COMP61332
Weeks 1, 2 and 5 Revision

COMP61332: Text Mining
Week 5
Riza Batista-Navarro

Why is NLP difficult?

Phonological

multiple interpretations due to **how it sounds** (important in speech processing)

e.g., *"I will be [writing|riding] this weekend."* (writing a piece of text or horseback riding?)

Lexical

multiple interpretations due to **a word having multiple senses**

e.g., *"I am going to the bank."* (financial entity or river?)

Why is NLP difficult?

Syntactic

- due to **a word having more than one possible part of speech**

  e.g., *"I saw her duck."* (animal [noun] or bend down [verb]?)

- or, due to **prepositional phrase attachment**

  e.g., *"I saw the man on the hill with the telescope."* (who has the telescope?)

Source: *https://americanenglish.state.gov/*

Why is NLP difficult?

Source: https://infinitewellbeing.co.uk/

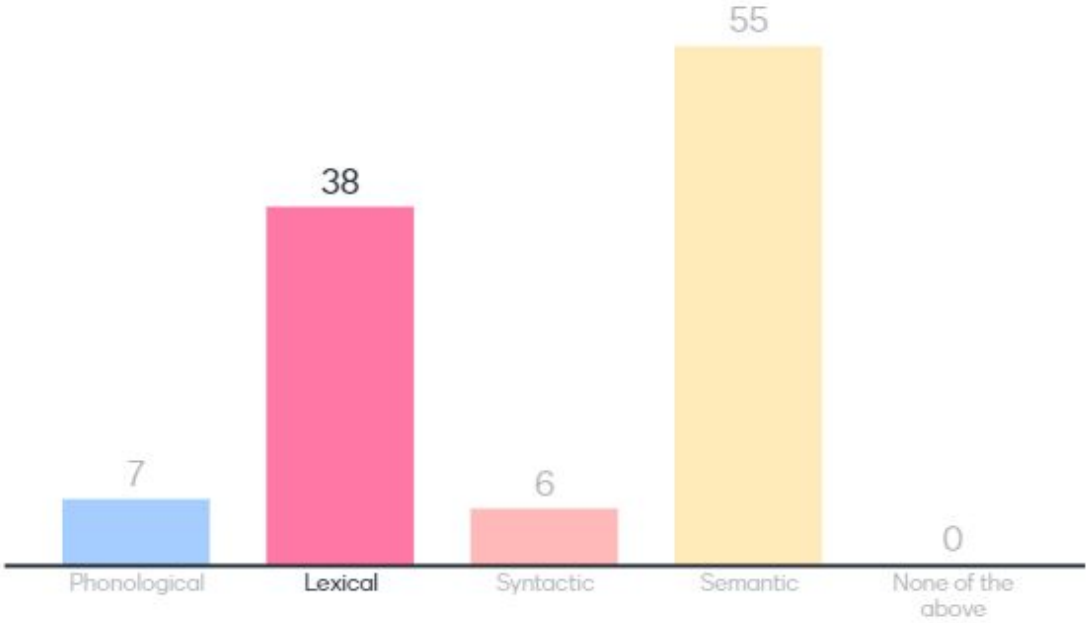
Semantic 语义的

multiple possible interpretations unless knowledge of the world is available

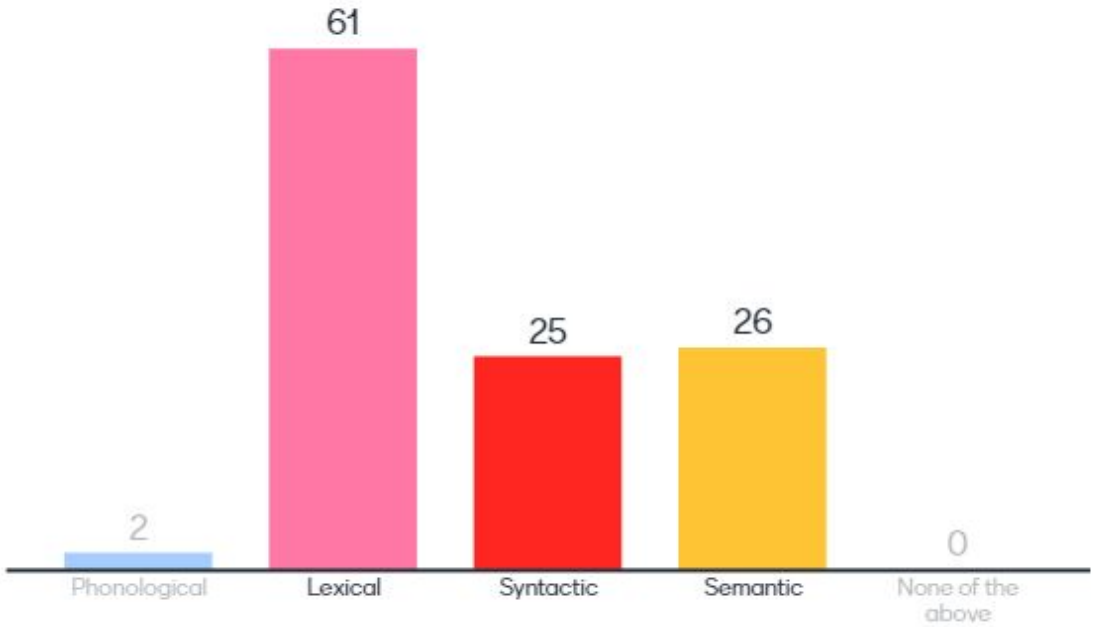
e.g., *"The children ate the cookies because they were very hungry."* (What does "they" refer to?)

- to a human: "children" (obviously)
- to a machine: "children" or "cookies" (unless it knows that cookies cannot feel hunger, or that eating usually follows being hungry)

What type(s) of ambiguity can be found in the following sentence? *I bought a mouse and took it home.*



What type(s) of ambiguity can be found in the following sentence? *I saw her bat.*



词汇的

Sentence Segmentation

Determination of boundaries between sentences

Sentences used in subsequent NLP tasks

Is it enough to detect the **full stop**?

- Could be an end-of-sentence (EOS) marker
- Or an end of abbreviation marker
- Or both?

Examples of useful rules or features

First character after potential EOS char

- Should be uppercase? Problematic for some languages, e.g. German
- Permissible chars after potential EOS, e.g. lowercase characters?

Abbreviations

- titles not likely to occur at EOS (e.g., Dr. Jones)
- company indicators could occur at EOS (e.g., MySocialMedia Inc.)

Tokenisation

How many tokens in: *you're*

- 1? (you‘re)
- 2? (you + are)
- 3? (you + ‘ + re)

How about: *president's speech*

- president‘s + speech
- president + ‘s + speech

How about: *Carla's home*

- home of Carla (?)
- Carla is home (?)

WordPunctTokenizer

TreebankWordTokenizer

PunktWordTokenizer

To split or not to split?

- Sentence segmentation (split)
- Tokenisation (split)
- Named entity recognition (combine)

In other words: tokenisation is **knowing when to split** (not when to combine)

Annotation Formats: Boundary Notation

Strengths

- simple

Limitations

- cannot handle hierarchical or structured annotations, e. g., nested entities, relations, events

Annotation Formats: Inline markup language elements

Strengths

- can handle annotations which are hierarchical (e.g., nested NEs, trees) and structured (e.g., events)

Limitations

- requires substantial processing with standard XML parsers
- impossible to encode overlapping/intersecting annotations, e.g.,

second Iraqi city of Basra

Annotation Formats: Stand-off annotations

Strengths

- original raw text is left untouched
- can handle structured and overlapping annotations

Limitations

- not readily human-readable

Part-of-Speech (POS) Tagging

Assign POS tags to individual **tokens**

Tokenisation is usually performed before (although some approaches do tokenisation and POS tagging jointly)

*Book/**VB** that/**DT** flight/**NN** ./.*

*Does/**VBZ** that/**DT** flight/**NN** serve/**VB** dinner/**NN** ?/.*

How can we disambiguate?

- A token is very unlikely to be a verb if its preceding word is a determiner

  I want a go
- A token is unlikely to be a noun if the immediately preceding word is *to*

  I want to go
- A token is more likely to be a possessive pronoun when followed by a common noun

  He stroked her cat
- ...but not always

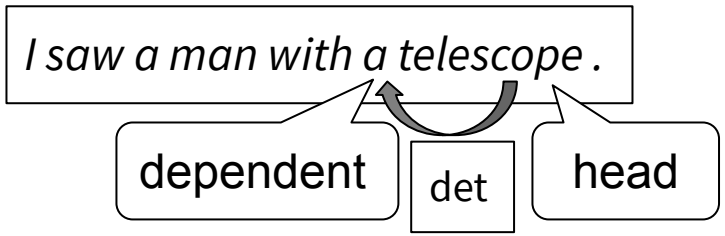
  He gave her money

Dependency structure

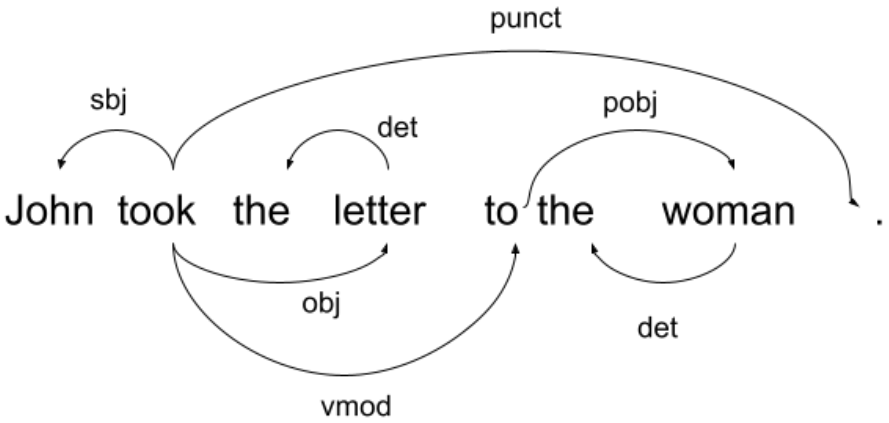
Sentence structure based on **dependencies**

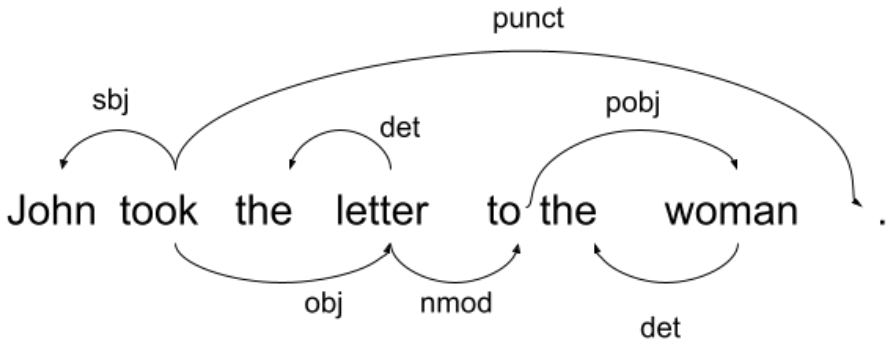
Analysis of dependency structure:

- looking for dependencies between token pairs
- drawing a link between two tokens and specifying a **label**: grammatical function

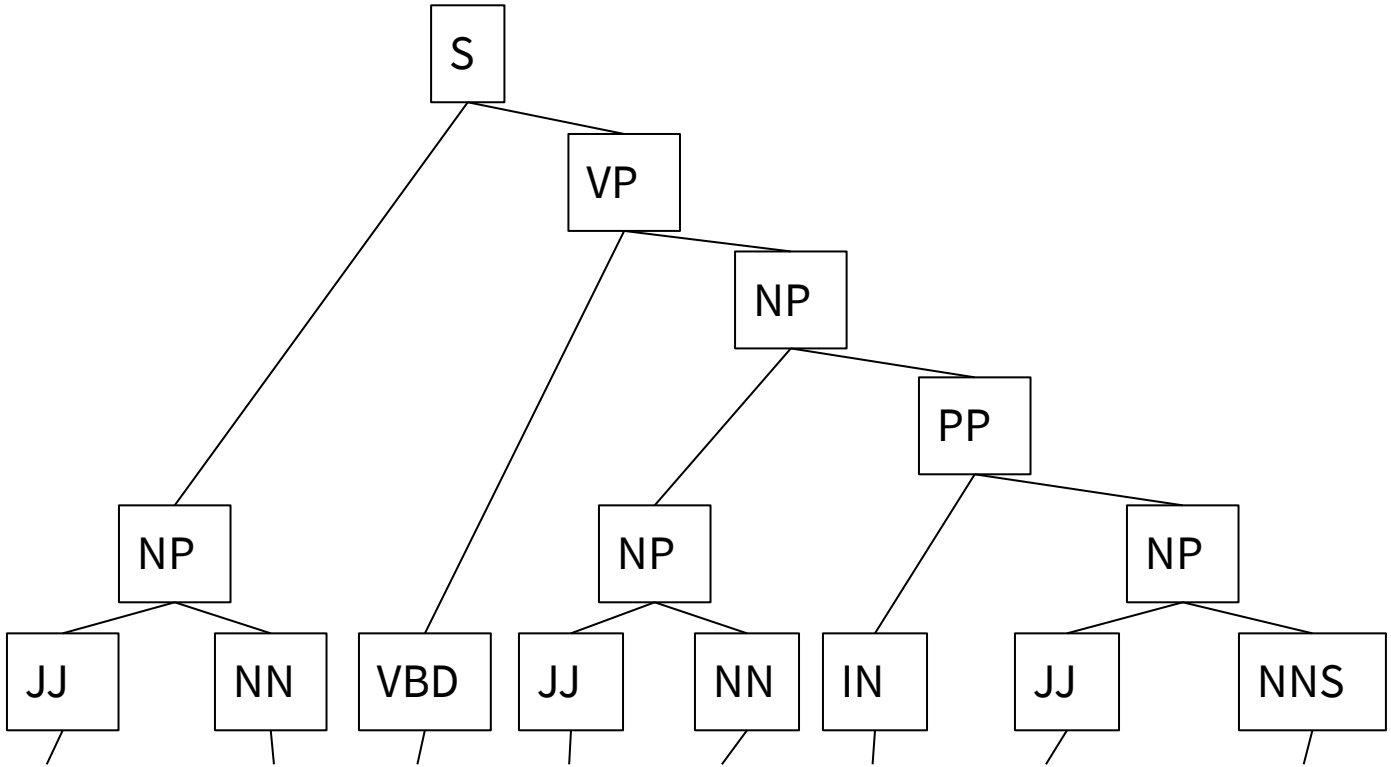


On paper, draw the dependency graph for each of the two interpretations of: *John took the letter to the woman.*



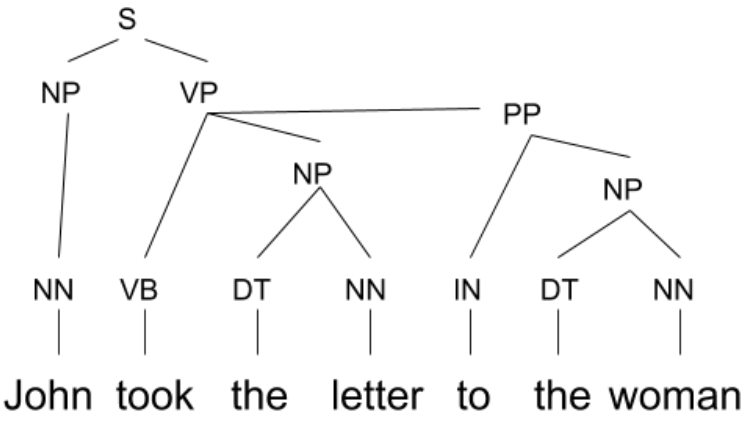


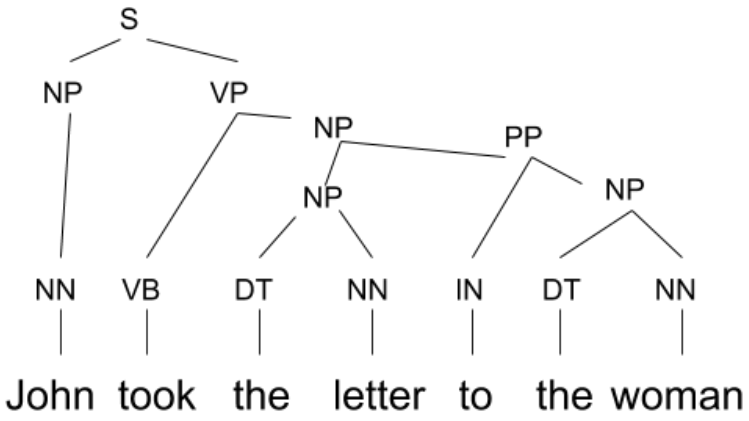
Phrase structure



Economic news had little effect on financial markets

On paper, draw the phrase structure tree for each of the two interpretations of: *John took the letter to the woman.*





Relation Extraction

Discerning **relationships** amongst **entities** in piece of text

[American Airlines], a unit of [AMR Corp.], immediately matched the move, spokesman [Tim Wagner] said. [United], a unit of [UAL Corp.], said the increase took effect Thursday and applies to most routes.

Relationships:

Tim Wagner *spokesman for* American Airlines

United *unit of* UAL Corp.

American Airlines *unit of* AMR Corp.

Pattern-based Approaches

Using **regular expressions** (regexes)

Example: extract *airline-hub cities* relations

> regex: */* has a hub at *\/*
>
> would match: *KLM has a hub at Amsterdam.*
>
> but also false positives: *The wheel has a hub at its centre.*

Pattern-based Approaches

Regex can be modified to put entity constraints

/[ORGANISATION] has a hub at [LOCATION]/

but still problematic as it would miss:

easyJet has established a hub at Liverpool.

Ryanair has a continental hub at Charleroi, Belgium.

You searched the Internet for text mentioning those entities and found the sentences below. Write down possible regexes using [PERSON] and [PRODUCT].

- Thomas Edison invented the first working phonograph.
- The phonograph was invented in 1877 by Thomas Edison.
- The first working phonograph was invented by Thomas Edison in 1877.
- Thomas Edison is the esteemed inventor of the phonograph.

Possible regexes

- [PERSON] invented the * [PRODUCT]
- [PRODUCT] was invented * by [PERSON]
- [PERSON] is the * inventor of the [PRODUCT]
- invention of the [PRODUCT] by [PERSON]

Kappa coefficient

		Annotator 1		
		yes	no	total
Annotator 2	yes	31	1	32
	no	2	6	8
	total	33	7	40

$$\textit{Kappa} = (P(a)-P(e))/(1-P(e))$$

$$= (0.925-0.695)/(1-0.695) = 0.754$$

Precision and Recall

Precision: fraction of annotated items that are correct

Recall: fraction of correct items that are annotated

Confusion matrix:

	Correct	Not correct
Annotated	True positive (TP)	False positive (FP)
Not annotated	False negative (FN)	True negative (TN)

Precision = TP / (TP+FP) 正确判断 / 标记为阳性

Recall = TP / (TP + FN) 正确判断/ 真正阳性

F-score (a.k.a. F-measure, F1-score)

Weighted harmonic mean

$$F_\beta = (\beta^2 + 1)PR/\beta^2P+R$$

Usually balanced F1 measure is used, where β=1

$$F1 = 2PR/(P+R)$$

Harmonic mean is a more conservative average (truer picture)

Multiple Categories: Micro vs Macro-averaging

Category	TPs	FPs	FNs	Precision	Recall
Person	78	5	33	0.94	0.70
Location	20	3	2	0.87	0.91

How do we report combined performance for Person and Location?
Option 1: **Macro-averaging** -- Simply get the average

$$P_{Person+Location} = (0.94+0.87)/2 = 0.91$$

macro方法没考虑样本不均匀

$$R_{Person+Location} = (0.70+0.91)/2 = 0.81$$

$$F1_{Person+Location} = (2*0.91*0.81)/(0.91+0.81) = 0.86$$

Multiple Categories: Micro vs Macro-averaging

Category	TPs	FPs	FNs	Precision	Recall
Person	78	5	33	0.94	0.70
Location	20	3	2	0.87	0.91

How do we report combined performance for Person and Location?

Option 2: **Micro-averaging** -- Pool together the TPs, FPs and FNs

micro方法考虑样本占比

$$P_{Person+Location} = (78+20)/((78+20)+(5+3)) = 0.92$$

$$R_{Person+Location} = (78+20)/((78+20)+(33+2)) = 0.74$$

$$F1_{Person+Location} = (2*0.92*0.74)/(0.92+0.74) = 0.82$$

Which is better?

Category	TPs	FPs	FNs	Precision	Recall
Person	78	5	33	0.94	0.70
Location	20	3	2	0.87	0.91

Macro-averaging does not consider class imbalance; micro-averaging is less sensitive to imbalance

Weighted average:
average weighted by the number of true instances for each class