Dependency Syntax

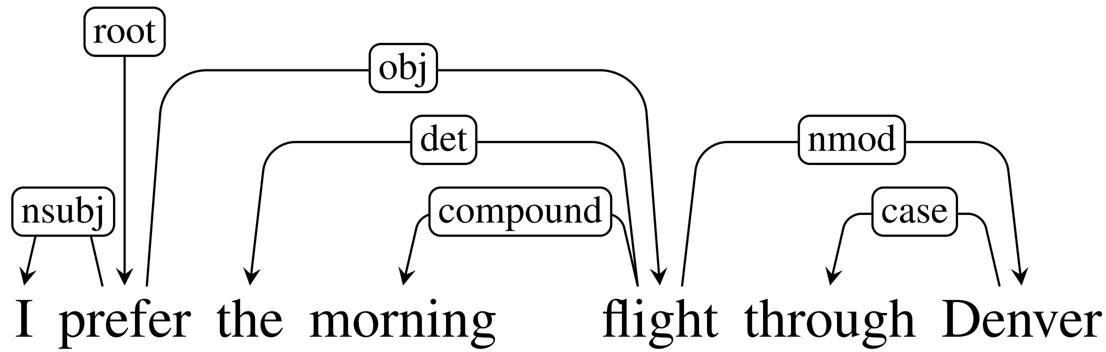
CS 485, Fall 2024
Applications of Natural Language Processing

Brendan O'Connor
College of Information and Computer Sciences
University of Massachusetts Amherst

Proj. Proposals

due Fri 10/18

Typed dependency parse



- (Labeled) directed graph among all words (tokens) in a sentence
  - Every word has exactly one parent
  - Single root node. Often a tree (and always a DAG)

- Edge labels indicate grammatical relationships
- Dependency structures work well with free word order languages
- https://universaldependencies.org/en/dep/

Clausal Argument Relations	**Description**
NSUBJ	Nominal subject
OBJ	Direct object
IOBJ	Indirect object
CCOMP	Clausal complement
Nominal Modifier Relations	**Description**
NMOD	Nominal modifier
AMOD	Adjectival modifier
APPOS	Appositional modifier
DET	Determiner
CASE	Prepositions, postpositions and other case markers
Other Notable Relations	**Description**
CONJ	Conjunct
CC	Coordinating conjunction

Figure 19.2 Some of the Universal Dependency relations (de Marneffe et al., 2021).

From constituency structure to dependency graphs

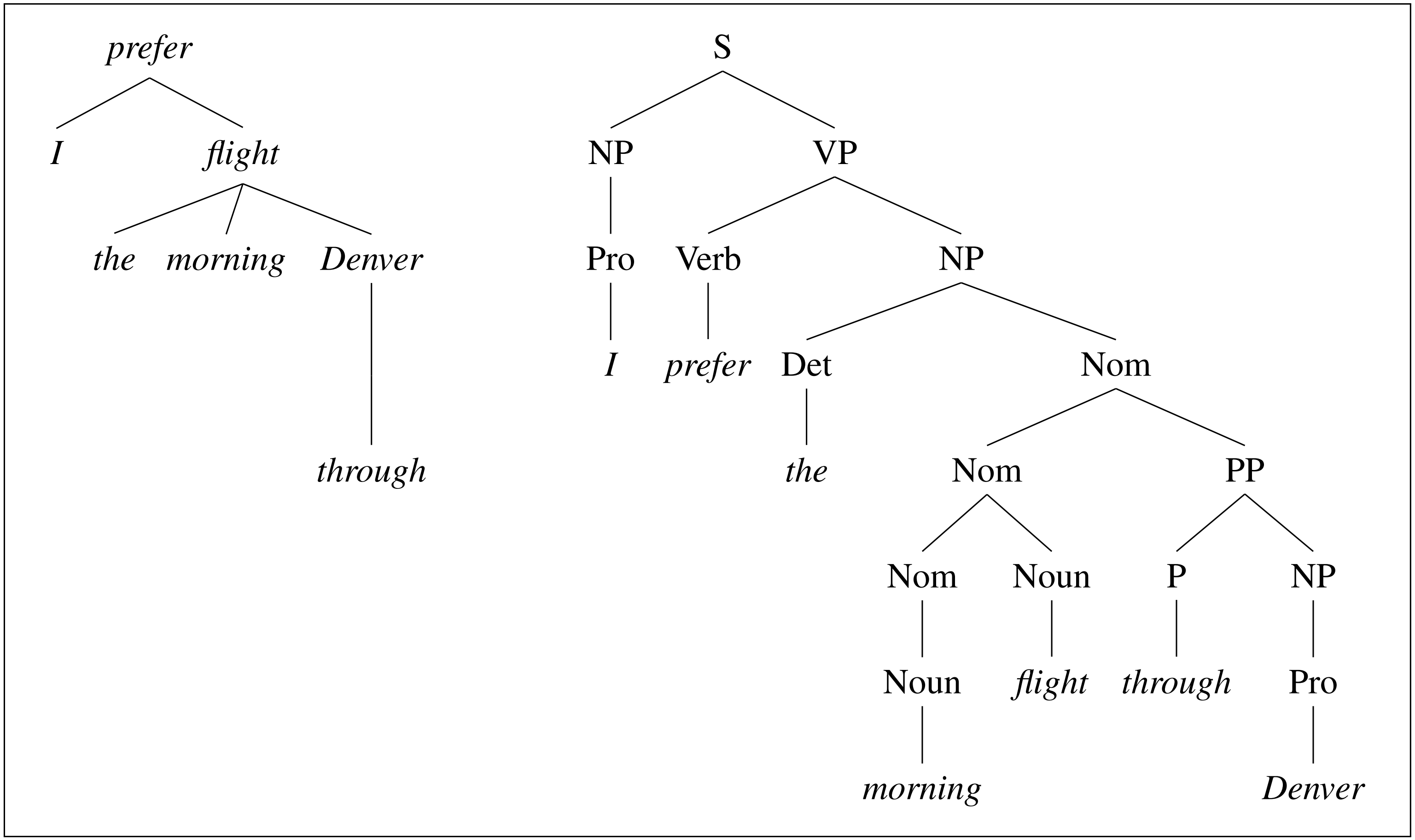
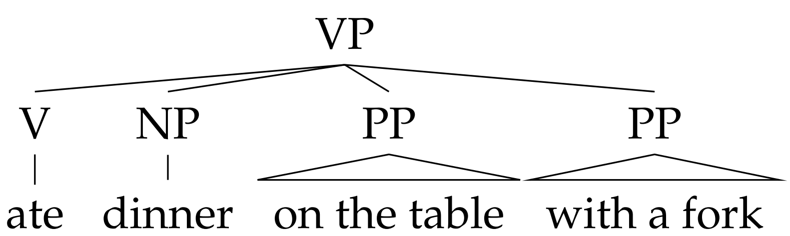


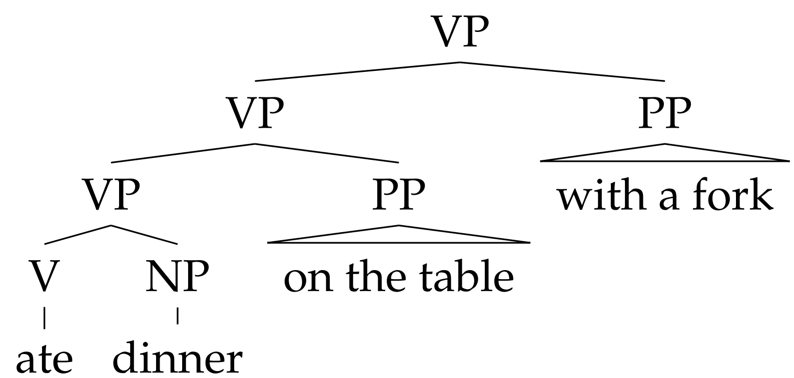
Figure 19.1 Dependency and constituent analyses for *I prefer the morning flight through Denver.*

- Dep. subgraph corresponds to a some constituent
  - ... and the dep. subgraph's root is the "***head***" of that constitutent

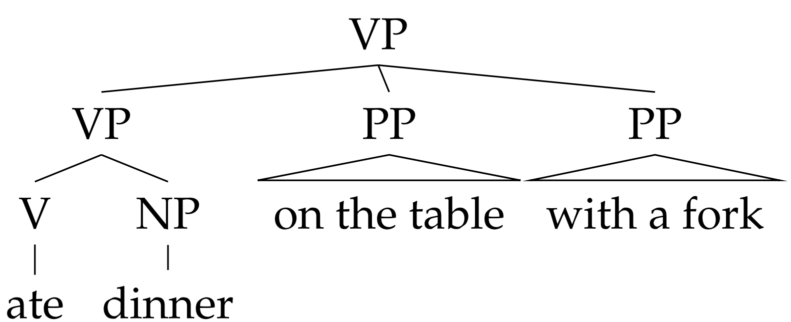
- Dependencies tend to be less specific than constituent structure



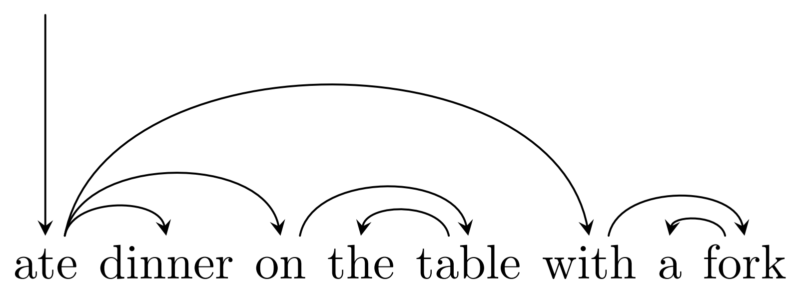
(a) Flat



(b) Two-level (PTB-style)



(c) Chomsky adjunction



(d) Dependency representation

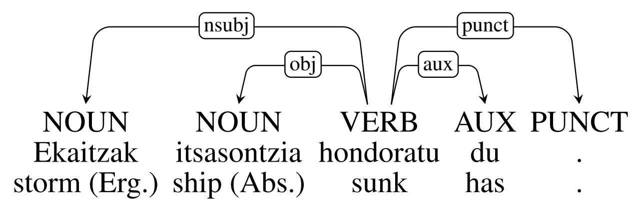
[Example: Jacob Eisenstein]

Headedness for *phrase* relations

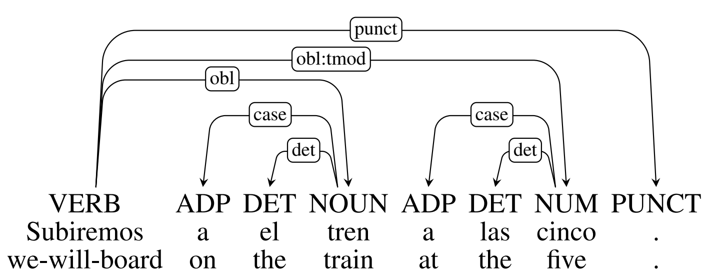
- Is a given word X the subject of verb Y?
- Is a given *phrase* X the subject of verb Y?

Universal Dependencies

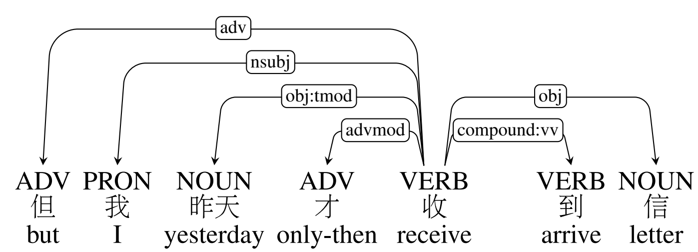
- Dependency treebanks are available for *many* different languages
  - https://universaldependencies.org/
- Many open-source dependency parsers (and tagging/POS/morphology) trained on them are also widely available; e.g. Stanza, SpaCy, etc.



[Basque] Ekaitzak itsasontzia hondoratu du. “The storm has sunk the ship.”(19.5)



[Spanish] Subiremos al tren a las cinco. “We will be boarding the train at five.”(19.4)



[Chinese] 但我昨天才收到信 “But I didn’t receive the letter until yesterday”(19.6)

Shift-reduce, transition parsing

- How to predict a parse structure for an input sentence? Deal with massive ambiguity.
- *Incremental parsing*: proceed left-to-right, building up the parse structure incrementally.
  - Interesting analogy for human sentence processing
  - (Many similar or sometimes quite different algorithms exist for both constituency and dependency parsing!)

Transition-based parsing

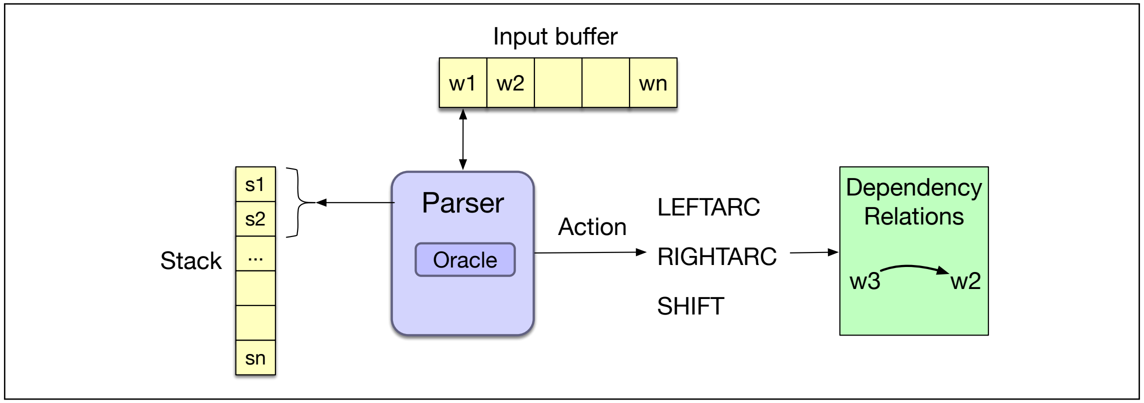


Figure 19.4 Basic transition-based parser. The parser examines the top two elements of the stack and selects an action by consulting an oracle that examines the current configuration.

```
function DEPENDENCYPARSE(words) returns dependency tree

  state ← {[root], [words], [] }  ; initial configuration
  while state not final
    t ← ORACLE(state)        ; choose a transition operator to apply
    state ← APPLY(t, state)  ; apply it, creating a new state
  return state
```

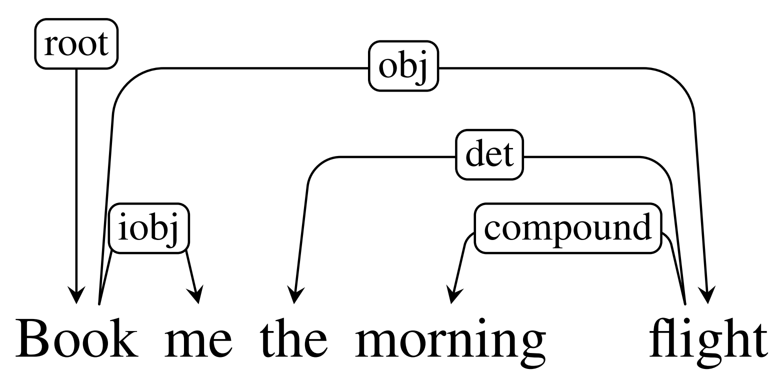
Figure 19.5 A generic transition-based dependency parser

- State machine with exactly 3 allowed actions
- At runtime: machine learned classifier to decide action

- Possible actions
  - SHIFT: remove word from front of buffer, push word on top of stack
  - LEFTARC: create edge between top and second-to-top of stack
  - RIGHTARC: create edge between top and second-to-top of stack

Step	Stack	Word List	Action	Relation Added
0	[root]	[book, me, the, morning, flight]	SHIFT	
1	[root, book]	[me, the, morning, flight]	SHIFT	
2	[root, book, me]	[the, morning, flight]	RIGHTARC	(book → me)
3	[root, book]	[the, morning, flight]	SHIFT	
4	[root, book, the]	[morning, flight]	SHIFT	
5	[root, book, the, morning]	[flight]	SHIFT	
6	[root, book, the, ~~morning~~, flight]	[]	LEFTARC	(morning ← flight)
7	[root, book, ~~the~~, flight]	[]	LEFTARC	(the ← flight)
8	[root, book, ~~flight~~]	[]	RIGHTARC	(book → flight)
9	[root, book]	[]	RIGHTARC	(root → book)
10	[root]	[]	Done	

Figure 19.6 Trace of a transition-based parse.



Transition model

- Feature templates: can use all information from current parser state (stack, buffer, edges so far)
  - Current word on top of each? At second position?
  - Current POS tag on top of each? At second position?
  - Are they left or right of each other in the sentence?
  - etc.

Stack	Word buffer	Relations
[root, canceled, flights]	[to Houston]	(canceled → United) (flights → morning) (flights → the)

$$\langle s_1.w = \textit{flights}, op = \textit{shift}\rangle$$
$$\langle s_2.w = \textit{canceled}, op = \textit{shift}\rangle$$
$$\langle s_1.t = \textit{NNS}, op = \textit{shift}\rangle$$
$$\langle s_2.t = \textit{VBD}, op = \textit{shift}\rangle$$
$$\langle b_1.w = \textit{to}, op = \textit{shift}\rangle$$
$$\langle b_1.t = \textit{TO}, op = \textit{shift}\rangle$$
$$\langle s_1.wt = \textit{flightsNNS}, op = \textit{shift}\rangle$$

- Training time: use rule system to extract oracle transition from gold-standard annotations

Dependency applications

- Dependency paths (e.g. *(fly, -nsubj->, bird)*) can be used as less sparse alternative to n-grams
    - Sometimes helps, sometimes doesn't
- Dependency relations can be selected for semantic relationships
- At a higher level, word-to-word dependencies are key to current "Transformer" neural net models, but explicit syn. deps are used less often

Dependency pattern statistics

Hand-built dependency patterns to get specific semantic relationships between words

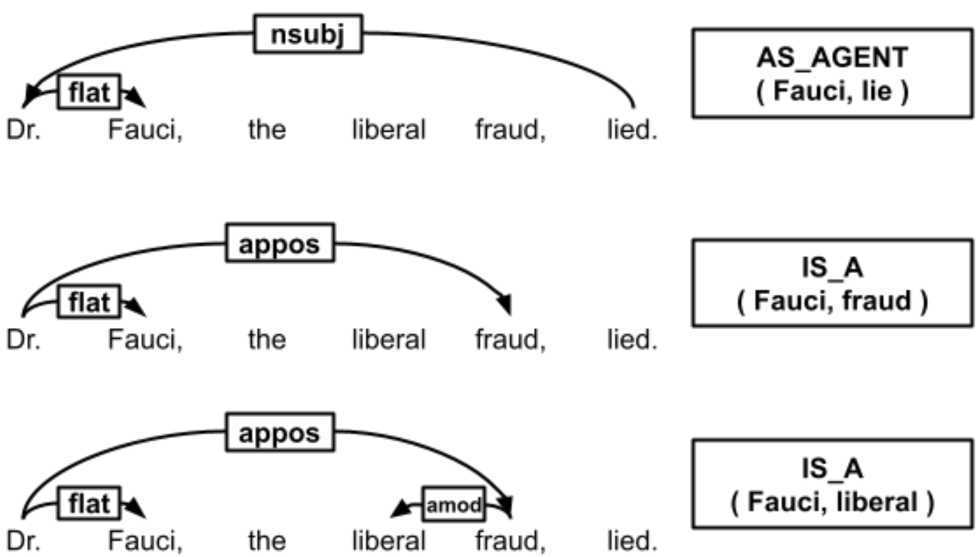


Figure 1: Examples of dependencies and TweetIE's entity attribute extraction system (§4).

4.3.1 IS_A

The IS_A relation covers any nominal or adjectival properties stated to directly pertain to the target entity, represented using the following patterns:[5]

1. target $\overset{\text{nsubj}}{\longleftrightarrow}$ property$_{nom}$
2. property$_{adj}$ $\xrightarrow{\text{nsubj}}$ target
3. target $\overset{\text{appos}}{\longleftrightarrow}$ property$_{nom}$
4. target $\xrightarrow{\text{compound}}$ property$_{nom}$
5. target $\xrightarrow{\text{amod}}$ property$_{adj}$
6. target $\overset{\text{nsubj}}{\longleftrightarrow}$ property$_{nom}$ $\xrightarrow{\text{amod}}$ property$_{adj}$
7. target $\overset{\text{appos}}{\longleftrightarrow}$ property$_{nom}$ $\xrightarrow{\text{amod}}$ property$_{adj}$

[Eggleston and O'Connor, 2022]

Relation	**Trump-Leaning** ($t < -2$)	**Biden-Leaning** ($t > 2$)
IS_A(fauci, $property_{\text{nom}}$)	murderer**, joke**, hack*, fraud*, rat*, flip*, idiot, flop, state, prison, fake, jail	nih**, hero, md, director, president
IS_A(fauci, $property_{\text{adj}}$)	fake*, little*, deep, liberal, wrong, corrupt	beloved, optimistic, best
AS_AGENT(fauci, *verb*)	sweat**, force**, need*, help*, read*, lie*, know*, let*, not_fund*, not_understand*, flip, predict, write, make, stick, hold, prove, want, not_say, admit, not_get, demand, issue, laugh, state, put, spread, pull	speak**, join*, warn*, throw, not_recommend, offer, provide, respond, consider, debunk, fail, reveal
AS_PATIENT(fauci, *verb*)	not_trust***, screw, prosecute, grill, keep to, arrest, expose, lock, do to, remove, accord to, look like, mean, blast, read	know*, feature, discredit, threaten, worship, join, insult
HAS_A(fauci, *object*)	friend*, nih*, family, mind, hand, ex-employee, involvement, fraud, mask	guidance, time
AS_CONJUNCT(fauci, *conj.*)	gates***, obama**, bill gates*, biden*, brix, cdc, rest, covid, nih, company, government	director, experts

Table 5: TweetIE extractions with at least 20 unique users with a county-level political valence t-statistic outside of [-2, 2]. Results are reported in decreasing absolute value t-statistic. * $|t| > 3$, ** $|t| > 4$, *** $|t| > 5$.

- From geo-located tweets, Mar-Dec 2020

[Eggleston and O'Connor, 2022]