

# LING/C SC/PSYC 438/538

Lecture 14

Sandiway Fong

# Today's Topics

- Predicate-Argument Structure *contd.*
  - Framenet: *verbs and their semantic roles*
  - Stanford CoreNLP: *dependency parses*
- Homework 9

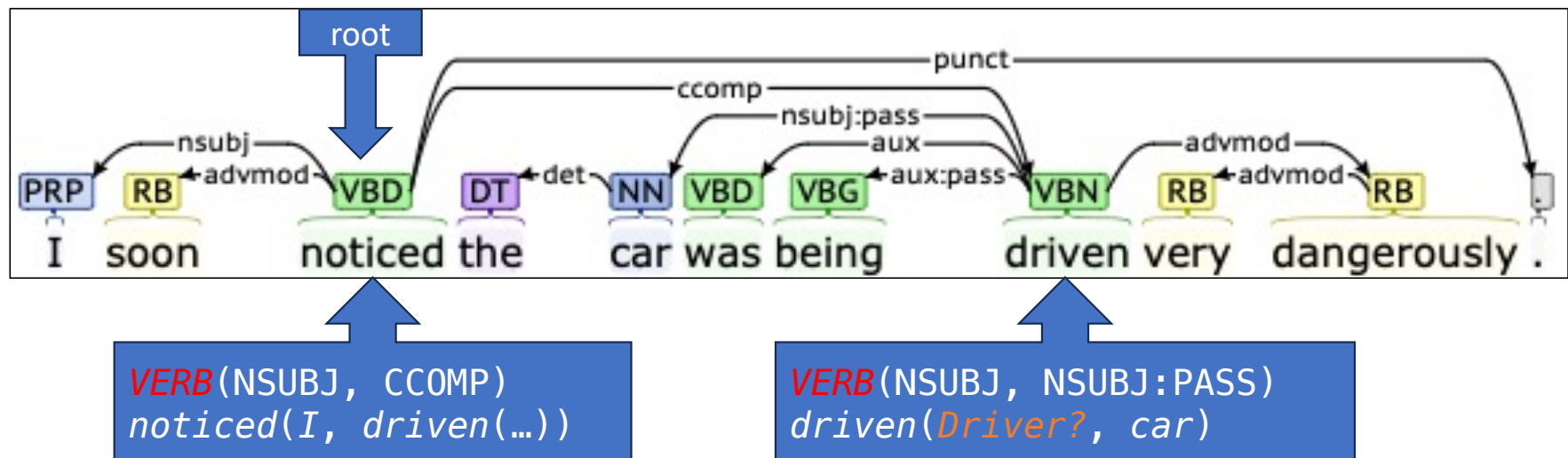
# Background: CoreNLP

- Examples (from Framenet):

1. [Cognizer I] soon **NOTICED** [Phenomenon the car was being driven very dangerously] .

- Verb (VBD) = noticed, Cognizer = nsubj, Phenomenon = ccomp

- Verb(VBN) = driven, Vehicle = nsubj:pass



# Background: Framenet: *drive*

## Operate\_vehicle

[Lexical Unit I](#)

### Definition:

The words in this frame describe motion involving a **Vehicle** and someone who controls it, the **Driver**. Some words normally allow the **Vehicle** to be expressed as a separate constituent.

**Tim** **DROVE** **his car** **all the way across North America**.

**Tom** **PADDLED** **my canoe** **across the Canadian border**.

Other words in this domain are based on the names of vehicles, and do not normally allow the **Vehicle** to be expressed as a separate constituent.

**The group** **BIKED** **all the way across the country**.

However, a separate **Vehicle** constituent can occur if it adds information not included in the verb.

**Tim** **BIKED** **across the country** **on an old 10-speed**.

# Background: Framenet: *drive*

## Core:

### Area [Area]

This frame element is used for expressions which describe a general area in which motion takes place when the motion is understood to be irregular and not to consist of a single linear path. Locative setting adjuncts of motion expressions may also be assigned this frame element.

The bikers **RODE** **all over the place**.

### Driver [Driv]

Semantic Type: Sentient

This is the being, typically human, that controls the **Vehicle** as it moves.

**Kim** **DROVE** my old car cross-country.

### Goal [Goal]

Semantic Type: Goal

Excludes: Area

Any expression which tells where the moving object(s) ends up as a result of the motion expresses the frame element **Goal**. Some particles imply the existence of a **Goal** which is understood in the context of utterance.

Kim **DROVE** **into the parking lot**.

Kim **DROVE** **in**.

# Background: Framenet: *drive*

## Path [Path]

Excludes: Area

Any description of a trajectory of motion which is neither a **Source** nor a **Goal** expresses the frame element **Path**. This includes directional expressions and "middle of path" expressions, e.g.:

The bikers **RODE** **through the desert**.

## Source [Src]

Semantic Type: Source

Excludes: Area

Any expression which implies a definite starting-point of motion expresses the frame element **Source**. In prepositional phrases, the prepositional object expresses the starting point of motion. With particles, the starting point of motion is understood from context.

Kim **DROVE** **out of town**.

Kim **DROVE** **away**.

## Vehicle [Veh]

This is the means of conveyance controlled by the **Driver**. It may move in any way or medium. With verbs, **Vehicle** is usually expressed as Object:

I **DROVE** **my car** to Canada.

# ChatGPT



I soon noticed the car was being driven very dangerously. Who drove the car?



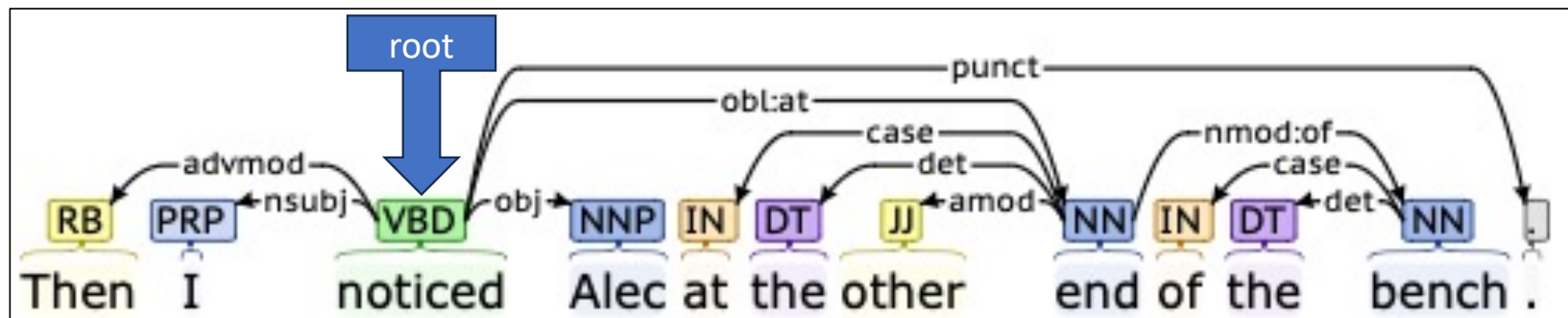
I'm sorry, but I don't have enough information to answer your question. Could you please provide more context or details about the situation, such as who was present in the car or any other relevant information?



# Background: CoreNLP

- Examples (from Framenet):

2. Then <sub>[Cognizer I]</sub> **NOTICED** <sub>[Phenomenon Alec]</sub> <sub>[Ground at the other end of the bench]</sub> .



**VERB**(NSUBJ, OBJ)  
*noticed(I, Alec)*

*event(e, noticed) &  
cognizer(e, I) &  
phenomenon(e, Alec) &  
ground(e, other end of the bench)*



# ChatGPT



Then I noticed Alec at the other end of the bench. Where was I?



Based on the information provided, you were likely at the same bench where you noticed Alec but at a different end of it.



# Background: Stanford Dependencies

- Some definitions you may find useful  
[https://nlp.stanford.edu/software/dependencies\\_manual.pdf](https://nlp.stanford.edu/software/dependencies_manual.pdf)
  - **ccomp**: clausal complement  
A clausal complement of a verb or adjective is a dependent clause
  - **dobj**: direct object  
The direct object of a VP is the noun phrase which is the (accusative) object of the verb.
  - **ibj**: indirect object  
The indirect object of a VP is the noun phrase which is the (dative) object of the verb.
  - **nsubj**: nominal subject  
A nominal subject is a noun phrase which is the syntactic subject of a clause.
  - **rcmod**: relative clause modifier  
A relative clause modifier of an NP is a relative clause modifying the NP. The relation points from the head noun of the NP to the head of the relative clause, normally a verb.
  - **vmod**: reduced non-finite verbal modifier  
A reduced non-finite verbal modifier is a participial or infinitive form of a verb heading a phrase (which may have some arguments, roughly like a VP).

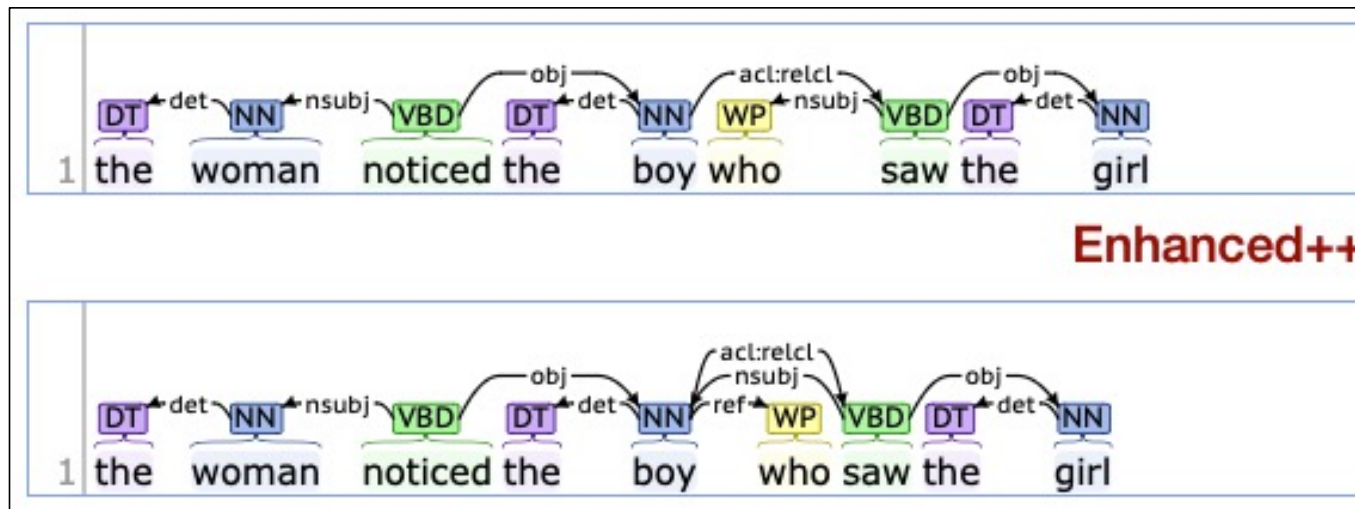
# Background: Universal Dependencies

<https://universaldependencies.org/u/dep/index.html>

	Nominals	Clauses	Modifier words	Function Words
<b>Core arguments</b>	<a href="#">nsubj</a> <a href="#">obj</a> <a href="#">iobj</a>	<a href="#">csubj</a> <a href="#">ccomp</a> <a href="#">xcomp</a>		
<b>Non-core dependents</b>	<a href="#">obl</a> <a href="#">vocative</a> <a href="#">expl</a> <a href="#">dislocated</a>	<a href="#">advcl</a>	<a href="#">advmod*</a> <a href="#">discourse</a>	<a href="#">aux</a> <a href="#">cop</a> <a href="#">mark</a>
<b>Nominal dependents</b>	<a href="#">nmod</a> <a href="#">appos</a> <a href="#">nummod</a>	<a href="#">acl</a>	<a href="#">amod</a>	<a href="#">det</a> <a href="#">clf</a> <a href="#">case</a>
<b>Coordination</b>	<b>MWE</b>	<b>Loose</b>	<b>Special</b>	<b>Other</b>
<a href="#">conj</a> <a href="#">cc</a>	<a href="#">fixed</a> <a href="#">flat</a> <a href="#">compound</a>	<a href="#">list</a> <a href="#">parataxis</a>	<a href="#">orphan</a> <a href="#">goeswith</a> <a href="#">reparandum</a>	<a href="#">punct</a> <a href="#">root</a> <a href="#">dep</a>

# Background: CoreNLP

- Root: noticed(woman, boy)
- ACL:RELCL points back to NOUN boy
- ACL:RELCL/**VERB**(NSUBJ/PRON, OBJ)
- We infer saw(boy, girl)



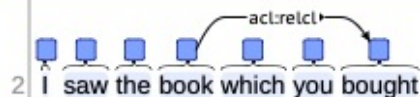
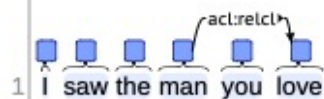
# Background: Universal Dependencies

- `acl` = adnominal clause (basically, a sentence that modifies a noun)

## `acl:relcl`: relative clause modifier

A relative clause modifier of a nominal is a clause that modifies the nominal, whereas the nominal is coreferential with a constituent inside the relative clause (here the constituent may be realized as a relative pronoun, another relative word, or it may not be overtly realized at all). The `acl:relcl` relation points from the head of the modified nominal to the head of the relative clause.

Depending on language, it may be required that relative clauses are finite. For example, English non-finite clauses are traditionally not termed relative; therefore, *the girl **that was born today*** is a relative clause because it is finite, while *the girl **born today*** is non-finite (the participle is not accompanied by a finite auxiliary) and it uses the plain `acl` relation. In other languages however, the distinction between finite and non-finite clauses may not exist or may not be used as a criterion for relative clauses.



# Homework 9

Question 1: We will compute predicate-argument structure using Perl regex with recursively embedded subject relative clauses:

1. the woman encountered the boy
  2. the woman encountered the boy who encountered the girl
  3. the woman encountered the boy who encountered the girl who found the man
  4. the woman encountered the boy who encountered the girl who found the man who chased the cat
- You can assume fixed expression *the noun* for noun phrases (NPs) and *who* for the relative pronoun. Also for simplicity, you may assume exactly one space between words.

# Homework 9

- Write a Perl program using a regex to compute the predicate-argument relations for sentences 1-4 and print them.
  - Examples:
    1. *the woman encountered the boy*
      - encountered(woman, boy)
    2. *the woman encountered the boy who encountered the girl*
      - encountered(woman, boy)
      - encountered(boy, girl)
- etc.*

# Homework 9: ChatGPT



The woman encountered the boy who encountered the girl. Who encountered who?



In this sentence, the woman encountered the boy, and the boy encountered the girl. So, the encounters happened in the following order:



1. The woman encountered the boy.
2. The boy encountered the girl.



# Homework 9

- Code should be general, i.e. **you can swap out the verbs and common nouns etc.**, and it should still work.
- For simplicity, you may assume the patterns:
  - the *noun*<sub>1</sub> *verb* the *noun*<sub>2</sub>      ⇒      *verb*(*noun*<sub>1</sub>, *noun*<sub>2</sub>)
  - the *noun*<sub>1</sub> who *verb* the *noun*<sub>2</sub>      ⇒      *verb*(*noun*<sub>1</sub>, *noun*<sub>2</sub>)
- **Hints:**
  - note the pattern overlap, use lookahead (*?=pattern*)
  - Recommend you write a Perl program (not a one-liner)
  - Input taken from the command line
  - you can collect the words together on the command line into a single string with `$sentence = qq/@ARGV/;`

# Homework 9

- Examples:

```
$ perl hw9.perl the woman encountered the boy  
encountered(woman, boy)
```

```
$ perl hw9.perl the woman encountered the boy who encountered the girl who  
found the man  
encountered(woman, boy)  
encountered(boy, girl)  
found(girl, man)
```

```
$ perl hw9.perl the woman encountered the boy who encountered the girl who  
found the man who chased the cat  
encountered(woman, boy)  
encountered(boy, girl)  
found(girl, man)  
chased(man, cat)
```

# Homework 9

Question 2: consider a second type of embedded relative clauses.

Examples:

5. the woman discovered the boy the girl encountered
  6. the woman discovered the boy the girl the man found encountered
  7. the woman discovered the boy the girl the man the cat chased found encountered
- Explain the differences between sentences in 5–7 from Q2 vs. 2–4 from Q1 with respect to:
    - predicate-argument structure, and
    - the relative pronoun.

# Homework 9: ChatGPT



The woman discovered the boy the girl the man found encountered.  
Who encountered who?



In the sentence, "The woman discovered the boy the girl the man found encountered," the man found the girl, and the girl encountered the boy. So, the encounters happened in the following order:



1. The man found the girl.
2. The girl encountered the boy.

# Homework 9: ChatGPT

SA

The woman discovered the boy the girl the man the cat chased found encountered. Who found who?



In the sentence, "The woman discovered the boy the girl the man the cat chased found encountered," the cat was chased by the man, and the man found the girl. So, the man found the girl.



# Homework 9: ChatGPT



The woman discovered the boy the girl the man the cat chased found encountered. Who chased who?



In the sentence, "The woman discovered the boy the girl the man the cat chased found encountered," the cat chased the man.



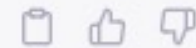
# Homework 9: ChatGPT

SA

The woman discovered the boy the girl the man the cat chased found encountered. Who encountered who?



In the sentence, "The woman discovered the boy the girl the man the cat chased found encountered," the girl encountered the boy.



# Homework 9

Question 3: try [CoreNLP](#) on the sentences with relative clauses from Q2.

5. the woman discovered the boy the girl encountered
  6. the woman discovered the boy the girl the man found encountered
  7. the woman discovered the boy the girl the man the cat chased found encountered
- Which one(s) does/do CoreNLP get wrong or are they all correct?
  - As a human processor, which of 5–7 do you find difficult to parse?



# Background: Universal Dependencies

<https://universaldependencies.org/u/dep/index.html>

	Nominals	Clauses	Modifier words	Function Words
<b>Core arguments</b>	<a href="#">nsubj</a> <a href="#">obj</a> <a href="#">iobj</a>	<a href="#">csubj</a> <a href="#">ccomp</a> <a href="#">xcomp</a>		
<b>Non-core dependents</b>	<a href="#">obl</a> <a href="#">vocative</a> <a href="#">expl</a> <a href="#">dislocated</a>	<a href="#">advcl</a>	<a href="#">advmod*</a> <a href="#">discourse</a>	<a href="#">aux</a> <a href="#">cop</a> <a href="#">mark</a>
<b>Nominal dependents</b>	<a href="#">nmod</a> <a href="#">appos</a> <a href="#">nummod</a>	<a href="#">acl</a>	<a href="#">amod</a>	<a href="#">det</a> <a href="#">clf</a> <a href="#">case</a>
<b>Coordination</b>	<b>MWE</b>	<b>Loose</b>	<b>Special</b>	<b>Other</b>
<a href="#">conj</a> <a href="#">cc</a>	<a href="#">fixed</a> <a href="#">flat</a> <a href="#">compound</a>	<a href="#">list</a> <a href="#">parataxis</a>	<a href="#">orphan</a> <a href="#">goeswith</a> <a href="#">reparandum</a>	<a href="#">punct</a> <a href="#">root</a> <a href="#">dep</a>

# Homework 9

- Extra Credit Question 4:
  - based on what we've learnt so far, do you think it's possible to write a Perl regex program that prints the correct predicate-argument structure for the examples from Q2?