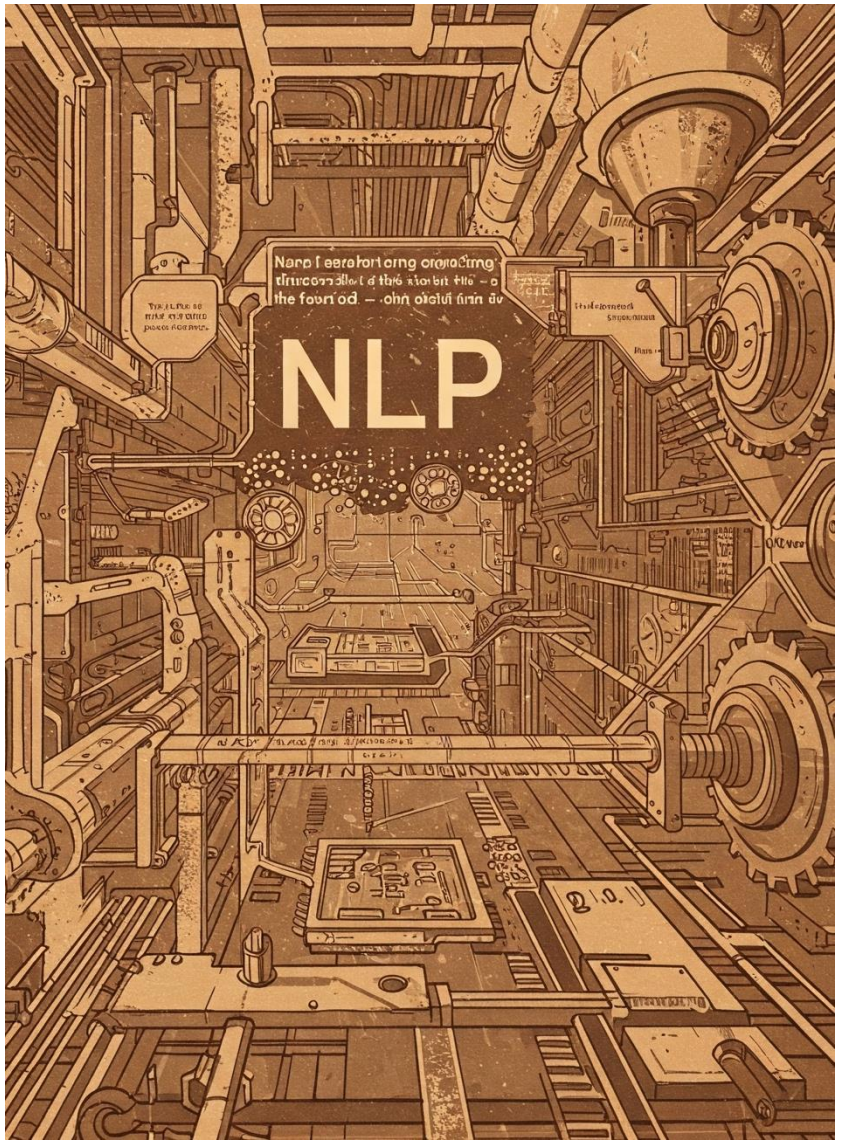
TRADITIONAL NATURAL LANGUAGE PROCESSING

Mehmet Can Yavuz, PhD.

Adapted from Julia Hockenmaier, NLP S2023 - course material
https://courses.grainger.illinois.edu/cs447/sp2023/



TRADITIONAL NATURAL LANGUAGE PROCESSING

Tokenizer/Segmentation

- to identify words and sentences

Morphological analyzer/POS-tagger

- to identify the part of speech and structure of words

Syntactic/semantic Parser

- to obtain the structure and meaning of sentences

Word sense disambiguation

- to identify the meaning of words

Coreference resolution

- to keep track of the various entities mentioned

TOKENIZATION

- We need to split text into words and sentences.
  - Tokens are word-like units.
- Tokenization is a language dependent task, where it becomes more challenging in some languages.
  - Languages like Chinese or Thai don't have spaces between words.
  - Even in English, this cannot be done deterministically based on space:

  he's, can't, 5-year-old, O'Reilly, rock 'n' roll, New York-based, wake him up

猫占领了婴儿床

The cat occupied the crib

- Tokenization is often regarded as trivial, and a mostly solved task.

PART-OF-SPEECH-TAGGING

TOKENIZATION | **MORPHOLOGY** | SYNTAX | SEMANTICS | DISCOURSE

Part-of-speech (POS) tagging is a common topic in NLP slides:

- POS tagging assigns grammatical categories (noun, verb, etc.) to words.
- Words can have multiple POS tags depending on context.

Verb	Det	Noun	Noun,	Name.
Open	the	pod	door,	Hal.

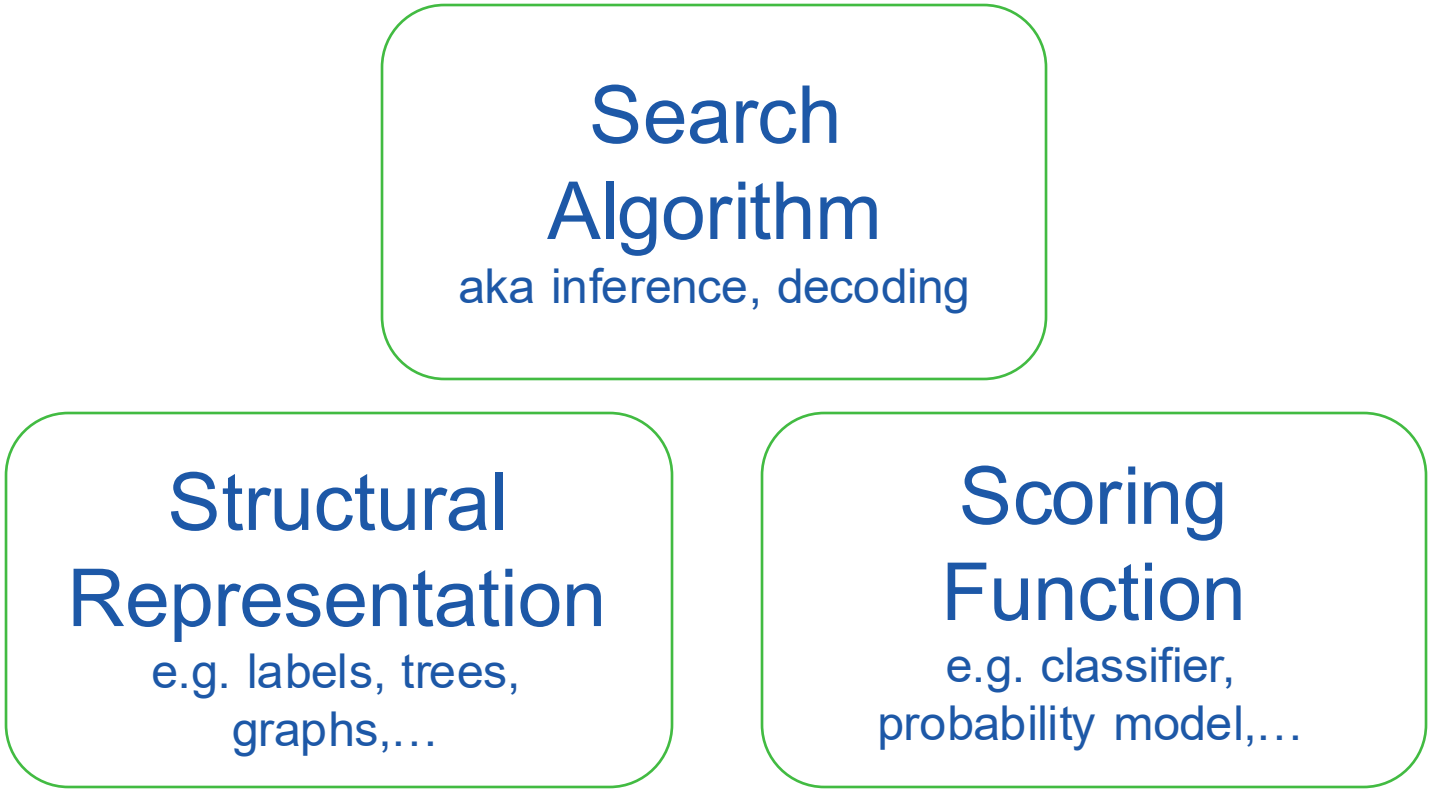
***open*:**
Verb, adjective, or noun?
Verb: **open** the door
Adjective: the **open-door**
Noun: in the **open**

AMBIGUITY IN NATURAL LANGUAGE

Language has different kinds of ambiguity, e.g.:

- **Structura**l ambiguity
  - *"I eat sushi **with tuna**"* vs. *"I eat sushi **with chopsticks**"*
  - *"I saw the man **with the telescope on the hill**"*
- **Lexical (word sense)** ambiguity
  - *"I went to the **bank**"*: financial institution or riverbank?
- **Referential** ambiguity
  - *"**John** saw **Jim**. **He** was drinking coffee."*
  - Who was drinking coffee?

DEALING WITH AMBIGUITY



DISAMBIGUATION REQUIRES STATISTICAL MODELS

Ambiguity is a core problem for any NLP task

Statistical models* are one of the main tools to deal with ambiguity.

* any machine learning model

These models need to be **trained** (estimated, learned) before they can be **used** (tested, evaluated).

“I MADE HER DUCK CASSOULET”

(Cassoulet = a French bean casserole)

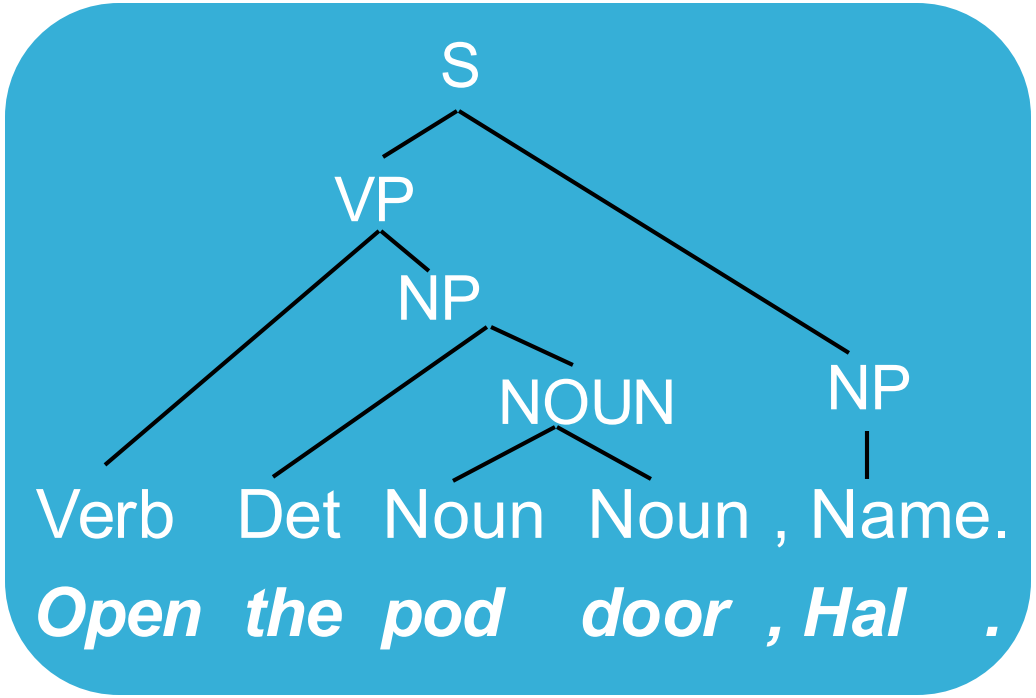
Another major problem in NLP is **coverage**: We will always encounter unfamiliar words and constructions.

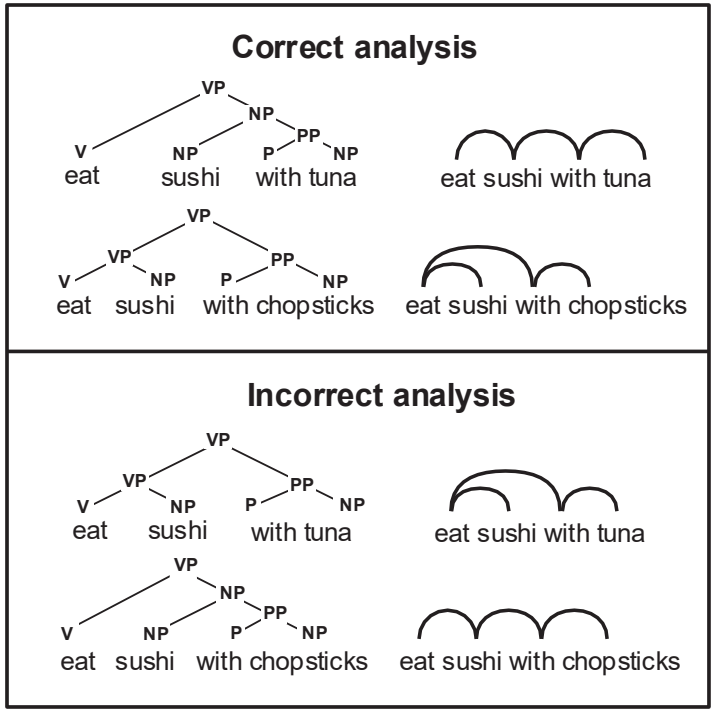
Our models need to be able to deal with this.

This means that our models need to be able to *generalize* from what they have been trained on to what they will be used on.

SYNTACTIC PARSING

TOKENIZATION | MORPHOLOGY | **SYNTAX** | SEMANTICS | DISCOURSE

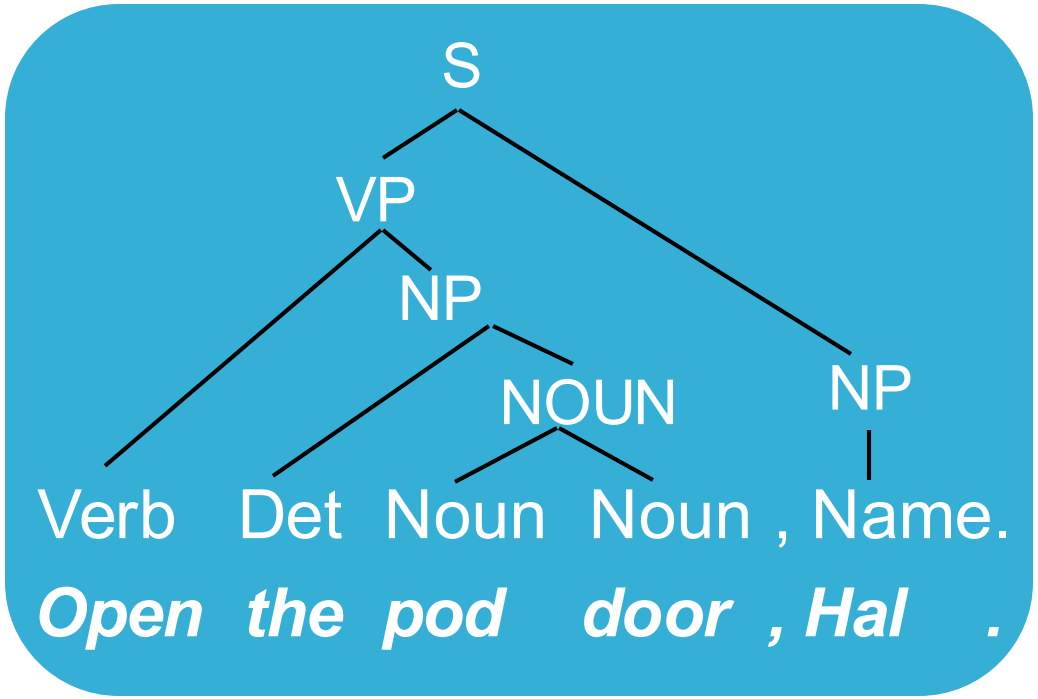




OBSERVATION: STRUCTURE CORRESPONDS TO MEANING

SEMANTIC PARSING

TOKENIZATION | MORPHOLOGY | SYNTAX | **SEMANTICS** | DISCOURSE




```
∃x∃y(pod_door(x) & Hal(y)
  & request(open(x, y)))
```

REPRESENTING MEANING

If a natural language understanding system needs to return a symbolic representation (or data structure) of the meaning of text, it needs a pre-defined **meaning representation language**.

"Deep" semantic analysis: (Variants of) **formal logic**

$$\exists x \exists y(pod\_door(x) \& Hal(y) \& request(open(x,y)))$$

"Shallow" semantic analysis: Template-filling
(Often used in information extraction)
Named-Entity Recognition: identify all organizations, locations, dates,... Event Extraction: identify specific events (e.g. 'protest', 'purchase', …)

We also distinguish between

Lexical semantics (the meaning of words) and

Compositional semantics (the meaning of sentences)

COREFERENCE RESOLUTION

On Monday, John went to Einstein's. He wanted to buy lunch. But the store was closed. That made him angry, so the next day he went to Green Street instead.

Can you answer the following questions?

Was Einstein's open for lunch on Monday? [No]

This requires the ability to identify that "*Einstein's*" and "*the store*" refer to the same entity. (**coreference resolution**).

On which day did John go to Green Street? [On Tuesday].

This requires the ability to understand the **implicit** information that "the next day" means really "the next day after Monday" (and the knowledge that that is a Tuesday).

NLP PIPELINE: ASSUMPTIONS

Each step in the NLP pipeline embellishes the input with **explicit information about its linguistic structure**

POS tagging: Parts of speech of word,

Syntactic parsing: Grammatical structure of sentence,….

Each step in the NLP pipeline requires **its own explicit ("symbolic") output representation:**

POS tagging requires a **POS tag set**

(e.g. NN=common noun singular, NNS = common noun plural, …)

Syntactic parsing requires **constituent** or **dependency labels**
(e.g. NP = noun phrase, or nsubj = nominal subject)

These representations should capture **linguistically appropriate generalizations/abstractions**

Designing these representations requires linguistic expertise

NLP PIPELINE: SHORTCOMINGS

Each step in the pipeline relies on a **learned model** that will return the *most likely* representations

- This requires a lot of **annotated training data** for each step
- Annotation is **expensive** and sometimes **difficult** (people are not 100% accurate)
- These models are **never 100% accurate**
- Models make more mistakes if their input contains mistakes

How do we know that we have captured the **"*right*" generalizations** when designing representations?

- Some representations are **easier to predict** than others
- Some representations are **more useful** for the next steps in the pipeline than others
- But we won't know how easy/useful a representation is until we have a model that we can plug into a particular pipeline

SIDESTEPPING THE NLU PIPELINE

Many current neural approaches for natural language understanding and generation go directly from the raw input to the desired final output.

With large amounts of training data, this often works better than the traditional approach.

— We will soon discuss why this may be the case.

But these models don't solve everything:

— How do we incorporate knowledge, reasoning, etc. into these models?

— What do we do when don't have much training data? (e.g. when we work with a low-resource language)

REVIEW OF THE KEY CONCEPTS

Tokenizer/Segmenter

Morphological analyzer/POS-tagger

Word sense disambiguation

Syntactic/semantic Parser

Coreference resolution