CS 577: NATURAL LANGUAGE PROCESSING

Abulhair Saparov

Fall 2025

WHAT IS NLP?

- Build algorithms that work with natural language.
- How do these algorithms work?
- Why do they sometimes work well?
- Why do they sometimes not work?
- We will take a research perspective.

NLP TASKS

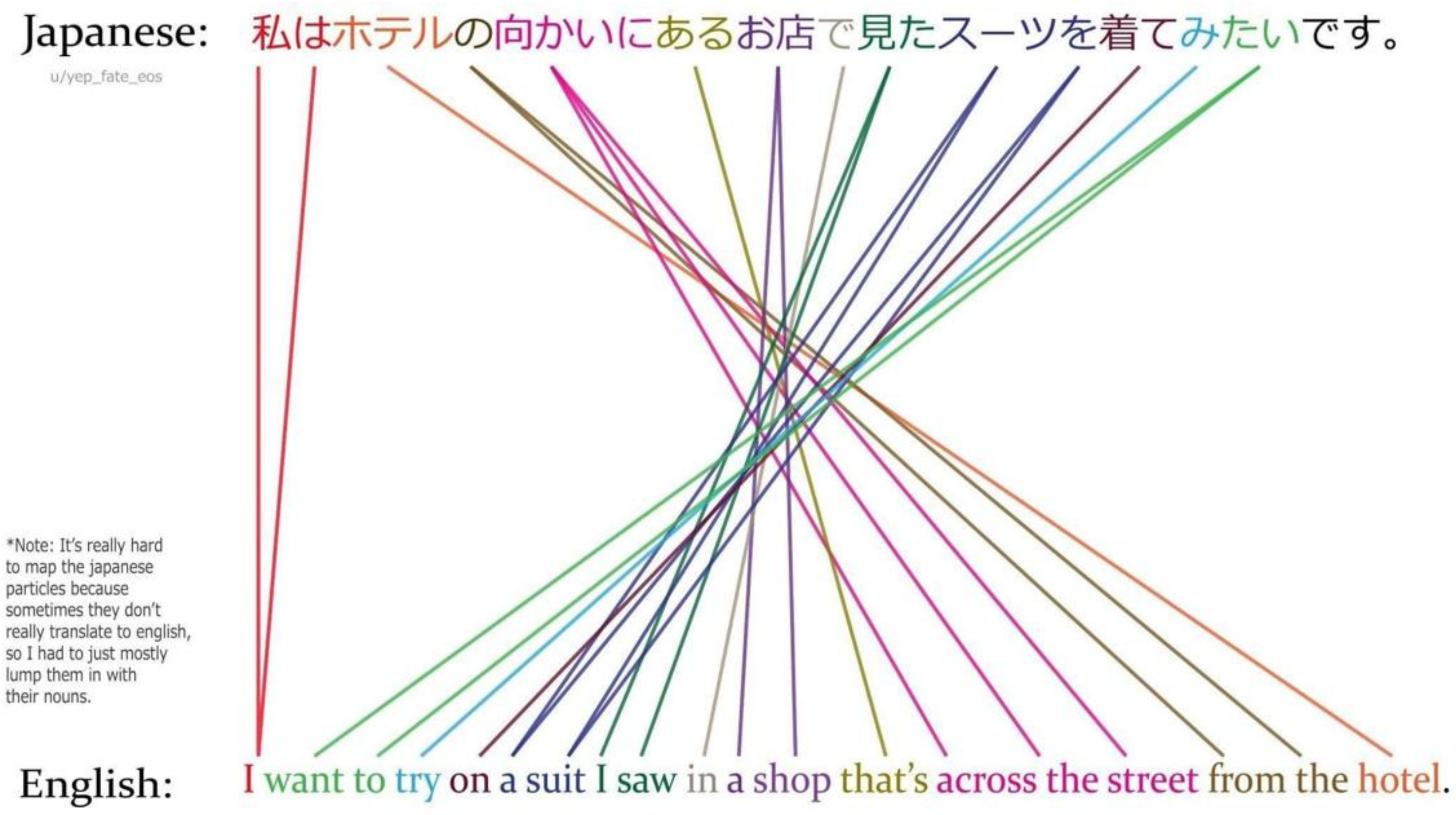
- Machine translation

Input: The quick brown fox jumped over the lazy dog.

Output: 素早い茶色のキツネは怠け者の犬を飛び越えました。

NLP TASKS

- Machine translation



[source: Reddit u/yep_fate_eos]

NLP TASKS

- Machine translation
- Named entity recognition

Input: Honda

Output: company

NLP TASKS

- Machine translation
- Named entity recognition

Input: Noam Chomsky

Output: person

NLP TASKS

- Machine translation
- Named entity recognition

Input: Apple

Output: company

NLP TASKS

- Machine translation
- Named entity recognition

Input: apple

Output: not named entity

NLP TASKS

- Machine translation
- Named entity recognition
- Coreference resolution

Input: The souvenir didn't fit into the suitcase because it was too big.

Output: "it" = "souvenir"

NLP TASKS

- Machine translation
- Named entity recognition
- Coreference resolution

Input: The souvenir didn't fit into the suitcase because it was too small.

Output: "it" = "suitcase"

NLP TASKS

- Machine translation
- Named entity recognition
- Coreference resolution
- Question answering

Input: You are in the middle of a circular lake. You can swim at 1 m/s. A dog is at the edge of the lake. The dog can run on land at x m/s, but cannot swim, and you can run faster. What is the highest value for x such that you can still escape?

Output: 4.6033

Reasoning is needed to solve this task.

NLP TASKS

- Machine translation
- Named entity recognition
- Coreference resolution
- Question answering
- Image description

Input:

Output: This image features a framed painting of Purdue University. The artwork is displayed on a white marble wall…

Multi-modal NLP includes the study of tasks involving other modalities, such as vision, sound, speech, motion, etc.

NLP AS MACHINE LEARNING

- It is infeasible to write a function to solve these tasks directly.
- So we rely on machine learning to learn this function from data.
  - We use a dataset containing many input-output examples.
  - We train a machine learning model predict the output from the input.
- The specific choice of model and training regimen is the “method.”

LANGUAGE IS AMBIGUOUS

"Teachers strike idle kids."

- Interpretation 1: Teachers physically strike kids who are idle.
- Interpretation 2: The teacher's strike is causing the kids to be idle.

LANGUAGE IS AMBIGUOUS

"Time flies like an arrow."

- Interpretation 1: Time moves forward similar to how an arrow flies.
- Interpretation 2: This is a command, telling you to measure the speed of the flies similar to how you would measure the speed of an arrow.
- Interpretation 3: Also a command, telling you to measure the speed of the flies, but in a manner similar to how arrows would measure the speed of the flies.
- Interpretation 4: There are things called "time flies" and they like an arrow.
- We will cover probabilistic methods that handle ambiguity.

[source: https://en.wikipedia.org/wiki/Time_flies_like_an_arrow;_fruit_flies_like_a_banana]

NLP ≠ LANGUAGE MODELS

- Language modeling is a task in NLP.

Input: The quick brown fox jumped over the lazy

Output: dog

- These days, “language models” almost exclusively refers to large-scale transformer models that are trained on the language model task.

NLP ≠ LANGUAGE MODELS

- It seems as though large language models (LLMs) have “taken over” NLP.
  - And we will discuss how they work.
- Language modeling task has a nice property:
  - Many (all?) other NLP tasks can be written as a language modeling task.
  - So if you train a good language model, you train it to perform many NLP tasks simultaneously.
  - Valid question: Can LLMs “solve” all NLP tasks?
- But this property is not unique to language modeling.
  - E.g., many NLP tasks can also be phrased as question-answering.

NLP ≠ LANGUAGE MODELS

CIO

Home • Artificial Intelligence •
With o3 having reached AGI, OpenAI turns its sights toward superintelligence

by Taryn Plumb

With o3 having reached AGI, OpenAI turns its sights toward superintelligence

NLP ≠ LANGUAGE MODELS


ARC-AGI is a genuine AGI test but o3 cheated :(

by **Knight Lee**

21st Dec 2024 2 min read 6

- LLMs are making evaluation very difficult.
- We will discuss how NLP models are evaluated.
  - And how their evaluation differs from before LLMs.

NLP ≠ DEEP LEARNING

- Modern approaches in NLP rely heavily on deep learning methods.
- But NLP is “method-agnostic”:
  - Many different kinds of methods can be used to solve various problems in NLP.
- NLP is not only the study of the methods for solving natural language tasks.
- NLP includes the modeling of natural language itself:
  - What is language, formally?
  - How can we describe it?
- Studying the nature of language itself will help us build better models and implement better methods to solve NLP tasks.

LANGUAGE HAS STRUCTURE

- Language is more than just a sequence of words.
- There is recursive structure:
  - “Fae sees Alex.”
  - “Fae sees the person sitting under the tree.”
  - “Fae sees the person sitting under the tree that had been planted 10 years ago.”
  - etc…
- There is structure *within* words too:
  - “Recalculating” -> “re”- “calculate”- “ing”
  - “Sleeplessness” -> “sleep”- “less”- “ness”
  - Other languages have much more complex morphologies.

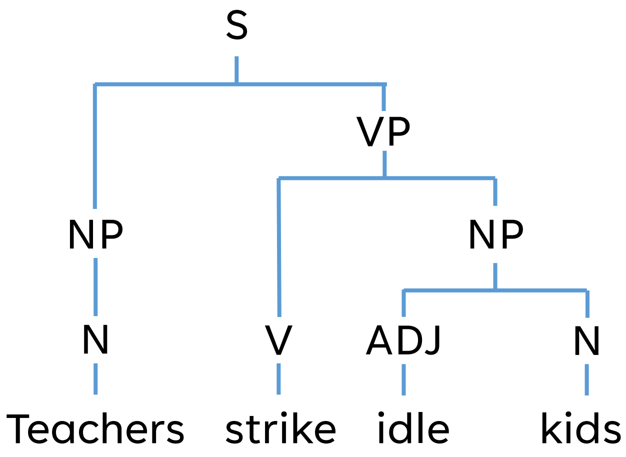
LANGUAGE HAS STRUCTURE

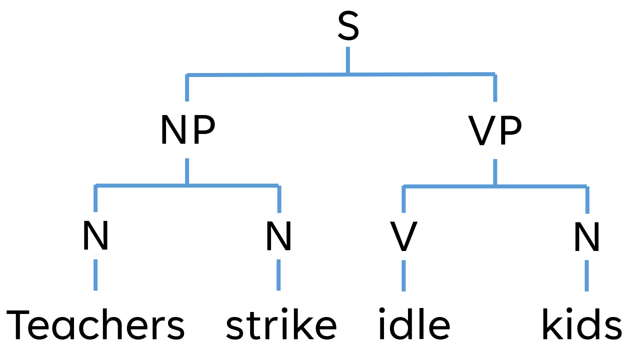
- Most NLP tasks require understanding the structure of language.
- How can we evaluate how well an NLP method performs on a task (or build better methods) if we don't understand the structure underlying the task?
  - E.g., I can train a model to land a rocket on the moon by having it try many attempts.
  - The model will try different rocket shapes, fuels, maneuvers, etc.
  - But if I understand Newtonian gravity, I can build a better model/rocket.
- Understanding linguistic theory is similarly important.
- We will cover foundational concepts of linguistic theory, such as morphology, syntax, semantics, pragmatics, etc.

LANGUAGE HAS STRUCTURE

"Teachers strike idle kids."

- Interpretation 1: Teachers physically strike kids who are idle.
- Interpretation 2: The teacher's strike is causing the kids to be idle.





COURSE OUTLINE

- First part: NLP methods and tasks
- Text classification
  - Methods: Logistic regression, SVM, neural networks, training, evaluation
  - Tasks: Author identification, information retrieval, document classification
- Language modeling
  - Methods: n-gram models, neural language models, RNNs, LSTMs, transformers

COURSE OUTLINE

- Second part: NLP foundations
- Morphology and lexical semantics
  - Lexical relations, tokenization, byte-pair encoding
- Syntax
  - Context-free grammars (CFGs), dependency grammar
  - CFG parsing algorithms, dependency parsing
- Semantics
  - Compositional semantics, categorial grammar, formal semantics
  - Reasoning, code generation, constrained decoding
- Discourse and pragmatics
  - Conversational NLP

MORPHOLOGY

- Morphology is the study of how words are constructed from smaller components.
  - E.g., verb conjugation: “I walk,” “she walks,” “We walked yesterday,” …
  - “I sit,” “she sits,” “We sat,” …
  - “I am,” “she is,” “We were,” …
- Simply adding/deleting endings is not sufficient:
  - “gorge” vs “gorgeous”
  - “good” vs “goods”
  - “arm” vs “army”

SYNTAX

- Syntax describes the structural relationship between words in a sentence.

"Sally caught the butterfly with a net."

vs

"Sally caught the butterfly with a spot."

- However, syntax is not enough to capture the meaning of the sentence.

SEMANTICS

- Semantics describes the meaning of words, phrases, and sentences.
- Compositional semantics describes how the meaning of smaller phrases combines to form the meaning of larger phrases:
  - “Sally caught the butterfly” + “with a net”
- Meaning can be represented in a formal language, such as logic, programming languages, or math.
  - “Mary gave 10 apples to Bob.”

    can be semantically-parsed into:

```
bob[‘apple’] += 10
mary[‘apple’] -= 10
```

COURSE LOGISTICS

EXPECTATIONS

- This is an advanced research class.
  - We expect students to play an active role.
  - Use this class to advance *your* own research interests!
- Evaluation:
  - Two assignments (30%)
    - First assignment will check whether you have sufficient technical background
  - One paper critique (10%)
  - Final project (30%)
  - Final exam (30%)

GUIDELINES

- Working in groups
  - We encourage you to work in groups on the assignments and final project.
  - Groups should have 3-4 people.
  - You are free to collaborate.
  - But to state the obvious: No cheating or plagiarism
  - You can discuss homeworks with other but must write up your own solution.

- Late policy: 5 late days total
  - We strongly recommend you start assignments early.

- Attend office hours to seek guidance, and to discuss papers and projects.

USE OF GENERATIVE AI

- If you find generative AI useful (e.g., ChatGPT), you are encouraged to use it.
- However, do not simply copy the output of AI into your assignments.
- You should write your own solutions.
- When coding, you may use AI to generate snippets of code (e.g., boilerplate).
  - But be wary of over-relying on/putting too much trust in the AI.
  - We will design assignments that are not as easily solved by current AI models.
- AI will not help you on the final exam.

ONLINE DISCUSSION

- We will be using Ed Discussion as the online platform for discussion.
- Join the discussion forum using the following link:

https://edstem.org/us/join/3SW3SB

- If you have any questions, please make a post there!
- I will announce this link on Brightspace.
- If you are not registered on Brightspace, send me an email and I will add you.

PAPER CRITIQUE

- To help prepare you for the project, you will write a paper critique.
  - You will select a paper from a list.
  - **Hint:** Pick the one that is most helpful for your project/research!
- Write a short review of the paper:
  - What is it about? What research questions do the authors attempt to answer?
  - What are the key claims? Why should we care? What's new in the paper?
  - How was the problem modeled? How were the methods evaluated?
- Key idea: You will have to answer the same questions for your project!
  - Good practice!

FINAL PROJECT

- Find a topic you care about!
  - Can be related to your own research, or other projects, etc.
  - E.g., applications of NLP to other domains.
- Key points:
  - Identify a language-related problem and define it precisely.
  - Interesting approach in tackling the problem
    - We will cover several different kinds of methods
    - You will have to choose the methods and justify your choice
  - What *not* to do: avoid generic problems and generic solutions
    - LLM with chain-of-thought is not novel or interesting

FINAL PROJECT

- Proposal: due end of October or early November
  - Define the problem and research question(s)
  - Related work
  - Basic intuitions and preliminary model
  - Dataset and experimental settings
  - No more than 5 pages!
- Final report: due in December
  - Short report describing your findings
  - Presentations? (depending on class size)

FINAL PROJECT IDEAS

- The TAs have provided a handful of suggested project ideas.
- Each TA is offering to provide guidance to project groups who choose to work on any suggested project (or a related project).
- Guidance can involve weekly or bi-weekly meetings.
- I suggest you reach out to the TAs and ask questions about project ideas and whether they would be willing to provide guidance.

FINAL PROJECT IDEAS: YUNXIN'S SUGGESTIONS

- Replicate "long chain-of-thought" reasoning models
  - These models solve problems by producing a long "chain-of-thought".
  - As their chain-of-thought grows longer, their accuracy on reasoning tasks increases.
- Dynamic evaluation of agents
  - Agents interact with each other and their environment, so a "static" evaluation is inappropriate.
  - Can we evaluate them in an interactive fashion?
- Test-time compute
  - Can we teach a model to more accurately solve specific problems if given more time?
  - Maybe we can borrow ideas from classical AI (e.g., search).
- Assess the environmental impact of large models
  - Can we automatically quantify the energy usage of large models during training and inference?

FINAL PROJECT IDEAS: NATHANIEL'S SUGGESTIONS

- Evaluate a model in a way that is robust against data contamination
  - Find a task that models have been found to perform well.
  - Perform trivial perturbations on the task to measure its effect on model performance.
- Use a larger/more complex model to teach a smaller model
  - Train a small model on the outputs of a larger model
  - E.g, can the small model learn how to reason from the chains-of-thought of the larger model?
- Multilingual evaluation
  - Does a model's ability depend heavily on the language of the input?
- Multi-agent vs single-agent approaches
  - For a particular task, examine whether multiple smaller models collaborating can perform as well as (or better than) one large model

QUESTIONS?