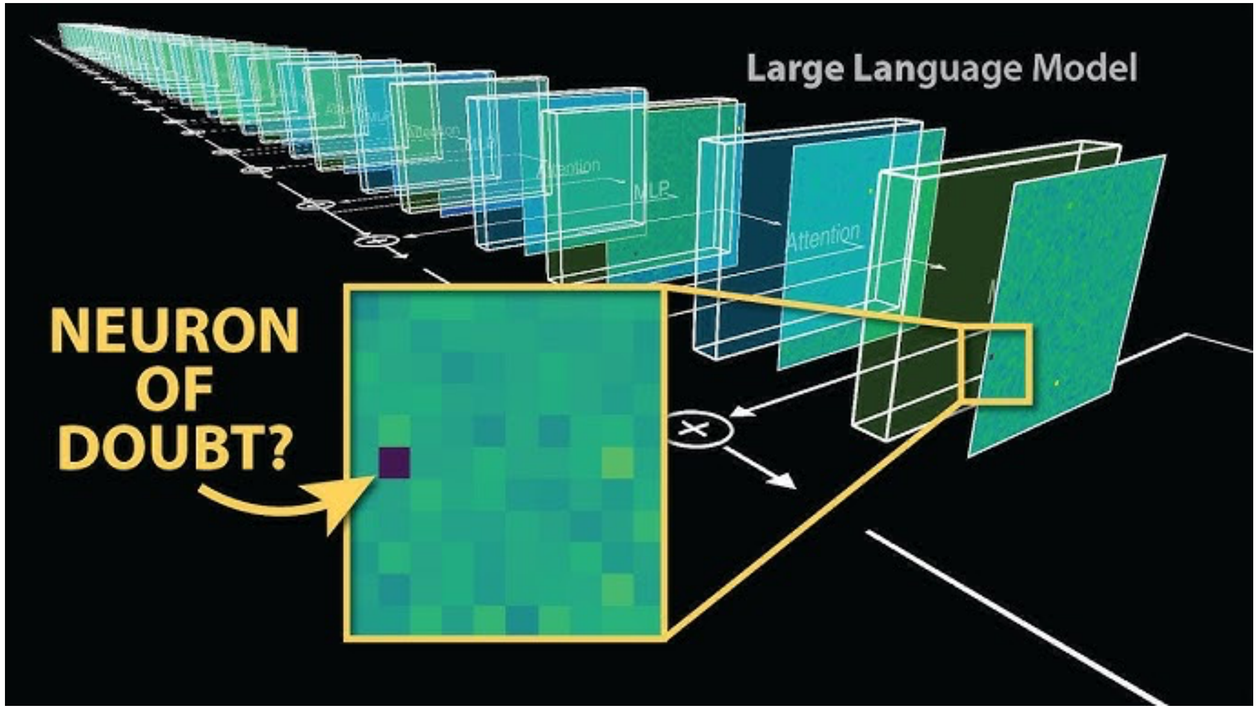
[Radford*, Kim* et al., 2021]



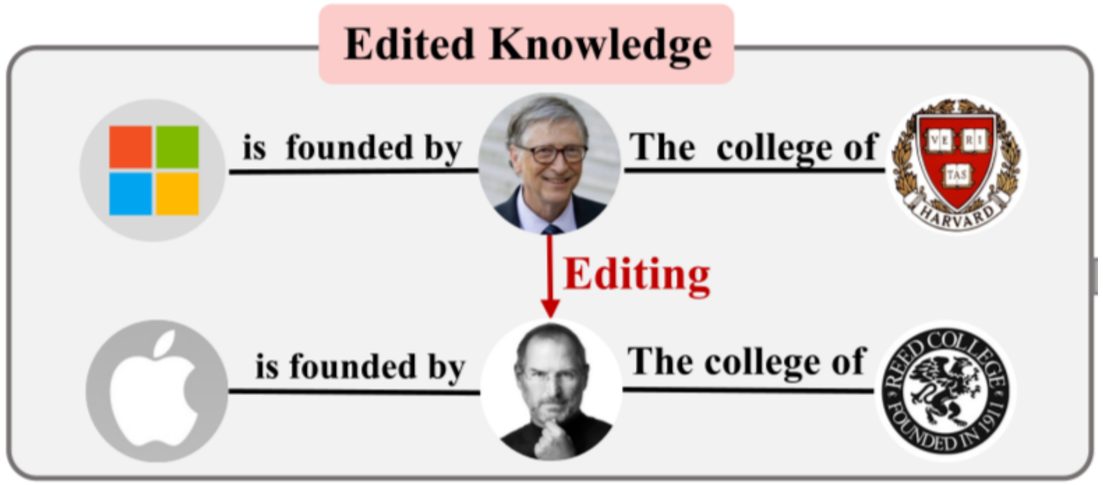
Can we do better at learning language if we also give a model other modalities, like images?

Some Hot Topics

Interpretability

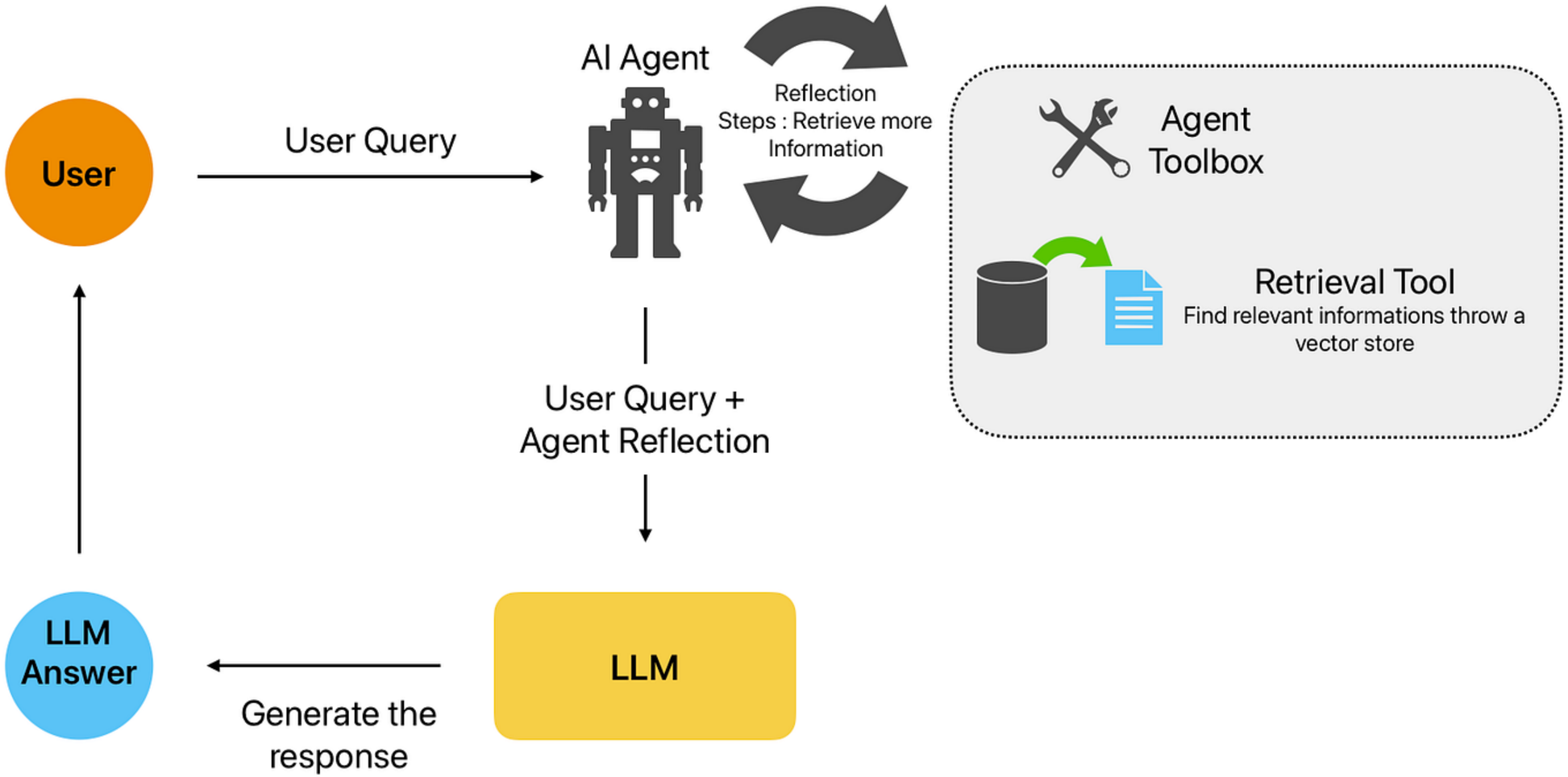


From Welch Labs on YouTube



Some Hot Topics

Retrieval and Tool Use



Course Logistics

Prerequisites

- We expect you to know the following:
  - Probability
  - Calculus
  - Linear algebra
  - How to program in Python

- We *do not* expect you to know much about machine learning or how to program with PyTorch yet—but it will certainly be helpful to have taken an ML course.
  - Please check out the syllabus for links to tutorials for extra guidance on machine learning or PyTorch programming.

Discussions

- **The discussion sections are cancelled!** We will not be using the Monday slots.

- Instead, please come to office hours with any questions related to the homeworks, project, content of the course, among other questions.

Grading

This class has 100 total points.

Homeworks: 15 points

- HW0: 4 points
- HW1: 4 points
- HW2: 4 points
- HW3: 3 points

Midterm: 35 points

- In-person and on paper

Attendance will *not* be graded! Attend as you wish.

Final project: 50 points

- Proposal: 5 points
- Midway report: 15 points
- Final report: 30 points

Homeworks

- **Prereq review** (ungraded, released *today!*): Linear algebra, probability, calculus

- **HW0:** Text classification. You'll learn how to build a bag-of-words text classifier and do some feature engineering.
- **HW1:** N-gram and recurrent language modeling. You'll implement a high-quality statistical language model and a neural network–based language model, and compare them.
- **HW2:** Transformers. You'll implement Transformers from scratch, train them on some text, and adapt them to do machine translation.
- **HW3:** Syntax and parsing. You'll work through common constituency and dependency parsing algorithms, and get familiar with probabilistic context-free grammars.

- These can be done collaboratively, but should be submitted individually.

Final Project

You'll work in groups of 2–4 students on a final project with a topic of your choosing.

- **Final project proposal:** A week after the exam, you'll submit a proposal for what you'd like to work on for your final project.
- **Midway report:** A report in the style of an NLP research paper with some preliminary experiments.
- **Final report:** Like the midway report, but expanded with more related work and experiments.

Final Project Logistics

- Final projects will be conducted in groups of 2–4 students.
    - We discourage solo projects. Most machine learning and NLP work happens in teams, so this is good practice for how NLP projects work in the real world.
    - Solo projects do *not* get extra leeway! All projects will be graded using the same standards as for groups of 2–4.

- I recommend signing up for Google Colab if you'll work with local models that require a lot of compute. If you want to work with LLMs, you could also sign up for an API with one of the big models.

AI Policy

- For the homeworks, you can use AI tools however you wish. These are mainly to help your learning and to help you prepare for the midterm.
  - We encourage using AI to help your conceptual understanding.
  - We *discourage* but *allow* using AI to answer conceptual questions on HWs.

- For the final project, you can use AI—but as an *editing tool*, not to generate the whole reports.

- **You are 100% responsible for what you submit, regardless of whether you use AI.**

- No electronic resources will be allowed during the exam. This includes AI tools.

Late Work Policies

- You have **6 free late days** for which no penalties will be applied to your scores.
  No excuses are necessary to use them.
    - Turning in an assignment 2 minutes late uses the same number of late days as turning it in 23 hours late.
- Extensions can be negotiated in cases of medical emergency or other sudden pressing circumstances.
    - Students should contact the course staff ASAP and negotiate *before the assignment's original due date*.
- Each assignment turned in after using all late days *with no negotiated extension* will incur a **10% drop** in the score. After 5 days past the deadline, the assignment can no longer be turned in.

- The final report cannot be turned in late.

Office Hours

- We'll mainly have office hours in-person at the CDS building; sometimes, they will be on Zoom. See the syllabus and follow the Piazza for updates.
    - The first office hours will happen *right after class today!*

- If you have questions that can be answered as a message (and the vast majority of them can be, in my experience), please use Piazza! You will get a faster response.

Other Logistics

- Please get in touch with us ASAP if you know you need accomodations for religious or disability reasons!
- We require at least a 14-day notice to reschedule things like exams.
- If this applies to the first homework, let us know **today.** Talk to me after class!