Word Sense Disambiguation and Coreference Resolution

Natalie Parde

UIC CS 421

This Week's Topics

Tuesday

- Word Senses
- WordNet
- Word Sense Disambiguation

Thursday

- Coreference Resolution
- Referring Expressions
- Coreference Resolution Approaches
- Evaluating Coreference Resolution

This Week's Topics

Tuesday

- Word Senses
- WordNet
- Word Sense Disambiguation

Thursday

- Coreference Resolution
- Referring Expressions
- Coreference Resolution Approaches
- Evaluating Coreference Resolution

Words can carry many meanings.

- The different possible meanings for a word are its **senses**
- For example:
  - Book_1: To reserve something
  - Book_2: A large written source of fiction or non-fiction text
  - Book_3: To move quickly
- Word senses can be represented in numerous ways

Glosses

- Dictionaries or thesauruses often provide definitions for each sense of a word, referred to as **glosses**
  - Not a formal meaning representation!
  - Written to facilitate human understanding of the senses a word may take
  - May be circular
    - Direct self-reference (e.g., “Right: Located nearer the right hand”)
    - Implicit self-reference (e.g., “Left: Located nearer to the side opposite the right”)
    - Complementary external reference (e.g., “Red: The color of a ruby” and “Ruby: A red gemstone”)

Glosses

- Even if glosses aren't meaning representations themselves, they can still be useful for computationally modeling word senses
  - Glosses are sentences
    - Convenient input for representation learning
  - Glosses are often accompanied by example sentences
    - Additional useful data

Dictionary-Based Sense Definitions

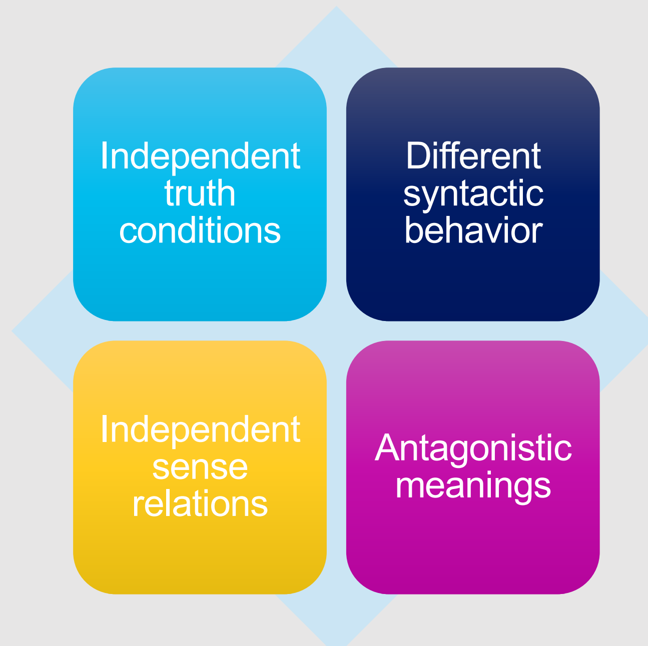
- Senses can be defined through their relationship with other senses
- Given a large database of senses and the relations between them, we can leverage these associations to perform semantic tasks

Words with numerous senses are **polysemous**.

- **Polysemy:** The phenomenon in which a single word is associated with two or more distinct senses
- There is no limit to how many senses a word can have!
- Sense distinctions vary depending on the dictionary:
  - Some dictionaries represent very fine-grained distinctions as different senses
  - Computational resources usually focus on broader, more coarse-grained sense categories

How can we distinguish between senses?

- Word embeddings offer continuous, high-dimensional word representations that aren't easily discretized into sentences
  - Contextual word embeddings produce a different representation for each unique use of a word
- Dictionaries separate words into senses based on predetermined criteria



Practical Technique for Determining Sense Distinction

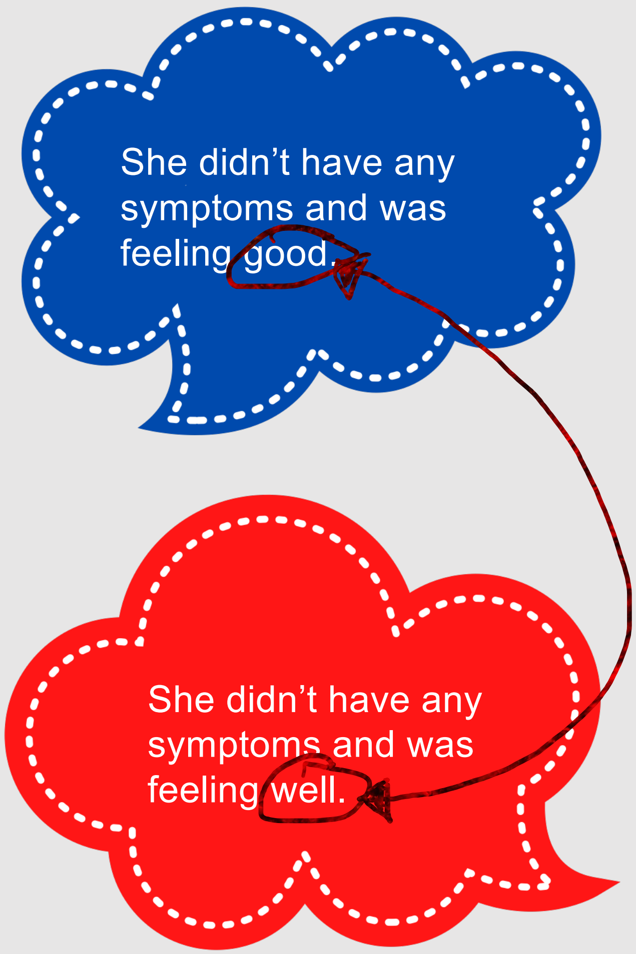
- Conjoin two uses of a word in a sentence
- For example:
  - Which of those flights serve ice cream?
  - Does American Airlines serve Chicago?
  - Does American Airlines serve ice cream and Chicago?
- If you observe that this creates a **zeugma** (a conjunction of antagonistic uses of the same word), consider these as distinct senses

How do word senses relate to one another?

- Many types of relations can exist between word senses
- Particularly useful for NLP purposes:
  - Synonymy
  - Antonymy
  - Hypernymy

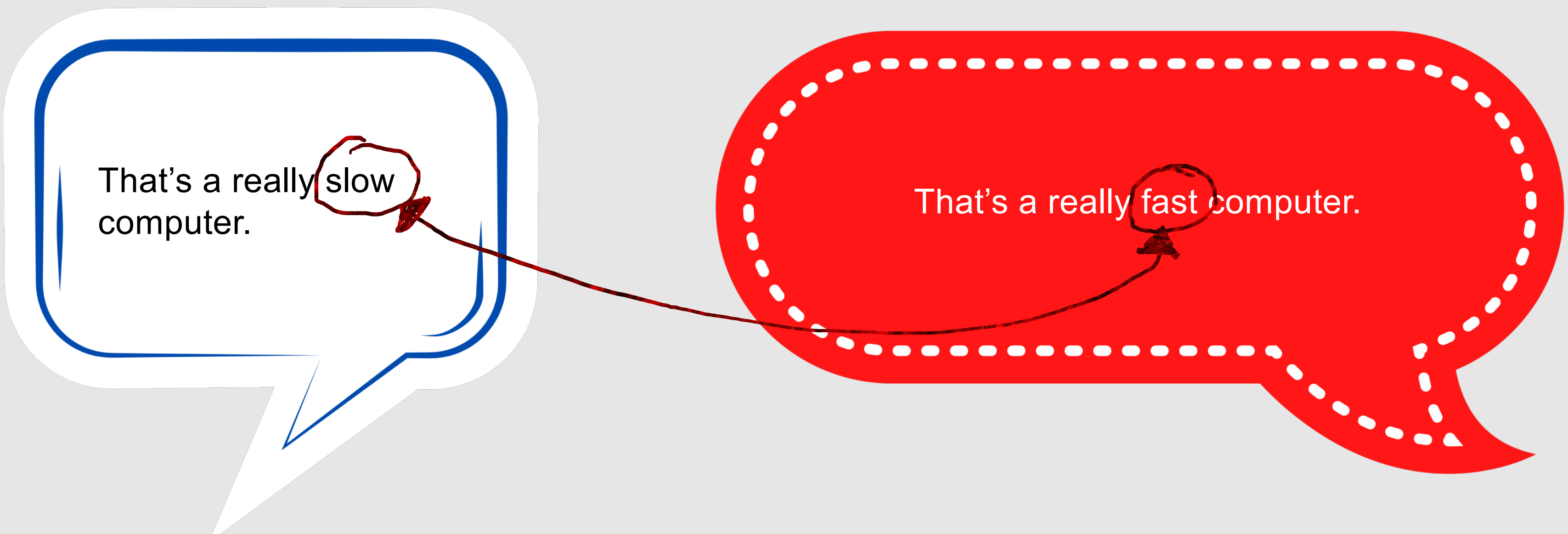
Synonymy

- Occurs when two word senses are highly similar to one another
  - Substituting one for another should convey essentially the same meaning
- *All* senses for both words do not need to be highly similar



Antonymy

- Occurs when two word senses convey opposite meaning to one another
- The word senses should otherwise be interchangeable in similar contexts

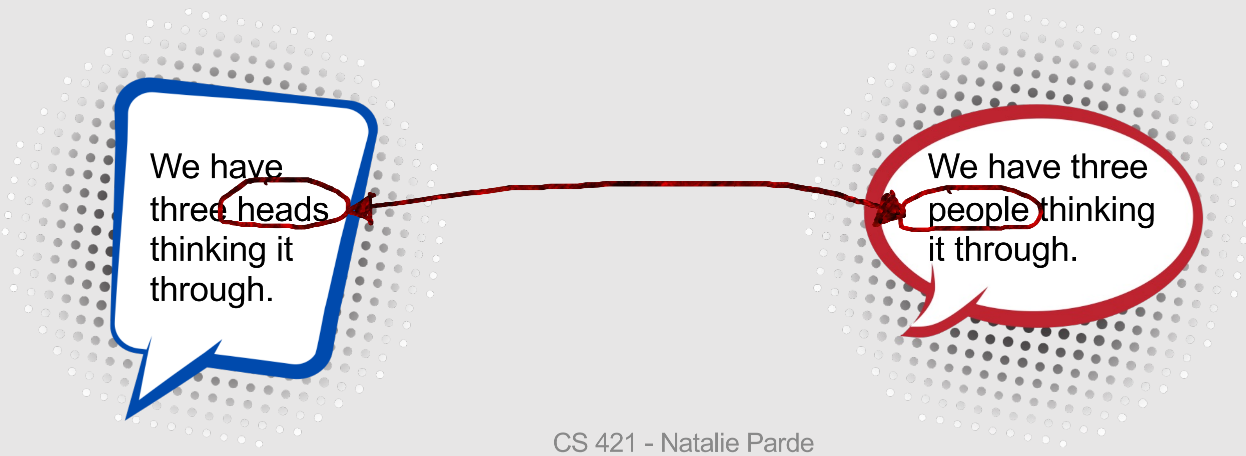


Hypernymy

- Occurs when one word sense is a generalization, or broader category, of another
- The word sense that is more general is the **hypernym**
- The word sense that is the more specific subclass of the broader word sense is the **hyponym**

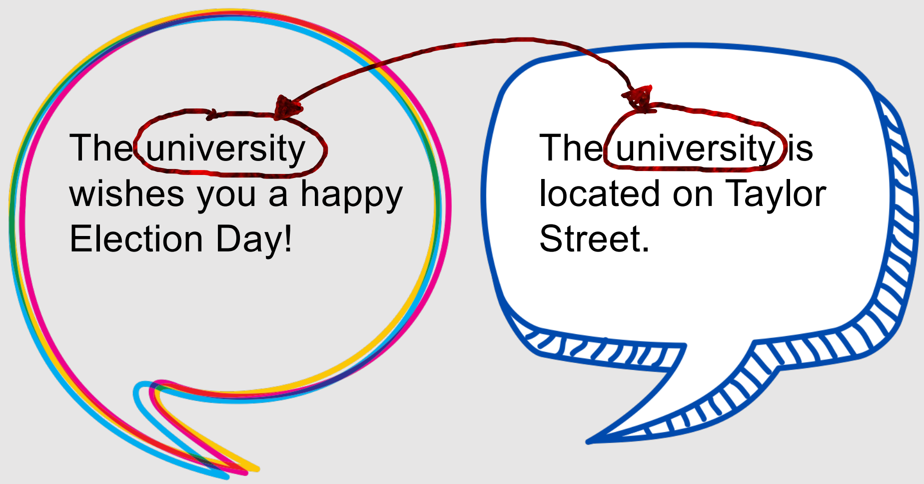
Meronymy

- Closely related to hypernymy
- Occurs when one word sense refers to a part of another word sense
- The word sense that is the more general whole is the **holonym**



Structured Polysemy

- Semantically related senses associated with the same word
- Often seen when one word sense refers to an organization, and another sense refers to the building housing that organization



Metonymy

- Structured polysemy for which one aspect of a concept or entity is used to refer to other aspects of the entity or the entity itself
- Common examples are also found in:
  - Pairings between authors or artists and their works
  - Pairings between plants and their respective foods

Did you see the new Van Gogh at the art institute?

This Week's Topics

Tuesday

- Word Senses
- WordNet
- Word Sense Disambiguation

Thursday

- Coreference Resolution
- Referring Expressions
- Coreference Resolution Approaches
- Evaluating Coreference Resolution

WordNet Search - 3.1
- WordNet home page - Glossary - Help

Word to search for: mask [Search WordNet]

Display Options: (Select option to change) [Change]

Key: "S:" = Show Synset (semantic) relations, "W:" = Show Word (lexical) relations
Display options for sense: (frequency) {offset} <lexical filename > [lexical file number] (gloss) "an example sentence"
Display options for word: word#sense number (sense key)

Noun

- (1){03730361} <noun.artifact>[06] S: (n) **mask#1 (mask%1:06:00::)** (a covering to disguise or conceal the face)
- (1){01051399} <noun.act>[04] S: (n) **mask#2 (mask%1:04:00::)** (activity that tries to conceal something) *"no mask could conceal his ignorance"; "they moved in under a mask of friendship"*
- {08270371} <noun.group>[14] S: (n) masquerade#1 (masquerade%1:14:00::), masquerade party#1 (masquerade_party%1:14:00::), masque#1 (masque%1:14:00::), **mask#3 (mask%1:14:00::)** (a party of guests wearing costumes and masks)
- {03730526} <noun.artifact>[06] S: (n) **mask#4 (mask%1:06:01::)** (a protective covering worn over the face)

Verb

- (1){02152033} <verb.perception>[39] S: (v) dissemble#2 (dissemble%2:39:00::), cloak#1 (cloak%2:39:00::), **mask#1 (mask%2:39:00::)** (hide under a false appearance) *"He masked his disappointment"*
- (1){01361031} <verb.contact>[35] S: (v) **mask#2 (mask%2:35:00::)** (put a mask on or cover with a mask) *"Mask the children for Halloween"*
- {02163017} <verb.perception>[39] S: (v) disguise#1 (disguise%2:39:00::), **mask#3 (mask%2:39:01::)** (make unrecognizable) *"The herb masks the garlic taste"; "We disguised our faces before robbing the bank"*
- {01361558} <verb.contact>[35] S: (v) **mask#4 (mask%2:35:02::)** (cover with a sauce) *"mask the meat"*
- {01361440} <verb.contact>[35] S: (v) **mask#5 (mask%2:35:01::)**, block out#3 (block_out%2:35:00::) (shield from light)

WordNet

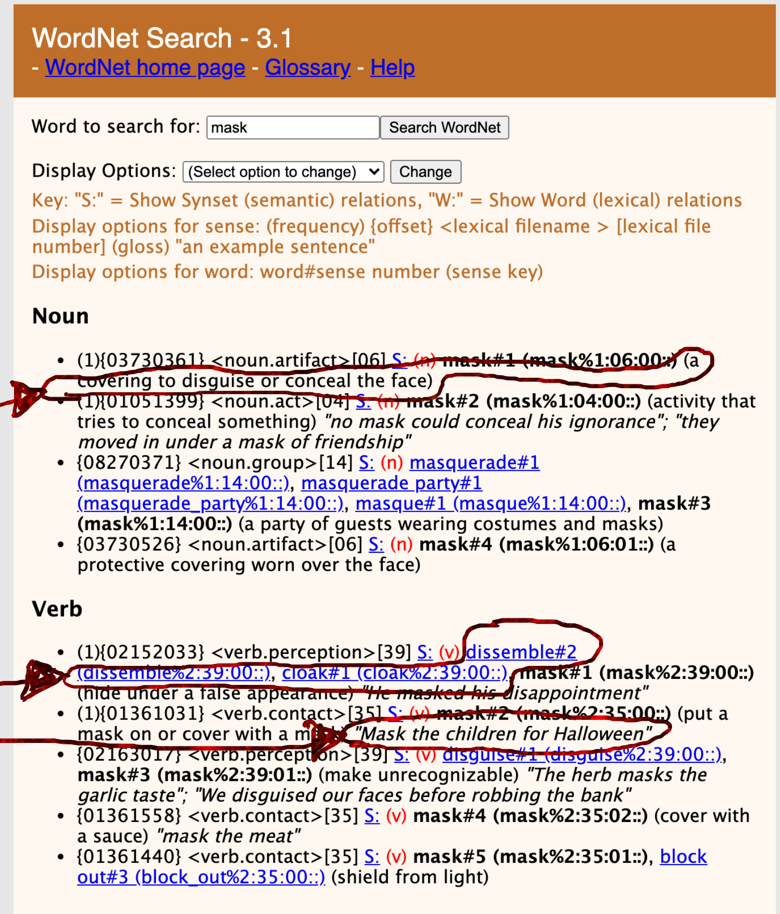
- Large lexical resource with information about:
  - Nouns
  - Verbs
  - Adjectives and adverbs
- Each entry is annotated with one or more **senses**
- Each sense provides a variety of information

WordNet

- Statistics for English WordNet 3.0:
  - 117,798 nouns
  - 11,529 verbs
  - 22,479 adjectives
  - 4,481 adverbs
- Average noun has 1.23 senses
- Average verb has 2.16 senses

WordNet Entries

- Senses contain:
  - **Gloss**
    - A definition of the sense
  - (Often) list of synonyms
    - Commonly referred to as a **synset**
  - (Sometimes) example sentence



WordNet Search - 3.1
- WordNet home page - Glossary - Help

Word to search for: mask Search WordNet

Display Options: (Select option to change) Change

Key: "S:" = Show Synset (semantic) relations, "W:" = Show Word (lexical) relations
Display options for sense: (frequency) {offset} <lexical filename > [lexical file number] (gloss) "an example sentence"
Display options for word: word#sense number (sense key)

Noun

- (1){03730361} <noun.artifact>[06] S: (n) **mask#1 (mask%1:06:00::)** (a covering to disguise or conceal the face)
- (1){01051399} <noun.act>[04] S: (n) **mask#2 (mask%1:04:00::)** (activity that tries to conceal something) *"no mask could conceal his ignorance"; "they moved in under a mask of friendship"*
- {08270371} <noun.group>[14] S: (n) masquerade#1 (masquerade%1:14:00::), masquerade party#1 (masquerade_party%1:14:00::), masque#1 (masque%1:14:00::), **mask#3 (mask%1:14:00::)** (a party of guests wearing costumes and masks)
- {03730526} <noun.artifact>[06] S: (n) **mask#4 (mask%1:06:01::)** (a protective covering worn over the face)

Verb

- (1){02152033} <verb.perception>[39] S: (v) dissemble#2 (dissemble%2:39:00::), cloak#1 (cloak%2:39:00::), **mask#1 (mask%2:39:00::)** (hide under a false appearance) *"He masked his disappointment"*
- (1){01361031} <verb.contact>[35] S: (v) **mask#2 (mask%2:35:00::)** (put a mask on or cover with a mask) *"Mask the children for Halloween"*
- {02163017} <verb.perception>[39] S: (v) disguise#1 (disguise%2:39:00::), **mask#3 (mask%2:39:01::)** (make unrecognizable) *"The herb masks the garlic taste"; "We disguised our faces before robbing the bank"*
- {01361558} <verb.contact>[35] S: (v) **mask#4 (mask%2:35:02::)** (cover with a sauce) *"mask the meat"*
- {01361440} <verb.contact>[35] S: (v) **mask#5 (mask%2:35:01::)**, block out#3 (block_out%2:35:00::) (shield from light)

Synsets

- Fundamental unit associated with WordNet entries
- Participate in lexical sense relations
- Facilitate relational navigation through the WordNet hierarchy

Sense Relations

- **Hypernym:** Relation between a concept and its superordinate
  - *Food* is a hypernym of *cake*
- **Hyponym:** Relation between a concept and its subordinate
  - *Corgi* is a hyponym of *dog*
- **Meronym:** Relation between a part and its whole
  - *Wheel* is a meronym of *car*
- **Holonym:** Relation between a whole and its parts
  - *Car* is a holonym of *wheel*
- **Antonym:** Relation between two semantically opposite concepts
  - *Leader* is an antonym of *follower*

Taxonomic Entities in WordNet

- Two kinds of taxonomic entities
  - **Classes**
  - **Instances**
- Instances: Individual proper nouns that represent unique entities
  - Chicago
- Classes: Generalized groups of instances
  - city

Additional Sense Relations

- Noun relations have a few additional distinctions:
  - **Instance hypernyms** are relationships from instances to their concepts (e.g., "Austen → author" rather than "breakfast → meal")
  - **Derivations** are lemmas with the same morphological root (e.g., "destruction ↔ destroy")
- So do verbs:
  - **Troponyms** are relationships from events to subordinate events (e.g., "stroll" is a troponym of "walk")
  - **Entailments** are relationships from verbs to the verbs they entail (e.g., "borrow" entails "obtain")

Lexicographic Categories

- Coarse-grained semantic categories
  - Often referred to as **supersenses**
- 26 categories for nouns
- 15 categories for verbs

Category	Example	Category	Example	Category	Example
Act	service	Group	place	Plant	tree
Animal	dog	Location	area	Possession	price
Artifact	car	Motive	reason	Process	process
Attribute	quality	Natural Event	experience	Quantity	amount
Body	hair	Natural Object	flower	Relation	portion
Cognition	way	Other	stuff	Shape	square
Communication	review	Person	people	State	pain
Feeling	discomfort	Phenomenon	result	Substance	oil
Food	food			Time	day

- {03211439} <noun.artifact>[06] S: (n) disguise#2 (disguise%1:06:00::) (any attire that modifies the appearance in order to conceal the wearer's identity)
  - {02759103} <noun.artifact>[06] S: (n) attire#1 (attire%1:06:00::), garb#1 (garb%1:06:00::), dress#2 (dress%1:06:01::) (clothing of a distinctive style or for a particular occasion) *"formal attire"; "battle dress"*
    - {03055525} <noun.artifact>[06] S: (n) clothing#1 (clothing%1:06:00::), article of clothing#1 (article_of_clothing%1:06:00::), vesture#2 (vesture%1:06:00::), wear#2 (wear%1:06:00::), wearable#1 (wearable%1:06:00::), habiliment#1 (habiliment%1:06:00::) (a covering designed to be worn on a person's body)
      - {03127399} <noun.artifact>[06] S: (n) covering#2 (covering%1:06:00::) (an artifact that covers something else (usually to protect or shelter or conceal it))
        - {00022119} <noun.Tops>[03] S: (n) artifact#1 (artifact%1:03:00::), artefact#1 (artefact%1:03:00::) (a man–made object taken as a whole)
          - {00003553} <noun.Tops>[03] S: (n) whole#2 (whole%1:03:00::), unit#6 (unit%1:03:00::) (an assemblage of parts that is regarded as a single entity) *"how big is that part compared to the whole?"; "the team is a unit"*
            - {00002684} <noun.Tops>[03] S: (n) object#1 (object%1:03:00::), physical object#1 (physical_object%1:03:00::) (a tangible and visible entity; an entity that can cast a shadow) *"it was full of rackets, balls and other objects"*
              - {00001930} <noun.Tops>[03] S: (n) physical entity#1 (physical_entity%1:03:00::) (an entity that has physical existence)
                - {00001740} <noun.Tops>[03] S: (n) entity#1 (entity%1:03:00::) (that which is perceived or known or inferred to have its own distinct existence (living or nonliving))

- (1){03730361} <noun.artifact>[06] S: (n) **mask#1 (mask%1:06:00::)** (a covering to disguise or conceal the face)
  - *direct hyponym* / *full hyponym*
  - *direct hypernym* / ***inherited hypernym*** / *sister term*
    - {03127399} <noun.artifact>[06] S: (n) covering#2 (covering%1:06:00::) (an artifact that covers something else (usually to protect or shelter or conceal it))
      - {00022119} <noun.Tops>[03] S: (n) artifact#1 (artifact%1:03:00::), artefact#1 (artefact%1:03:00::) (a man–made object taken as a whole)
        - {00003553} <noun.Tops>[03] S: (n) whole#2 (whole%1:03:00::), unit#6 (unit%1:03:00::) (an assemblage of parts that is regarded as a single entity) *"how big is that part compared to the whole?"; "the team is a unit"*
          - {00002684} <noun.Tops>[03] S: (n) object#1 (object%1:03:00::), physical object#1 (physical_object%1:03:00::) (a tangible and visible entity; an entity that can cast a shadow) *"it was full of rackets, balls and other objects"*
            - {00001930} <noun.Tops>[03] S: (n) physical entity#1 (physical_entity%1:03:00::) (an entity that has physical existence)
              - {00001740} <noun.Tops>[03] S: (n) entity#1 (entity%1:03:00::) (that which is perceived or known or inferred to have its own distinct existence (living or nonliving))

- {03098030} <noun.artifact>[06] S: (n) consumer goods#1 (consumer_goods%1:06:00::) (goods (as food or clothing) intended for direct use or consumption)
  - {03080712} <noun.artifact>[06] S: (n) commodity#1 (commodity%1:06:00::), trade good#1 (trade_good%1:06:00::), good#4 (good%1:06:00::) (articles of commerce)
    - {00022119} <noun.Tops>[03] S: (n) artifact#1 (artifact%1:03:00::), artefact#1 (artefact%1:03:00::) (a man–made object taken as a whole)
      - {00003553} <noun.Tops>[03] S: (n) whole#2 (whole%1:03:00::), unit#6 (unit%1:03:00::) (an assemblage of parts that is regarded as a single entity) *"how big is that part compared to the whole?"; "the team is a unit"*
        - {00002684} <noun.Tops>[03] S: (n) object#1 (object%1:03:00::), physical object#1 (physical_object%1:03:00::) (a tangible and visible entity; an entity that can cast a shadow) *"it was full of rackets, balls and other objects"*
          - {00001930} <noun.Tops>[03] S: (n) physical entity#1 (physical_entity%1:03:00::) (an entity that has physical existence)
            - {00001740} <noun.Tops>[03] S: (n) entity#1 (entity%1:03:00::) (that which is perceived or known or inferred to have its own distinct existence (living or nonliving))

Hierarchical Structure

Check out WordNet for yourself!

- You can browse WordNet using the link here: http://wordnetweb.princeton.edu/perl/webwn
- You can also programmatically access WordNet using NLTK: https://www.nltk.org/howto/wordnet.html

This Week's Topics

Tuesday

- Word Senses
- WordNet
- Word Sense Disambiguation

Thursday

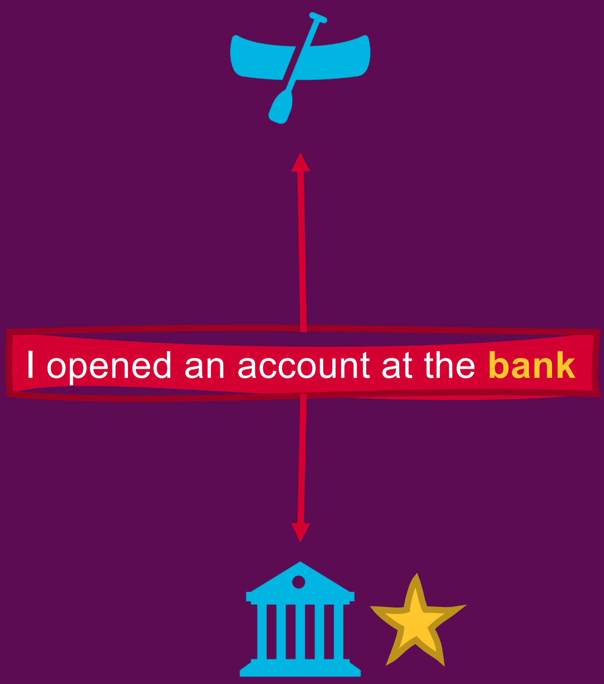
- Coreference Resolution
- Referring Expressions
- Coreference Resolution Approaches
- Evaluating Coreference Resolution

Somehow, NLP systems need to be able to determine which sense is used in a given context.

- How can we do this?
  - Word sense disambiguation

What is word sense disambiguation?

- **Word sense disambiguation:** The task of automatically selecting the correct sense for a given word
- Input: A word in context
- Output: The correct word sense from a fixed inventory of potential word senses
- The best approach for solving this will depend on your domain and the size of your word and sense sets



Popular Sense-Tagged Corpora

- SemCor: https://www.sketchengine.eu/semcor-annotated-corpus/
- Senseval Corpora: https://web.eecs.umich.edu/~mihalcea/senseval/senseval3/tasks.html
- Certain SemEval corpora: http://alt.qcri.org/semeval2015/task13/
- Sense tag inventories may be domain-specific
  - A word may have many senses in a specialized domain, but fewer senses in the general domain

Word Sense Disambiguation

Given a word, what is its correct sense?

I love my new purple plant!

?

WordNet Search - 3.1

- WordNet home page - Glossary - Help

Word to search for: plant [Search WordNet]

Display Options: (Select option to change) [Change]

Key: "S:" = Show Synset (semantic) relations, "W:" = Show Word (lexical) relations

Display options for sense: (gloss) "an example sentence"

Noun

- S: (n) **plant**, works, industrial plant (buildings for carrying on industrial labor) *"they built a large plant to manufacture automobiles"*
- S: (n) **plant**, flora, plant life ((botany) a living organism lacking the power of locomotion)
- S: (n) **plant** (an actor situated in the audience whose acting is rehearsed but seems spontaneous to the audience)
- S: (n) **plant** (something planted secretly for discovery by another) *"the police used a plant to trick the thieves"; "he claimed that the evidence against him was a plant"*

Verb

- S: (v) **plant**, set (put or set (seeds, seedlings, or plants) into the ground) *"Let's plant flowers in the garden"*
- S: (v) implant, engraft, embed, imbed, **plant** (fix or set securely or deeply) *"He planted a knee in the back of his opponent"; "The dentist implanted a tooth in the gum"*
- S: (v) establish, found, **plant**, constitute, institute (set up or lay the groundwork for) *"establish a new department"*
- S: (v) **plant** (place into a river) *"plant fish"*
- S: (v) **plant** (place something or someone in a certain position in order to secretly observe or deceive) *"Plant a spy in Moscow"; "plant bugs in the dissident's apartment"*
- S: (v) **plant**, implant (put firmly in the mind) *"Plant a thought in the students' minds"*

Task Complexity

- WSD grows more challenging as the number of words being disambiguated grows
- Lexical sample tasks
  - Small pre-selected set of target words
  - Inventory of senses for each word from a lexicon
- All-words tasks
  - Entire large texts
  - Inventory of senses for each word from a lexicon
  - Conceptually similar to POS tagging with a much larger tagset

Semantic Concordances

- All-words tasks are often trained using **semantic concordances**
  - Corpora for which each open-class word in a sentence is labeled with its word sense
- Word senses are then predicted similarly to other sequence tagging tasks

Effective word sense disambiguation is required for many tasks.

- Question answering
  - To which form of “mouse” is the user referring?
- Machine translation
  - Word senses associated with a source language word may not all directly transfer to its target language translation!
- Evaluating NLP models
  - Do word representations accurately reflect relevant word sense similarities?
- Word sense disambiguation tends to be especially challenging in low-resource or highly specialized domains

WSD Baselines

- **Most frequent sense**
  - Given a new word, assign the most frequent sense to it based on counts from a training corpus
  - Often used as a default method when a supervised model has insufficient data to learn the task effectively

WSD Baselines

- **One sense per discourse**
  - Given a new word, if an instance of the same word has already been assigned a sense earlier in the current discourse (by selecting the most frequent sense or applying some other method), assign that same sense
  - Words appearing multiple times in a text or discourse often appear with the same sense (Gale et al., 1992)
    - Gale, W.A., Church, K.W. & Yarowsky, D. A method for disambiguating word senses in a large corpus. Comput Hum 26, 415–439 (1992). https://doi.org/10.1007/BF00136984
  - Works especially well with coarse-grained senses that are unrelated

What are some more sophisticated WSD techniques?

- Lesk algorithm
- Feature-based models
- Contextual embedding models

Lesk Algorithm

- Classic, powerful, **knowledge-based approach**
- Intuition: Given the glosses for all possible senses of a word, the gloss that shares the most words with the immediate context of the target word corresponds to the correct sense

Simplified Lesk Algorithm

```
best_sense ← most frequent sense for word
max_overlap ← 0
context ← set of words in sentence
for each sense in senses of word do:
      signature ← set of words in the gloss and examples of sense
      overlap ← compute_overlap(signature, context)
      if overlap > max_overlap then:
            max_overlap ← overlap
            best_sense ←  sense
return best_sense
```

Case Example: Simplified Lesk Algorithm

The **bank** can guarantee deposits will eventually cover future tuition costs because it invests in adjustable-rate mortgage securities.

$bank^1$	Gloss	A financial institution that accepts deposits and channels the money into lending activities
	Examples	"he cashed a check at the bank," "that bank holds the mortgage on my home"
$bank^2$	Gloss	Sloping land (especially the slope beside a body of water)
	Examples	"they pulled the canoe up on the bank," "he sat on the bank of the river and watched the currents"

Case Example: Simplified Lesk Algorithm

The **bank** can guarantee **deposits** will eventually cover future tuition costs because it invests in adjustable-rate **mortgage** securities.

☺

bank[1]	Gloss	A financial institution that accepts **deposits** and channels the money into lending activities
	Examples	"he cashed a check at the bank," "that bank holds the **mortgage** on my home"
bank[2]	Gloss	Sloping land (especially the slope beside a body of water)
	Examples	"they pulled the canoe up on the bank," "he sat on the bank of the river and watched the currents

Feature-Based WSD

- Choose the best sense based on feature representations and feature-based classification algorithms
- Common features:
  - **Part-of-speech tags** for words before and after the target word
  - **N-grams** before and after the target word
  - **Weighted average of embeddings** for words before and after the target word

Contextual Embedding Models

- Current best-performing models for word sense disambiguation
- Task is framed similar to other neural sequence labeling tasks
- Contextual word embeddings:
  - Word embeddings that differ depending on a word's specific use
  - Word2Vec does *not* produce contextual word embeddings!
  - Contextual embeddings are generally produced using encoder-based approaches
    - **ELMo** was a pioneering implementation of this: https://aclanthology.org/N18-1202.pdf

Contextual Embedding Models

- To train:
  - Extract a contextual embedding for each word in a sense-labeled training set
  - For a given word sense *c*, average the contextual embeddings of all instances of that sense $\mathbf{c_i}$:
    - $\mathbf{v_s} = \frac{1}{n}\sum_i \mathbf{c_i}$
- To test:
  - Compute a contextual embedding $\mathbf{t_i}$ for the target word
  - Select the sense embedding $\mathbf{v_s}$ associated with that target word that has the highest cosine similarity with $\mathbf{t_i}$

What about words that didn't exist in the training data?

One option: Develop simple heuristics for these cases

More sophisticated option: Impute the missing sense embeddings using the WordNet taxonomy and supersenses

Imputing Missing Sense Embeddings

- Find sense embeddings for higher-level nodes in the WordNet taxonomy by averaging the embeddings of their children
  - For each missing sense in WordNet, $\hat{s} \in W$:
    - Let the sense embeddings for other members of its synset be $S_{\hat{s}}$
    - Let the hypernym-specific synset embeddings be $H_{\hat{s}}$
    - Let the lexicographic synset embeddings be $L_{\hat{s}}$
- This produces:
  - An embedding for each synset as the average of its sense embeddings
    - If $|S_{\hat{s}}| > 0$, $\mathbf{v}_{\hat{s}} = \frac{1}{|S_{\hat{s}}|}\sum \mathbf{v}_s$, $\forall \mathbf{v}_s \in S_{\hat{s}}$
  - An embedding for each hypernym as the average of its synset embeddings
    - Else if $|H_{\hat{s}}| > 0$, $\mathbf{v}_{\hat{s}} = \frac{1}{|H_{\hat{s}}|}\sum \mathbf{v}_{syn}$, $\forall \mathbf{v}_{syn} \in H_{\hat{s}}$
  - An embedding for each supersense as the average of the synset embeddings belonging to that lexicographic category
    - Else if $|L_{\hat{s}}| > 0$, $\mathbf{v}_{\hat{s}} = \frac{1}{|L_{\hat{s}}|}\sum \mathbf{v}_{syn}$, $\forall \mathbf{v}_{syn} \in L_{\hat{s}}$

This is guaranteed to produce a representation for every missing sense.

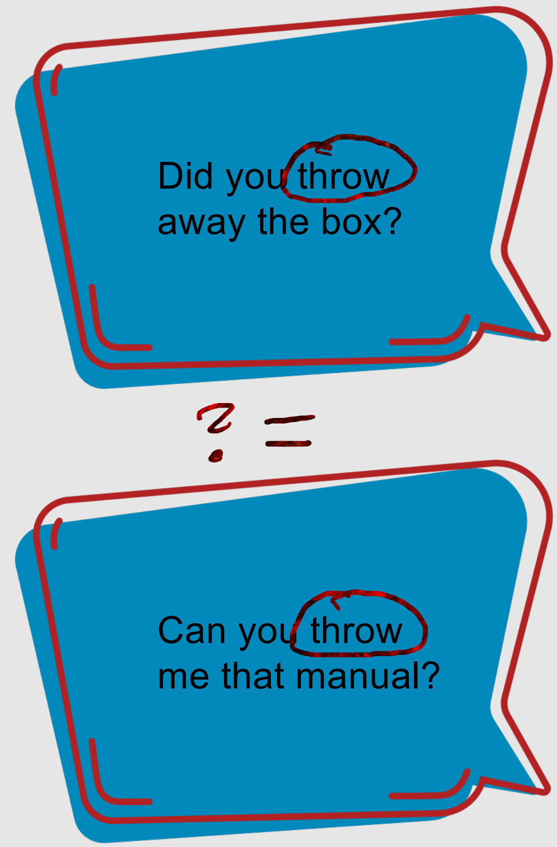
- All supersenses have labeled data in SemCor
- Thus, the algorithm will have some representation for all possible senses by the time it backs off to the lexicographic (supersense) information
- Using information from higher taxonomic levels will produce more coarse-grained sense embeddings

Word Similarity at Different Granularities

- WSD is more fine-grained than earlier word similarity tasks
- Context-free word similarity (how similar is "Chicago" to "Dallas"?)
- This is because word sense disambiguation is a contextualized similarity task
  - Goal is to distinguish the meaning of a word in one context from its meaning in another
- The **word-in-context** task lies between these two extremes

Word-in-Context Evaluation

- Given two sentences with the same target word but different context, decide whether the target words are used:
  - In the same sense, or
  - In different senses
- Can be coarse-grained or fine-grained
  - First-degree sense connections are clustered together
  - Senses belonging to the same supersense are clustered together
- Words are considered as belonging to the same “sense” if they belong to the same cluster



How can we solve word-in-context tasks?

- Simple approach:
  - Compute the contextual embedding for the target word in each of the two sentences
  - Compute the cosine similarity between those embeddings
  - If the cosine similarity is above a threshold, predict that the words are used in the same sense
  - Otherwise, predict that they are used in different senses

Additional Data Acquisition for WSD

- SemCor is often used for WSD, but other data sources can also be leveraged
- One useful resource: Wikipedia
  - Hyperlinks to concepts can be used as sense annotations
  - However, Wikipedia concepts must be mapped to relevant senses for WSD

How can we map Wikipedia concepts to WordNet senses?

- For a given WordNet synset, find the words in the:
  - Synset
  - Gloss
  - Related senses
- For a given Wikipedia concept, find the words in the:
  - Page title
  - Outgoing links
  - Page category
- Select the WordNet sense with the greatest lexical overlap with the Wikipedia concept

Using Lexical Resources to Improve Word Embeddings

- Beyond assisting with WSD, resources like WordNet can be used to improve the quality of learned word embeddings
- This can resolve well-known systemic embedding issues, such as poor estimation of antonymy in static word embeddings
- How can these resources be used?
  - **Retraining**
  - **Retrofitting**

Retraining Word Embeddings

- Modify the embedding's training process to incorporate word sense relations
  - Synonymy
  - Antonymy
  - Hypernymy
- In Word2Vec, this can be done by modifying the static embedding loss function to make use of this information

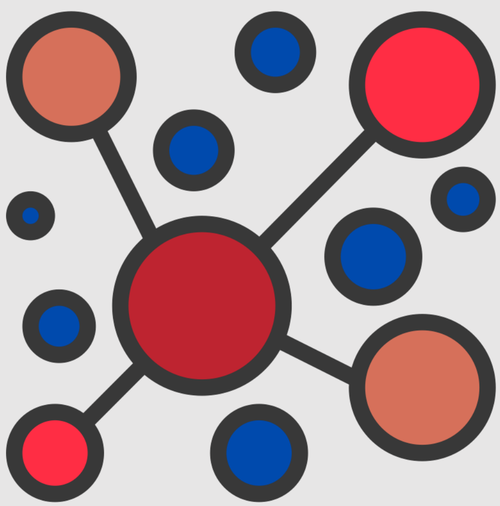
Retrofitting Word Embeddings

- Learn a second mapping based on the lexical resource that shifts the embeddings in such a way that synonyms are pushed closer together and antonyms are pulled further apart
- Also referred to as **counterfitting**

When working with large or unconstrained vocabularies, supervised WSD can be difficult.

- Expensive (and sometimes impractical) to build large corpora labeled with word senses!
- Alternative: Unsupervised word sense disambiguation, or **word sense induction**

Word Sense Induction



- Creates sets of words automatically from a large, unlabeled training set
- Often done using **clustering techniques**
  - Centroid of a cluster represents the **sense vector** corresponding to a sense
  - To induce word senses for new words, algorithms can assign them to the sense vector that is closest to the contextual vector for a given word

If we want to induce senses for each unique word in a training set....

- For each token w_i of word w in a corpus, compute a context vector $\mathbf{c}$
- Use a clustering algorithm to cluster the context vectors $\mathbf{c}$ into a predefined number of clusters, each of which define a sense of w
- Compute the vector centroid, $\mathbf{s_j}$, of each cluster to produce the sense vectors for w

To test....

- Compute a context vector $\mathbf{c}$ for a test token t of word w
- Retrieve all sense vectors $\mathbf{s_j}$ for w
- Assign t to the sense represented by the vector $\mathbf{s_j}$ that is closest to $\mathbf{c}$

Clustering

- Unsupervised machine learning approach that groups data points into "clusters" with similar representations
- Many clustering algorithms exist
  - K-means clustering
  - Density-based clustering
  - Gaussian mixture models
  - And many more!

What clustering method should we use?

- In theory we can use any clustering algorithm for word sense induction
- Common in NLP tasks: **Agglomerative clustering**
  - Each training instance is initially assigned to its own cluster
  - New clusters are formed using a bottom-up process in which the two most similar clusters are successively merged
  - This process continues until the specified number of clusters is reached, or a global cluster quality measure is achieved

Evaluating Unsupervised Word Sense Induction Approaches

- Best approach: Extrinsic evaluation
- If intrinsic evaluation is needed:
  - Measure cluster overlap
  - Map sense clusters to predefined senses
  - Devise other approaches that map automatically-derived sense classes to an established gold standard for performance comparison
- There is no standardized evaluation metric (yet!) for this task

Summary: Word Senses and WordNet

- Word **senses** define a word's meaning in context
- Many words are **polysemous**
- Word senses can be related to one another in many ways, such as through **synonymy**, **antonymy**, **meronymy**, and **hypernymy**
- **WordNet** is a large lexical database with word sense information for nouns, verbs, adjectives, and adverbs
- **Word sense disambiguation** is the task of determining the correct sense for a word, given its context
- WSD can be performed in a variety of ways, including with contextual embedding approaches, feature-based algorithms, the **Lesk algorithm**, or a most frequent sense baseline
- Word senses can also be **induced** using unsupervised clustering methods

This Week's Topics

Tuesday

- Word Senses
- WordNet
- Word Sense Disambiguation

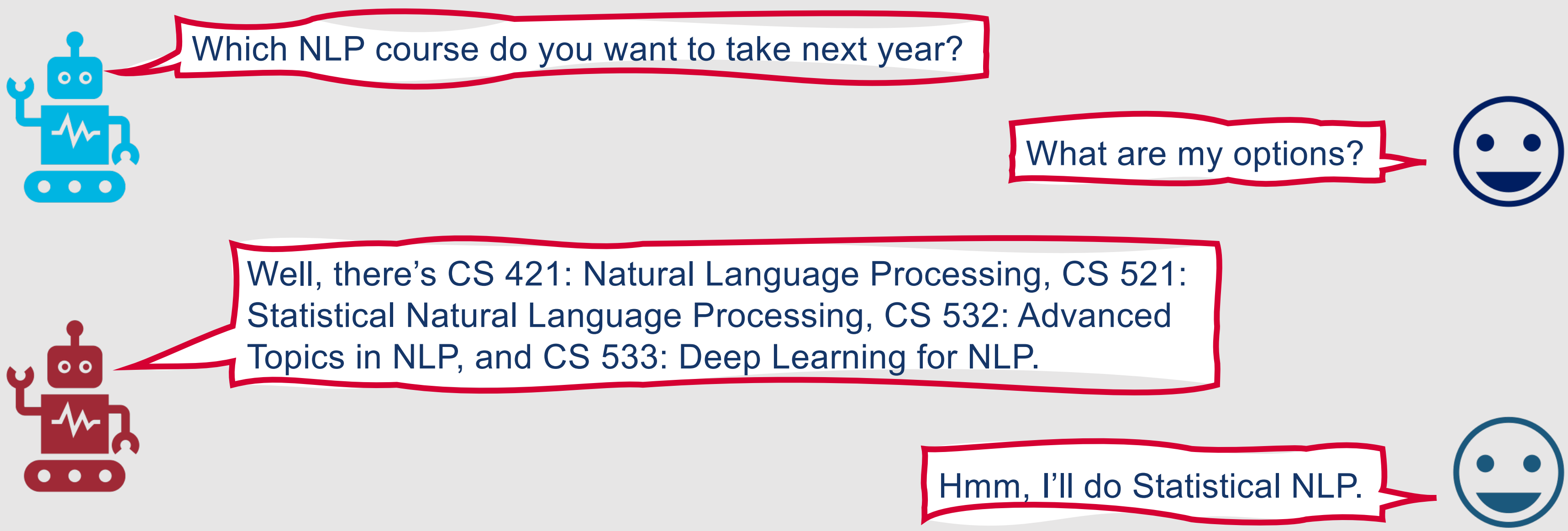
Thursday

- Coreference Resolution
- Referring Expressions
- Coreference Resolution Approaches
- Evaluating Coreference Resolution

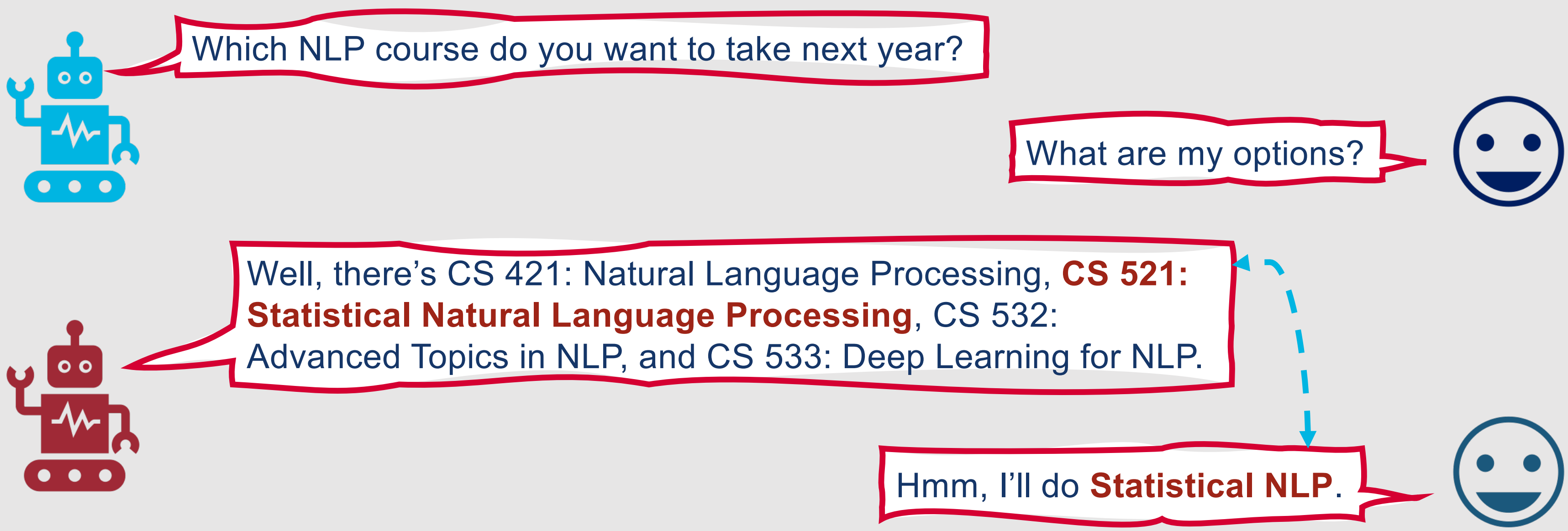
What is coreference resolution?

The process of automatically identifying expressions that refer to the same entity

Coreference resolution is essential to creating high-performing NLP systems.



Coreference resolution is essential to creating high-performing NLP systems.



Both humans and NLP systems interpret language with respect to a discourse model.

- **Discourse model:** Mental model that is built incrementally, containing representations of entities, their properties, and the relations between them
- **Referent:** The discourse entity itself
  - (CS 521: Statistical Natural Language Processing)
- **Referring expression:** The linguistic expression referring to a referent
  - "CS 521"
  - "CS 521: Statistical Natural Language Processing"
  - "521"
  - "Statistical NLP"
- Two or more referring expressions that refer to the same discourse entity are said to **corefer**

Anaphora

- **Anaphora:** Referring to an entity that has already been introduced in the discourse
  - First mention is the **antecedent**
  - Subsequent mentions are **anaphors**
  - Entities with only a single mention are **singletons**

The University of Illinois at Chicago is an excellent place to study natural language processing. UIC has many faculty currently working in the area, including but not limited to Natalie Parde, Barbara Di Eugenio, Cornelia Caragea, Bing Liu, and Philip Yu. The school is located in bustling downtown Chicago, and as a bonus it will be opening a snazzy new CS building in 2025.

Anaphora

- **Anaphora:** Referring to an entity that has already been introduced in the discourse
  - First mention is the **antecedent**
  - Subsequent mentions are **anaphors**
  - Entities with only a single mention are **singletons**

The **University of Illinois at Chicago** is an excellent place to study natural language processing. UIC has many faculty currently working in the area, including but not limited to Natalie Parde, Barbara Di Eugenio, Cornelia Caragea, Bing Liu, and Philip Yu. The school is located in bustling downtown Chicago, and as a bonus it will be opening a snazzy new CS building in 2025.

Anaphora

- **Anaphora:** Referring to an entity that has already been introduced in the discourse
  - First mention is the **antecedent**
  - Subsequent mentions are **anaphors**
  - Entities with only a single mention are **singletons**

The **University of Illinois at Chicago** is an excellent place to study natural language processing. **UIC** has many faculty currently working in the area, including but not limited to Natalie Parde, Barbara Di Eugenio, Cornelia Caragea, Bing Liu, and Philip Yu. **The school** is located in bustling downtown Chicago, and as a bonus **it** will be opening a snazzy new CS building in 2025.

Anaphora

- **Anaphora:** Referring to an entity that has already been introduced in the discourse
  - First mention is the **antecedent**
  - Subsequent mentions are **anaphors**
  - Entities with only a single mention are **singletons**

The **University of Illinois at Chicago** is an excellent place to study natural language processing. **UIC** has many faculty currently working in the area, including but not limited to **Natalie Parde**, Barbara Di Eugenio, Cornelia Caragea, Bing Liu, and Philip Yu. **The school** is located in bustling downtown Chicago, and as a bonus **it** will be opening a snazzy new CS building in 2025.

Coreference Chains

A set of coreferring expressions is often called a **coreference chain**

The **University of Illinois at Chicago** is an excellent place to study natural language processing. **UIC** has many faculty currently working in the area, including but not limited to **Natalie Parde**, Barbara Di Eugenio, Cornelia Caragea, Bing Liu, and Philip Yu. **The school** is located in bustling downtown Chicago, and as a bonus **it** will be opening a snazzy new CS building in 2025.

{“University of Illinois at Chicago”, “UIC”, “The school”, “it”}

{“Natalie Parde”}

Two Key Tasks

- **Coreference resolution** thus generally comprises two key tasks:
  - Identify **referring expressions** (mentions of entities)
  - Cluster them into **coreference chains**
- We can also perform **entity linking** to map coreference chains to real-world entities
  - {"University of Illinois at Chicago", "UIC", "The school", "it"} → https://en.wikipedia.org/wiki/University_of_Illinois_at_Chicago

This Week's Topics

Tuesday

- Word Senses
- WordNet
- Word Sense Disambiguation

Thursday

- Coreference Resolution
- Referring Expressions
- Coreference Resolution Approaches
- Evaluating Coreference Resolution

Linguistic Background

- Referring expressions can occur in several forms:
  - **Indefinite noun phrases**
  - **Definite noun phrases**
  - **Pronouns**
  - **Proper nouns (names)**
- These can be used to **evoke** and **access** entities in the discourse model in a variety of ways

Indefinite Noun Phrases

- Usually marked with the determiner *a* or *an*
- Can also be marked with other indefinite terms
  - E.g., *some*
- Generally introduce **new entities** to the discourse

The blue line was experiencing delays so I took **an** Uber.

Definite Noun Phrases

- Usually marked with *the*
- Generally refer to entities that have already been introduced to the discourse
- May refer to entities that haven't been introduced to the discourse, but are identifiable to the receiver due to:
  - World knowledge
  - Implications from the discourse structure

The blue line was experiencing delays so I took **an** Uber. Unfortunately, so did everyone else …**the** Uber got stuck in a traffic jam.

Have you checked out **the** Andy Warhol exhibit?

Make sure to order **the** tiramisu!

Pronouns

- Generally refer to entities that have already been introduced to the discourse and are easily identifiable

> The blue line was experiencing delays so I took **an** Uber. Unfortunately, so did everyone else …**the** Uber got stuck in a traffic jam. **It** ended up reaching UIC later than the original train I'd been hoping to catch.

Proper Nouns (Names)

- Can be used either to introduce new entities to the discourse, or to refer to those that already exist

Chicago is one of the largest cities in the United States. **Chicago** is known for its architecture, its thriving arts and music scene, its hot dogs and deep dish pizza, and---of course---its winter weather.

Information Status

- Referring expressions can also be categorized by their **information status**
  - The way they introduce **new information** or access **old information**
- Three main groups:
  - New noun phrases
  - Old noun phrases
  - Inferables

New Noun Phrases

- **Brand new NPs:** Introduce entities that are both **new to the discourse** and **new to the listener**
  - E.g., *an Uber*
- **Unused NPs:** Introduce entities that are **new to the discourse** but **not to the listener**
  - E.g., *Chicago*

Old Noun Phrases

- Introduce entities that already exist in the discourse model (and are thus **not new to the discourse nor to the listener**)
  - E.g., *she*

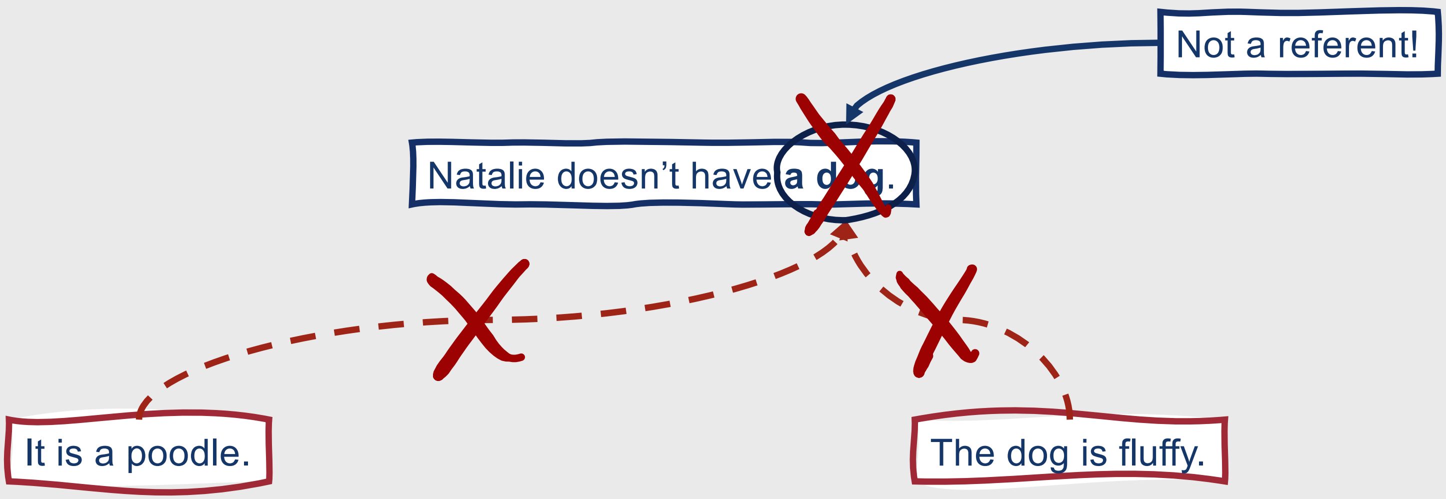
Inferables

- Introduce entities that are **new to the discourse** and **new to the listener** *but* the hearer can infer their existence by reasoning about other entities already introduced
  - E.g., I got in my Uber and told *the driver* to take us to UIC as fast as she could.

Generally, the form of a referring expression gives strong clues about its information status.

- **Very salient** (easily accessible) entities can be referred to using **less linguistic material**
  - E.g., pronouns
- **Less-salient** entities (e.g., those that are discourse-new and hearer-new) require **more linguistic material**
  - E.g., full names

Note: Not all noun phrases are referring expressions!



Structures Easily Confused with Referring Expressions

Appositives	Noun phrases that describe other noun phrases	Natalie Parde, *Associate Professor of Computer Science*, teaches CS 421.
Predicative and Prenominal Noun Phrases	Noun phrases that describe characteristics of other noun phrases	Natalie Parde is an *Associate Professor*.
Expletives	Non-referential pronouns	Natalie thought *it* was cool that so many students at UIC were interested in NLP.
Generics	Pronouns that refer to classes of nouns in general, rather than specific instances of those nouns	In Chicago, *you* get to experience all four seasons - summer, early winter, winter, and late winter.

So far, we've focused on linguistic properties of referring expressions….

- What linguistic properties should we look for when determining coreference relations?
  - Number agreement
  - Person agreement
  - Gender/noun class agreement
  - Binding theory constraints
  - Recency
  - Grammatical role
  - Verb semantics
  - Selectional restrictions

Number Agreement

- In general, antecedents and their anaphors should agree in number
  - Singular with singular
  - Plural with plural
- A few exceptions:
  - Some semantically plural entities (e.g., companies) can be referred to using either singular or plural pronouns
  - “They” can be used as a singular pronoun

Person Agreement

In general, antecedents and their anaphors should agree in person

First person with first person

- I, my, me

Third person with third person

- They, their, them

An exception:

Text containing quotations

- "**I** spent twelve hours making those slides," **she** pointed out.

Gender/Noun Class Agreement

- In general, antecedents and their anaphors should agree in grammatical gender
  - He with his
  - She with hers
  - They with theirs
- This is an even bigger deal in languages for which all nouns have grammatical gender
  - La casa 🏠
  - El banco 🏦

Binding Theory Constraints and Recency

- **Binding Theory Constraints:** Antecedents and their anaphors should adhere to the syntactic constraints placed upon them
  - Reflexive pronouns (e.g., herself) corefer with the subject of the most immediate clause that contains them
    - **Natalie** told **herself** that she wouldn't be nearly as busy next week.
- **Recency:** Antecedents introduced recently tend to be more salient than those introduced earlier
  - Pronouns are likelier to be anaphors for the most recent plausible antecedent
    - Natalie went to a **faculty meeting**. Shahla went to a **student government meeting**. **It** was mainly about new policy changes that had recently been approved.

Grammatical Role

- Antecedents in some grammatical roles are more salient than others
  - Subject position > object position
    - **Natalie** went to the Eiffel Tower with **Shahla**. **She** took a selfie.

Verb Semantics

- Salience may be influenced by the types of verbs to which antecedents and anaphors are arguments
  - **Natalie** congratulated **Shahla**. **Her** paper had just been accepted.
  - **Natalie** bragged to **Shahla**. **Her** paper had just been accepted.

Selectional Restrictions

- Finally, salience may also be influenced by other semantic knowledge about the verbs to which antecedents and anaphors are arguments
  - Natalie pulled her **suitcase** out of the **Uber**. **It** sped off into the sunset.

This Week's Topics

Tuesday

- Word Senses
- WordNet
- Word Sense Disambiguation

Thursday

- Coreference Resolution
- Referring Expressions
- Coreference Resolution Approaches
- Evaluating Coreference Resolution

Coreference Tasks

- We can formalize the task of coreference resolution as follows:
  - **Given a text *T*, find all entities and the coreference links between them**
- This requires a few subtasks:
  - **Detect mentions**
    - Likely to be mentions:
      - Pronouns
      - Definite noun phrases
      - Indefinite noun phrases
      - Names
    - Exclude non-referential pronouns or noun phrases
  - **Link those mentions into clusters**

What counts as a mention?

- Depends on the task specifications and dataset
- Some coreference datasets do not include singletons as mentions
  - Makes the task easier
    - Singletons are often difficult to distinguish from non-referential noun phrases, and constitute a majority of mentions

Sample Coreference Task

The University of Illinois at Chicago is an excellent place to study natural language processing. UIC has many faculty currently working in NLP, including but not limited to Natalie Parde, Barbara Di Eugenio, Cornelia Caragea, Bing Liu, and Philip Yu. The school is located in bustling downtown Chicago, and as a bonus it will be opening a snazzy new CS building in 2025.

Sample Coreference Task

The **University of Illinois at Chicago** is an excellent place to study **natural language processing**. **UIC** has many **faculty** currently working in **NLP**, including but not limited to **Natalie Parde**, **Barbara Di Eugenio**, **Cornelia Caragea**, **Bing Liu**, and **Philip Yu**. **The school** is located in bustling downtown **Chicago**, and as a bonus **it** will be opening a snazzy new **CS building** in 2025.

Detect mentions

Sample Coreference Task

The **University of Illinois at Chicago** is an excellent place to study **natural language processing**. **UIC** has many **faculty** currently working in **NLP**, including but not limited to **Natalie Parde**, **Barbara Di Eugenio**, **Cornelia Caragea**, **Bing Liu**, and **Philip Yu**. **The school** is located in bustling downtown **Chicago**, and as a bonus **it** will be opening a snazzy new **CS building** in 2025.

Detect mentions

Cluster mentions

Sample Coreference Task

The **University of Illinois at Chicago** is an excellent place to study **natural language processing**. **UIC** has many **faculty** currently working in **NLP**, including but not limited to **Natalie Parde**, **Barbara Di Eugenio**, **Cornelia Caragea**, **Bing Liu**, and **Philip Yu**. **The school** is located in bustling downtown **Chicago**, and as a bonus **it** will be opening a snazzy new **CS building** in 2025.

Detect mentions

Cluster mentions

Coreference Chains:
- {University of Illinois at Chicago, UIC, The school, it}
- {natural language processing, NLP}
- {faculty}
- {Natalie Parde}
- {Barbara Di Eugenio}
- {Cornelia Caragea}
- {Bing Liu}
- {Philip Yu}
- {Chicago}
- {CS building}

Popular Coreference Datasets

OntoNotes

- Chinese, English, and Arabic texts in a variety of domains (e.g., news, magazine articles, speech data, etc.)
- No singletons
- https://catalog.ldc.upenn.edu/LDC2013T19

ISNotes

- Adds information status to OntoNotes
- https://github.com/nlpAThits/ISNotes1.0

ARRAU

- English texts in a variety of domains
- Includes singletons
- https://catalog.ldc.upenn.edu/LDC2013T22

Moving on to the finer details….

- Mention detection: The process of finding spans of text that constitute a referring expression (mention)
  - It’s common to be very liberal in predicting mentions, and rely on downstream filtering to prune bad predictions

The **University of Illinois at Chicago** is an excellent ~~**place**~~ to study **natural language processing**. **UIC** has many **faculty** currently working in **NLP**, including but not limited to **Natalie Parde**, **Barbara Di Eugenio**, **Cornelia Caragea**, **Bing Liu**, and **Philip Yu**. **The school** is located in bustling downtown **Chicago**, and as a ~~**bonus**~~ **it** will be opening a snazzy new **CS building** in 2025.

Mention Detection

- How is filtering performed?
  - Sometimes, rules
  - More often, **classifiers**
- Classifiers for mention filtering often make use of features characterizing the words, their relationship, and their position in the surrounding text

1. Take all predicted mentions
2. Remove numeric quantities, mentions embedded in larger mentions, and stop words
3. Remove non-referential "it" based on regular expression patterns

Mention filtering can be a tricky balance!

- Filter too many → recall suffers
- Filter too few → precision suffers
- Some recent approaches also perform mention detection, filtering, and entity clustering jointly in an end-to-end model

Architectures for Coreference Algorithms

Several different ways to tackle the problem:

- **Entity-based classification**
  - Make decisions based on a given entity in the discourse model as a whole
- **Mention-based classification**
  - Make decisions locally for each mention
- **Ranking models**
  - Compare potential antecedents with one another (can be combined with either entity-based or mention-based approaches)

The Mention-Pair Architecture

Simple premise:

Given:

- Pair of mentions (candidate anaphor and candidate antecedent)

Decide:

- Whether or not they corefer

How does this work?

Compute coreference probabilities for every plausible pair of mentions

Goal: High probability for actual coreferring pairs, and low probability for other pairs

The Mention-Pair Architecture

The **University of Illinois at Chicago** is an excellent place to study **natural language processing**. **UIC** has many **faculty** currently working in **NLP**, including but not limited to **Natalie Parde**, **Barbara Di Eugenio**, **Cornelia Caragea**, **Bing Liu**, and **Philip Yu**. **The school** is located in bustling downtown **Chicago**, and as **a bonus** **it** will be opening a snazzy new **CS building** in 2025.

The Mention-Pair Architecture

The **University of Illinois at Chicago** is an excellent place to study **natural language processing**. **UIC** has many **faculty** currently working in **NLP**, including but not limited to **Natalie Parde**, **Barbara Di Eugenio**, **Cornelia Caragea**, **Bing Liu**, and **Philip Yu**. **The school** is located in bustling downtown **Chicago**, and as a **bonus** **it** will be opening a snazzy new **CS building** in 2025.

The Mention-Pair Architecture

The **University of Illinois at Chicago** is an excellent place to study **natural language processing**. **UIC** has many **faculty** currently working in **NLP**, including but not limited to **Natalie Parde**, **Barbara Di Eugenio**, **Cornelia Caragea**, **Bing Liu**, and **Philip Yu**. **The school** is located in bustling downtown **Chicago**, and as a **bonus** **it** will be opening a snazzy new **CS building** in 2025.

The Mention-Pair Architecture

The **University of Illinois at Chicago** is an excellent place to study **natural language processing**. **UIC** has many **faculty** currently working in **NLP**, including but not limited to **Natalie Parde**, **Barbara Di Eugenio**, **Cornelia Caragea**, **Bing Liu**, and **Philip Yu**. **The school** is located in bustling downtown **Chicago**, and as a **bonus** **it** will be opening a snazzy new **CS building** in 2025.

How do we learn these probabilities?

- Select training samples
  - For every one positive instance (m_i, m_j) where m_j is the closest antecedent to m_i,
  - Extract numerous negative instances (m_i, m_k) for each m_k between m_j and m_i
- Extract features
  - Manually engineered features, and/or
  - Implicitly learned representations
- Train classification model

How do we make predictions?

- Apply the trained classifier to each test instance in a clustering step
  - **Closest-first clustering**
    - For mention *i*, classifier is run backwards through prior *i*-1 mentions
    - First prior mention (candidate antecedent) with probability > 0.5 is selected and linked to *i*
  - **Best-first clustering**
    - Classifier is run on all possible *i*-1 antecedents (all mentions prior to mention *i*)
    - Mention with highest probability is selected as the antecedent for *i*

Mention-Pair Architecture

- Advantage:
  - **Simplest** coreference resolution architecture
- Disadvantage:
  - **Doesn't directly compare candidate antecedents** with one another
  - **Considers only mentions**, not overall entities

How can we address these limitations?

- One option: The **Mention-Rank Architecture**
  - Directly compares antecedents with one another
  - Selects the highest-scoring antecedent for each anaphor
- How does this work?
    - Use heuristics to determine the best antecedent for an anaphor (e.g., closest = best)
    - Or, train models to predict scores for candidate antecedents for given anaphors

Another Option: Entity-based Models

- Considers discourse entities, rather than individual mentions
- How does this work?
  - Have the model make decisions over clusters of mentions, where each cluster corresponds to an entity
  - Can be implemented using feature-based or neural classifiers

Feature-based Classification Models

- Common feature types:
  - Features of the candidate anaphor
  - Features of the candidate antecedent
  - Features of the relationship between the pair
- For entity-based models, this can also include:
  - Features of all mentions of the candidate antecedent's entity cluster
  - Features of the relation between the candidate anaphor and the mentions of the candidate antecedent in the entity cluster

Helpful Features for Coreference Resolution

- First word
- Head word
- Gender
- Named entity type
- Length
- Grammatical role
- Document genre
- ...and many more!

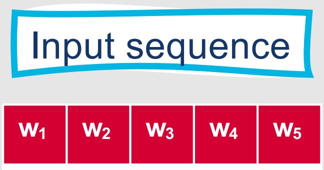
Neural Classification Models

- Generally end-to-end without a separate mention detection step
  - Instead, consider every possible text span of length < *k* as a possible mention
- Same overall goal as usual:
  - Assign to each span *i* an antecedent y_i ranging over the values $Y(i) = \{1, \ldots, i-1, \varepsilon\}$

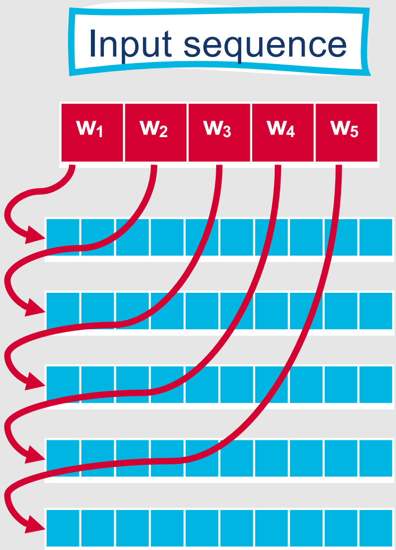
What goes on behind the scenes?

- For each pair of spans *i* and *j*, the system assigns a score $s(i,j)$ for the coreference link between the two
  - $s(i,j) = m(i) + m(j) + c(i,j)$
    - $m(i)$: Whether span *i* is a mention
    - $m(j)$: Whether span *j* is a mention
    - $c(i,j)$: Whether *j* is the antecedent of *i*
- The functions $m(\cdot)$ and $c(\cdot,\cdot)$ are computed using neural models:
  - $m(i) = w_m \cdot NN_m(g_i)$
  - $c(i,j) = w_c \cdot NN_c([g_i, g_j, g_i \circ g_j, \phi(i,j)])$
    - For example, where g_i is a vector representation of span *i* and $\phi(i,j)$ encodes manually-defined characteristics of the relationship between *i* and *j*
    - Exact definition of $c(i,j)$ may differ across models

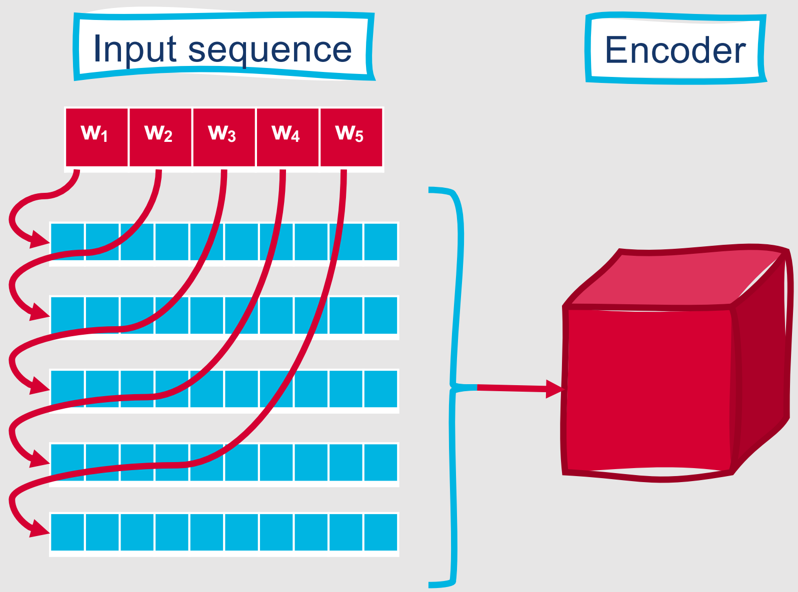
Altogether, a neural coreference resolution model might look like the following….



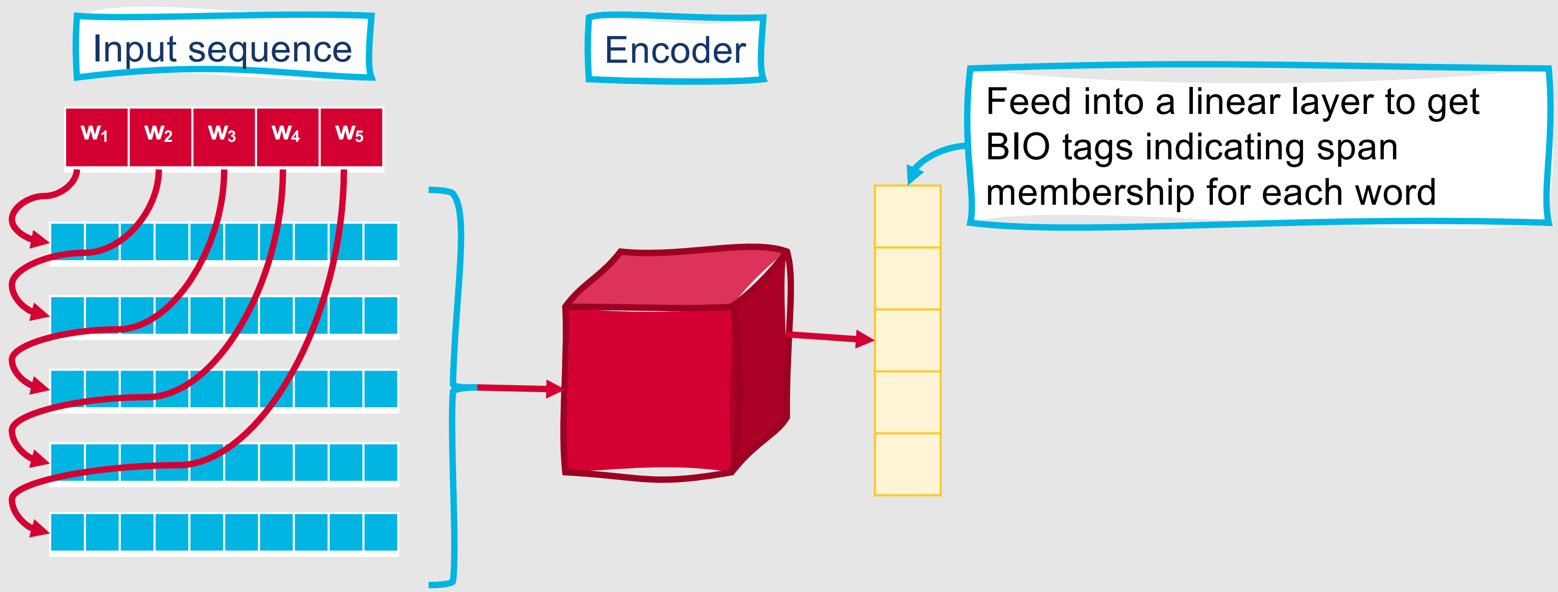
Altogether, a neural coreference resolution model might look like the following….



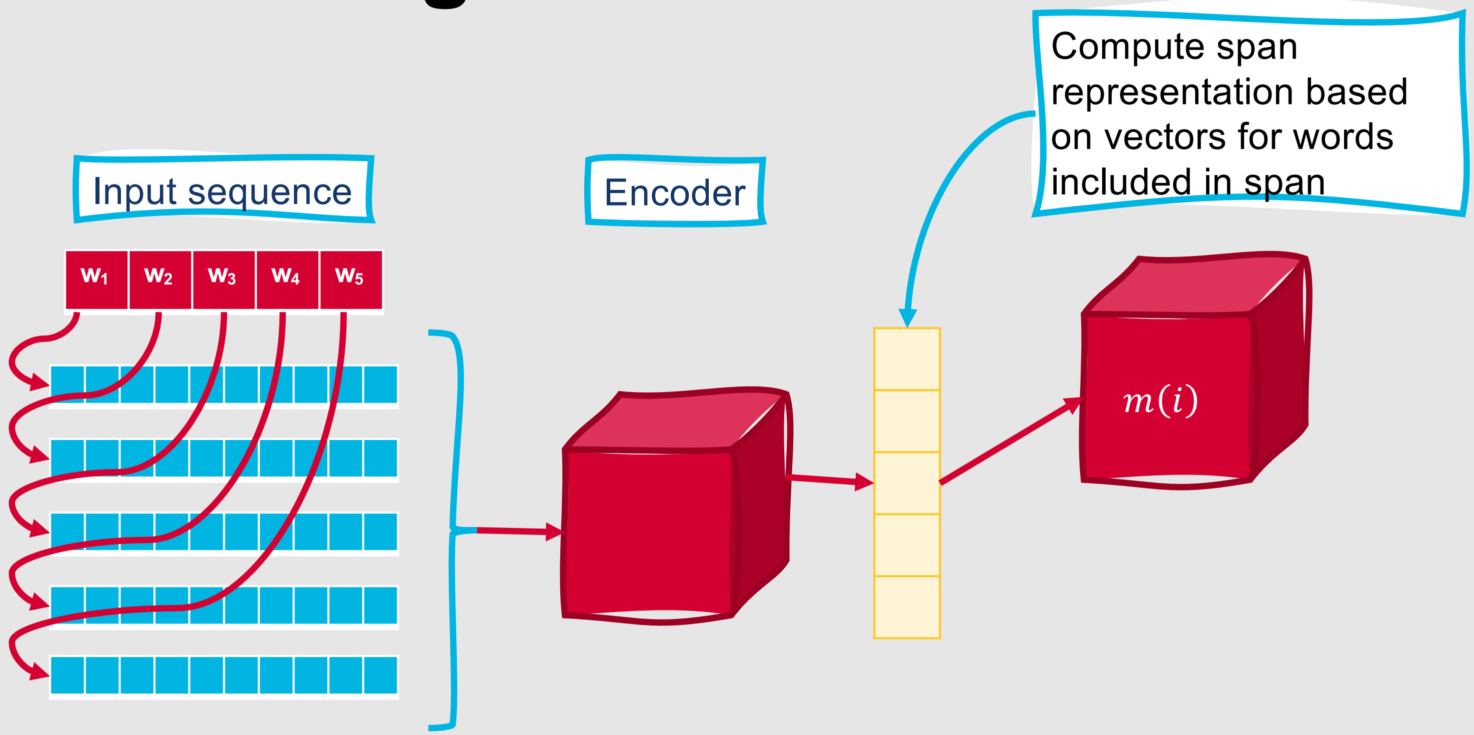
Altogether, a neural coreference resolution model might look like the following….



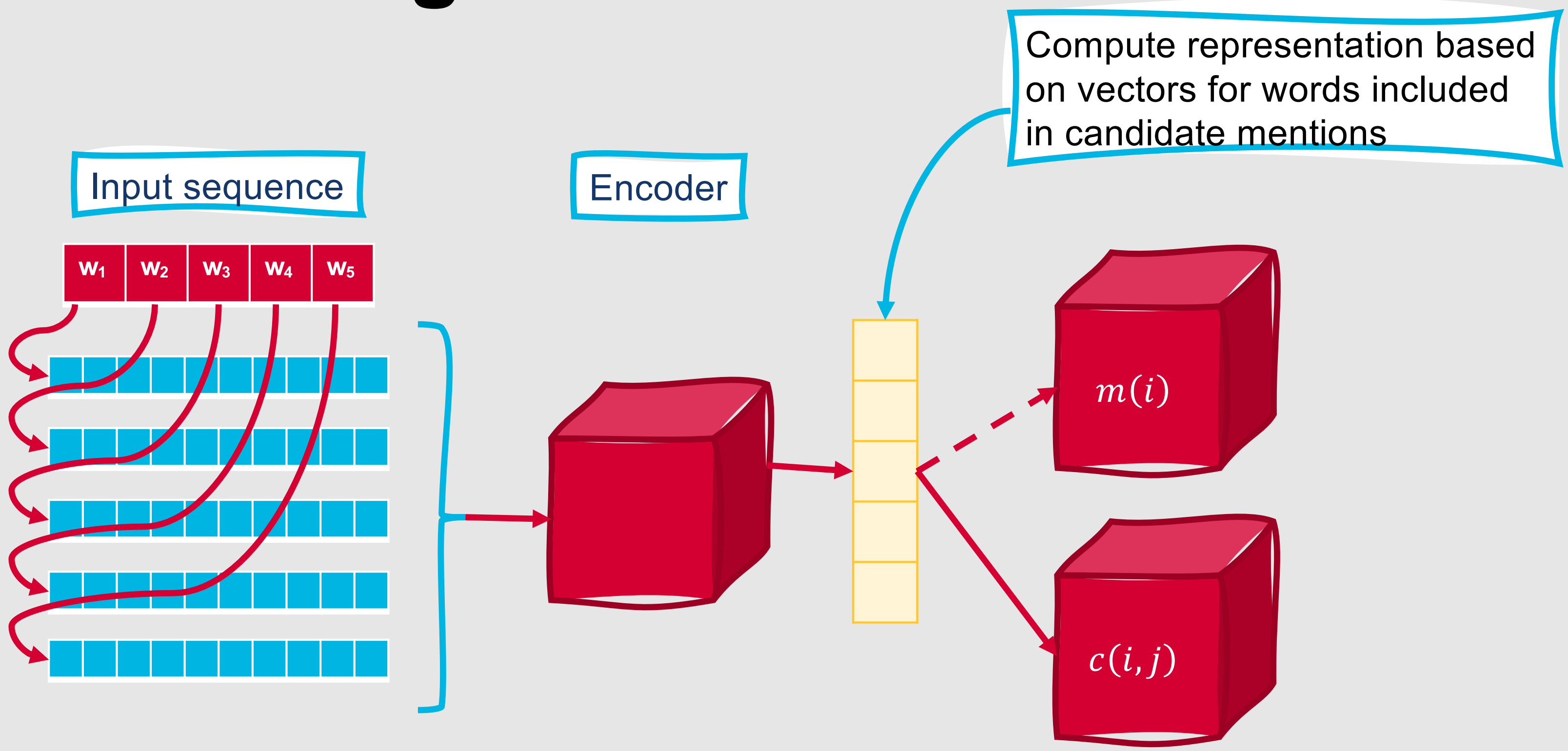
Altogether, a neural coreference resolution model might look like the following….



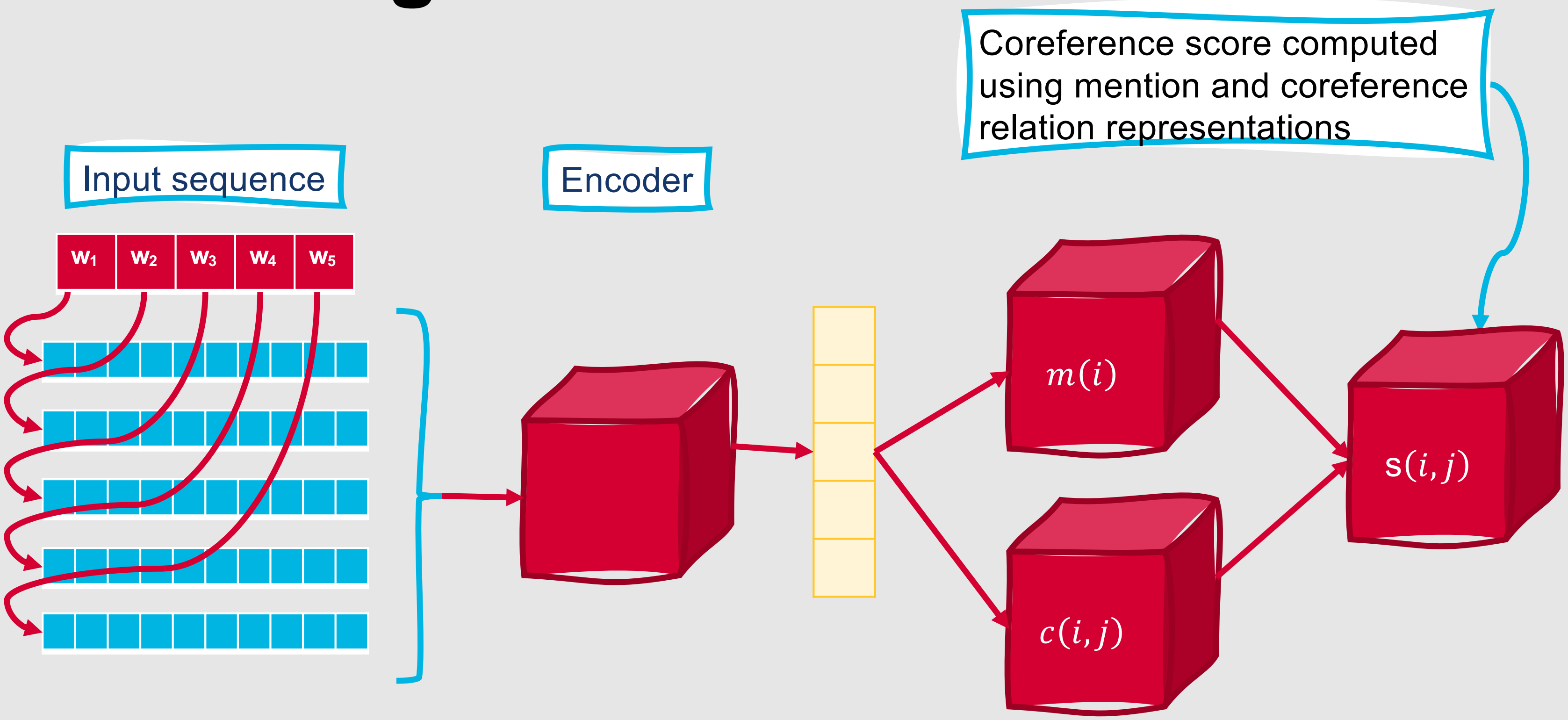
Altogether, a neural coreference resolution model might look like the following….



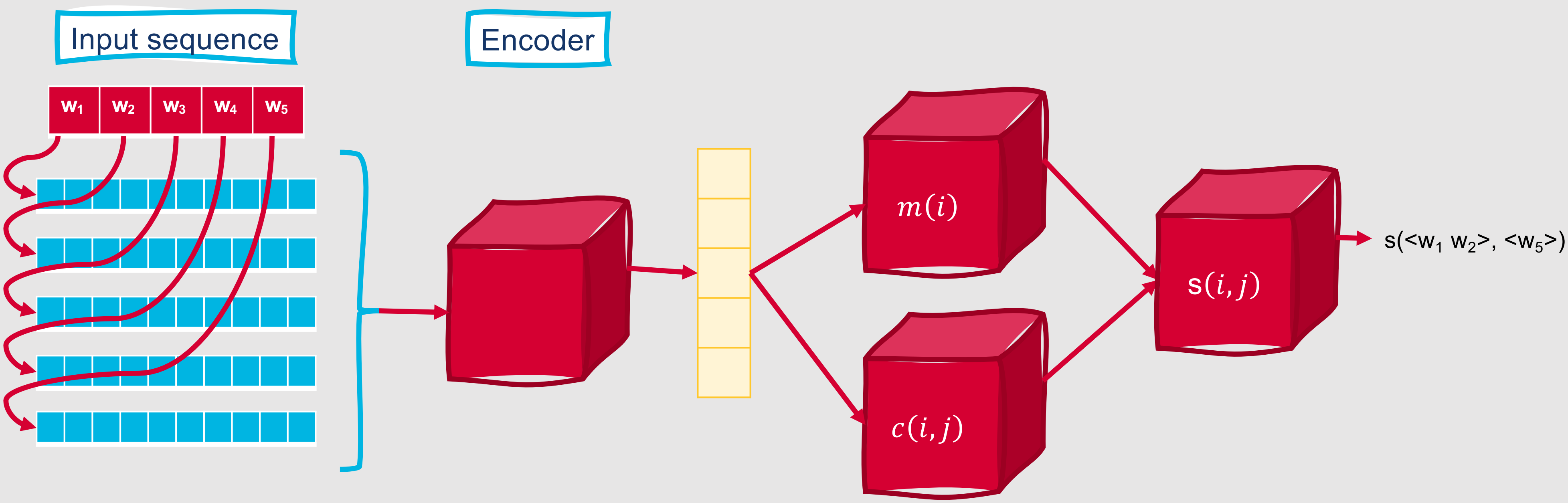
Altogether, a neural coreference resolution model might look like the following….



Altogether, a neural coreference resolution model might look like the following….



Altogether, a neural coreference resolution model might look like the following….



This Week's Topics

Tuesday

- Word Senses
- WordNet
- Word Sense Disambiguation

Thursday

- Coreference Resolution
- Referring Expressions
- Coreference Resolution Approaches
- Evaluating Coreference Resolution

How do we evaluate coreference resolution models?

- Compare hypothesis coreference chains or clusters with a gold standard
- Compute precision and recall

How do we compute precision and recall?

- Several approaches:
  - **Link-based:** MUC F-measure
  - **Mention-based:** B^3

MUC F-Measure

- Message Understanding Conference (MUC)
- True positives = Common coreference links (anaphor-antecedent pairs) between hypotheses and gold standard
- Precision = # Common links / # Links in hypotheses
- Recall = # Common links / # Links in gold standard
- A couple downsides to this approach:
  - Biased towards systems that produce large coreference chains
  - Ignores singletons (no links to count)

B^3

- Mention-based
- True positives for a given mention, *i* = # Common mentions in hypothesis and gold standard coreference chain including *i*
- Precision for a given mention, *i* = TP / # Mentions in hypothesis coreference chain including *i*
- Recall for a given mention, *i* = TP / # Mentions in gold standard coreference chain including *i*
- Total precision and recall are the weighted sums of precision and recall across all mentions

So …where are we now?

- Still plenty of room for growth in coreference resolution!
- Recently, lots of interest in **Winograd Schema** problems
  - Coreference resolution problems that are:
    - Easy for humans to solve
    - Particularly challenging for computers to solve, due to their reliance on world knowledge and commonsense reasoning

Winograd Schema Problems

- Winograd Schema problems are characterized by the following:
  - There are two statements that differ by only one word or phrase
    - That word or phrase influences the human-preferred answer
  - There are two entities that remain the same across statements
  - A pronoun preferentially refers to one of the entities, but could grammatically also refer to the other
  - A question asks to which entity the pronoun refers

Example Winograd Schema Problem

Nikolaos lost the race to Giuseppe because he was **slower**.

Who was “he”?

Nikolaos

Example Winograd Schema Problem

Nikolaos lost the race to Giuseppe because he was **slower**.

Who was “he”?

Nikolaos

Nikolaos lost the race to Giuseppe because he was **faster**.

Who was “he”?

Giuseppe

Example Winograd Schema Problem

Nikolaos lost the race to Giuseppe because he was **slower**.

Who was “he”?

Nikolaos

Nikolaos lost the race to Giuseppe because he was **faster**.

Who was “he”?

Giuseppe

Best way to solve Winograd Schema problems computationally?

- Currently, a mix of language modeling and external knowledge bases

Gender Bias in Coreference Resolution

- As with language modeling, coreference resolution systems can exhibit harmful gender biases
- How can we avoid these issues?
  - One solution: Increase sample size for underrepresented genders
    - Artificially: Generate gender-swapped versions of existing training corpora
    - Manually: Collect new, gender-balanced corpora
  - Other solutions?
    - Still very much an active research question!

Summary: Coreference Resolution

- **Coreference resolution** is the process of automatically identifying expressions that refer to the same entity
- This involves two tasks:
  - Identifying **referring expressions**
  - Clustering them into **coreference chains**
- Architectures for coreference resolution systems may be **mention-based** or **entity-based**, and may or may not compare potential **antecedents** with one another
- Models for coreference resolution may learn based on **manually defined features**, **neural features**, or a combination of the two
- Computing precision and recall for coreference resolution systems may be done using either **link-based** or **mention-based** methods
- **Winograd Schema** problems are particularly challenging coreference resolution tasks that rely on world knowledge and commonsense reasoning
- Care should be taken to avoid introducing harmful **gender biases** into coreference resolution systems