POLITECNICO DI MILANO

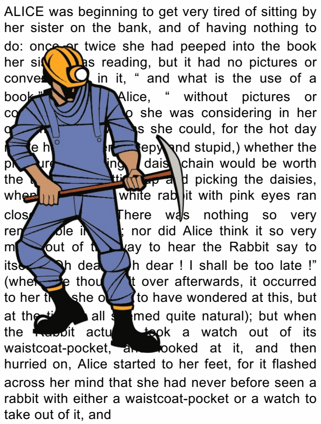
Natural Language Processing

Sequence Labelling

Natural Language Processing

Some slide content based on textbooks:
***Speech and Language Processing: An Introduction to Natural Language Processing, Computational Linguistics, and Speech Recognition** by Daniel Jurafsky and James H. Martin*

Mark Carman



Miner Image source: https://freesvg.org/miner-1574424884

Lecture Contents:

- What is sequence labelling?
- Application: POS tagging
- Application: Named Entity Recognition
- How do sequence labellers work?
- Related applications for extracting information from text
  - Entity linking
  - Relation extraction

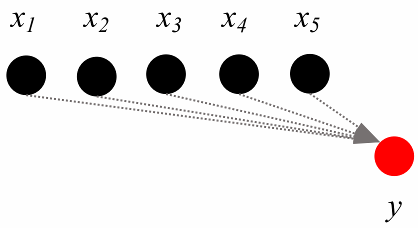
Mark Carman

What is Sequence Labelling?

What is sequence labelling?

Classification task:

- input: **ordered sequence** of tokens: $(x_1, x_2, \ldots. x_n)$
- output: single prediction for sequence: y

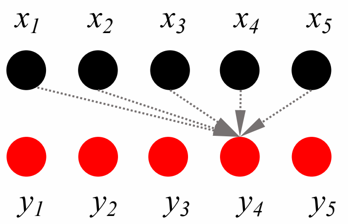


Sequence labelling task:

- input: **ordered sequence** of tokens: $(x_1, x_2, \ldots. x_n)$
- output **sequence of predictions:** $(y_1, y_2, \ldots. y_n)$

Note that:

- prediction for y_4 may depend not only on the sequence up to that point (x_1, x_2, x_3, x_4), but also the subsequent sequence $(x_5, \ldots)$
- dependencies exist across predicted sequence, so certain values y_4 may not make sense given values for (y_1, y_2, y_3) and y_5.



Application:
part-of-speech (POS) tagging

What is POS tagging?

POS tagging:
- task of assigning to each token in a sequence:
- a **part-of-speech** label
- e.g.: PRON (pronoun), VERB, DET (determiner), NOUN, etc.

```
[(He, 'PRON'),
(thought, 'VERB'),
(he, 'PRON'),
(saw, 'VERB'),
(an, 'DET'),
(elephant, 'NOUN'),
(riding, 'VERB'),
(a, 'DET'),
(bicyle, 'NOUN'),
(on, 'ADP'),
(the, 'DET'),
(freeway, 'NOUN')]
```

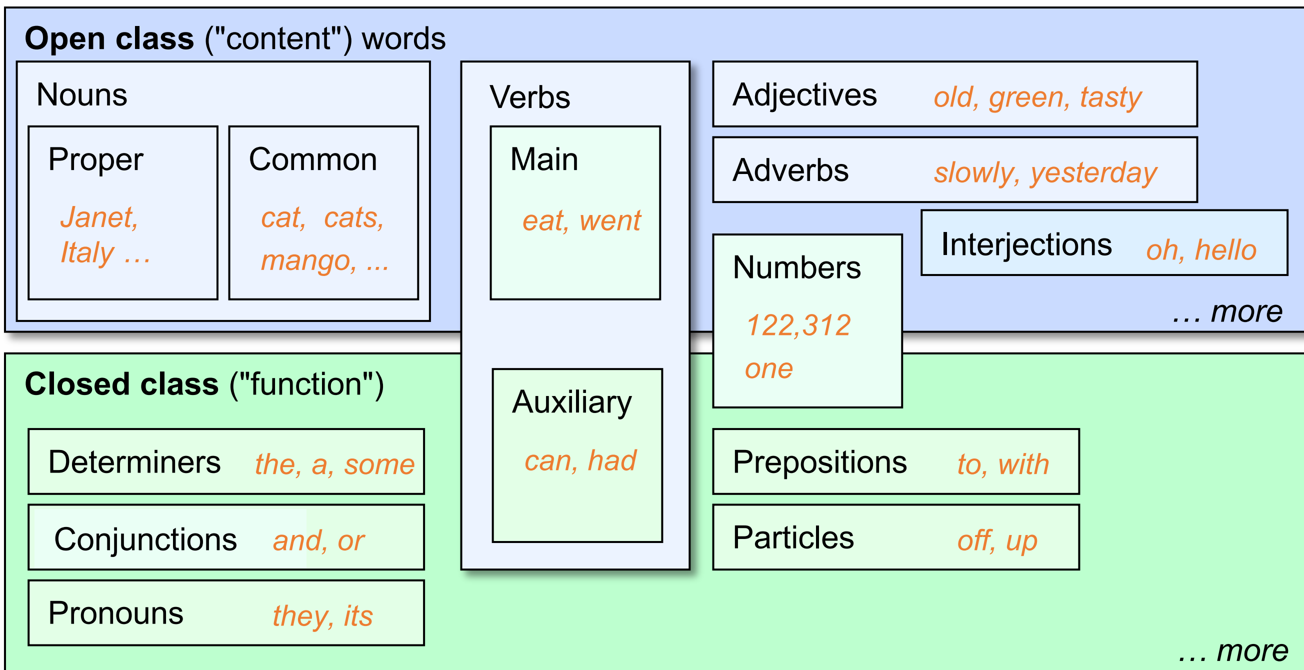
Why label parts-of-speech?
- useful for **developing features** for certain tasks
  - e.g. authorship attribution, particularly if only small amount of training data is available
- useful to **reduce ambiguity** in bag-of-words representation
  - some terms have different meaning depending on context "to **book**" vs "a **book**"
  - so append POS tag to each word occurrence: book_VERB vs book_NOUN
- useful as **initial step** for other NLP tasks or performing linguistic analysis
  - required for syntactic parsing
  - useful for text-to-speech
    - pronouncing "lead group" vs "lead weight" or "to object" vs "an object"
  - studying linguistic change like creation of new words, or meaning shift

Parts of Speech classes

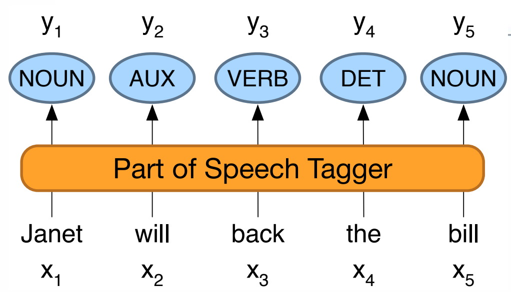
Word classes have been around a **long time**:

- Back in the 1st century BCE, Dionysius Thrax of Alexandria defined: *nouns, verbs, pronouns, prepositions, adverbs, conjunctions, participles, articles*

Modern grammar divides world classes into open and closed:



Parts of Speech tagging



Map sequence of words $x_1, \ldots, x_n$ to sequence of POS tags $y_1, \ldots, y_n$

- set of tags:

	Tag	Description	Example
Open Class	**ADJ**	Adjective: noun modifiers describing properties	*red, young, awesome*
	ADV	Adverb: verb modifiers of time, place, manner	*very, slowly, home, yesterday*
	NOUN	words for persons, places, things, etc.	*algorithm, cat, mango, beauty*
	VERB	words for actions and processes	*draw, provide, go*
	PROPN	Proper noun: name of a person, organization, place, etc..	*Regina, IBM, Colorado*
	INTJ	Interjection: exclamation, greeting, yes/no response, etc.	*oh, um, yes, hello*
Closed Class Words	**ADP**	Adposition (Preposition/Postposition): marks a noun's spacial, temporal, or other relation	*in, on, by under*
	AUX	Auxiliary: helping verb marking tense, aspect, mood, etc.,	*can, may, should, are*
	CCONJ	Coordinating Conjunction: joins two phrases/clauses	*and, or, but*
	DET	Determiner: marks noun phrase properties	*a, an, the, this*
	NUM	Numeral	*one, two, first, second*
	PART	Particle: a preposition-like form used together with a verb	*up, down, on, off, in, out, at, by*
	PRON	Pronoun: a shorthand for referring to an entity or event	*she, who, I, others*
	SCONJ	Subordinating Conjunction: joins a main clause with a subordinate clause such as a sentential complement	*that, which*
Other	**PUNCT**	Punctuation	; , ()
	SYM	Symbols like $ or emoji	$, %
	X	Other	asdf, qwfg

- example sentences:
  - There/PRO were/VERB 70/NUM children/NOUN there/ADV ./PUNC
  - Preliminary/ADJ findings/NOUN were/AUX reported/VERB in/ADP today/NOUN 's/PART New/PROPN England/PROPN Journal/PROPN of/ADP Medicine/PROPN

Is POS tagging difficult?

Approximately 85% of vocabulary terms in English are unambiguous

- *Janet* is always PROPN, *hesitantly* is always ADV

But ambiguous vocab terms are very common

- so ~60% of tokens are ambiguous

Example: word ***back*** could have 5 different POS tags:

- *earnings growth took a back/ADJ seat*
- *a small building in the back/NOUN*
- *a clear majority of senators back/VERB the bill*
- *enable the country to buy back/PART debt*
- *I was twenty-one back/ADV then*

Accuracy of POS tagging is about 97%

- changed little in last 10+ years: HMMs, CRFs, and BERT perform similarly
- similar to human accuracy
- baseline (label each word with its most frequent tag) performance already 92%

Features used for POS tagging

Consider the example:

- *Janet will back the bill*

  AUX/NOUN/VERB? NOUN/VERB?

Sources of evidence for determingint the POS tags:

- Prior probabilities of word/tag
  - "will" is usually an AUX
- Identity of neighboring words
  - "the" means the next word is probably not a verb
- Morphology and wordshape:
  - Prefixes unable: un- → ADJ
  - Suffixes importantly: -ly → ADJ
  - Capitalization Janet: CAP → PROPN

Application:
named-entity-recognition (NER)

What is entity recognition?

Named-Entity Recognition (NER):

- task of identifying entities that are mentioned in a text
- can be treated as a sequence labelling task
- often a first step in extracting knowledge from text

```
"Have you heard of an associate professor from
the Politecnico di Milano called Mark Carman?"
```

Institution Person

Named Entity Recognition

Named entity = object in real world

- most common tags:
  - PER (Person): e.g. "Marie Curie"
  - LOC (Location): e.g."Lake Michigan"
  - ORG (Organization): e.g. "Stanford University"
  - GPE (Geo-Political Entity): e.g. "Boulder, Colorado"
- often **multi-word phrases**
- term also extended to things that aren't entities:
  - dates, times, prices

Difference between a GPE and a LOC?
- *GPE : geopolitical entities, e.g. everything with a governing body like cities and countries. Examples: "Germany", "Buenos Aires".*
- *LOC : everything else that's a physical location or area, like "Kalahari Desert" or "Silicon Valley"*

Source: https://support.prodi.gy/t/ner-annotation-scheme-gpe-vs-loc/2913

NER task: find spans of text that constitute proper names

- tagging the type of the entity:

Citing high fuel prices, [ORG **United Airlines**] said [TIME **Friday**] it has increased fares by [MONEY **$6**] per round trip on flights to some cities also served by lower-cost carriers. [ORG **American Airlines**], a unit of [ORG **AMR Corp.**], immediately matched the move, spokesman [PER **Tim Wagner**] said. [ORG **United**], a unit of [ORG **UAL Corp.**], said the increase took effect [TIME **Thursday**] and applies to most routes where it competes against discount carriers, such as [LOC **Chicago**] to [LOC **Dallas**] and [LOC **Denver**] to [LOC **San Francisco**].

Why NER?

Traditionally perform NER for:

- **Sentiment analysis**: identify sentiment towards particular company or person?
- **Information extraction**: extracting facts about entities from text
- **Question answering**: answer questions about an entity?
- **De-identification**: remove references to individual from text to protect privacy

NER is hard because of:

1) segmentation: in POS tagging each word gets one tag, while in NER have to find and segment entities
2) type ambiguity: same word/phrase could have many types depending on the context

[PER Washington] was born into slavery on the farm of James Burroughs.
[ORG Washington] went up 2 games to 1 in the four-game series.
Blair arrived in [LOC Washington] for what may well be his last state visit.
In June, [GPE Washington] passed a primary seatbelt law.

Begin-Inside-Outside (BIO) Tagging

NER **finds phrases** in the text referring to named entities:

- *[PER Jane Villanueva] of [ORG United] , a unit of [ORG United Airlines Holding] , said the fare applies to the [LOC Chicago] route.*

How can we turn NER into sequence labeling problem (with one label per token)?

- use begin/inside/outside tags:
  - B: token that **begins** a span
  - I: tokens **inside** a span
  - O: tokens **outside** of any span

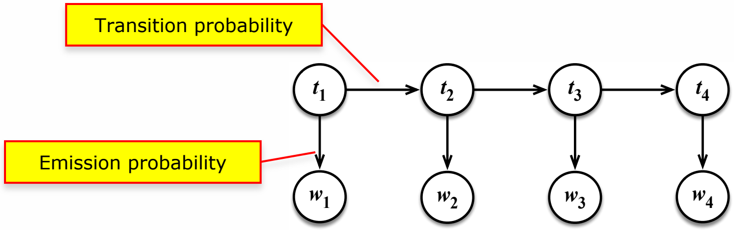
Words	BIO Label
Jane	B-PER
Villanueva	I-PER
of	O
United	B-ORG
Airlines	I-ORG
Holding	I-ORG
discussed	O
the	O
Chicago	B-LOC
route	O
.	O

Sequence Labelling: approaches

How do sequence labellers work?

Traditional methods for sequence labelling make use of

- **Hidden Markov Models** (HMMs) = Naïve Bayes applied to sequences

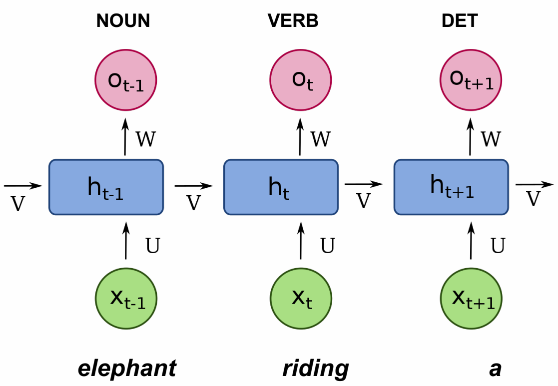


- **Conditional Random Fields** (CRFs) = Logistic Regression applied to sequences

Recent methods make use of

- Recurrent Neural Networks (RNNs) to improve performance

Recurrent Neural Networks (RNNs)



Source: https://en.wikipedia.org/wiki/Recurrent_neural_network

RNNs build upon word embeddings

- by aggregating information along sequences 👍

Provide general mechanism for combining:

- **context from previous words**
- with **embedding of current word**

Implemented as NN which:

- takes 2 input vectors: (current input, previous state)
- produce 2 output vectors: (current output, updated state)

By updating an internal state, RNN is able to:

- process arbitrarily long inputs
- produce output prediction at each input position

Aside: RNNs and word order

Word order is very important for interpreting the meaning of text

- and interpreting the meaning is important **for classifying it**

For example consider the **meaning** of the following phrases:

- There's a white rat in the house …
- There's a rat in the White House …

Negation provides another important example of word order:

- I am happy about …
- I am not happy about …
- I'd be lying if I said I was not happy about …
- I would not be lying if I said that I was not happy about …

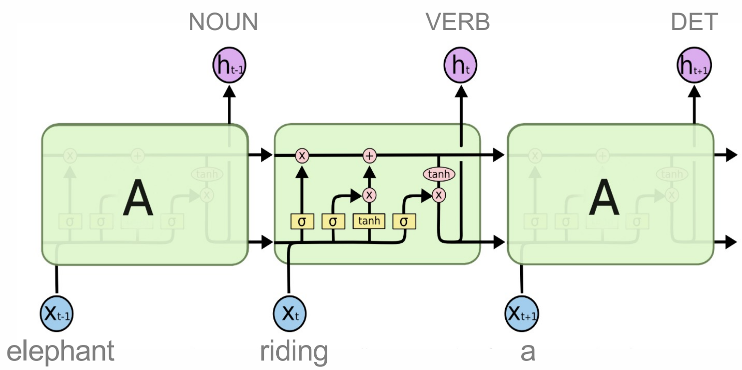
N-grams can be used to capture word order

- **but** we can never make them long enough

Source: https://commons.wikimedia.org/wiki/File:White_rat_on_table.jpg

Source: https://commons.wikimedia.org/wiki/File:White_House_DC.JPG

Long Short-Term Memory (LSTM)



Clever implementation of RNN that is able to

- learn **contexts** and **long range dependencies**

Does this by using a **gating mechanism**

- passes through information by default
- unless new information is added to state
- or deleted from it (forgotten)

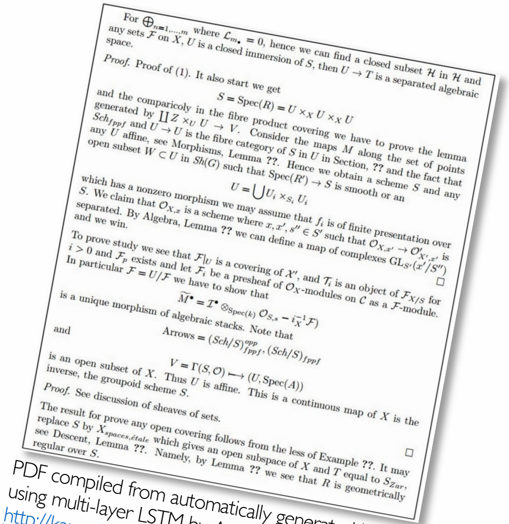
LSTM learns when & what information to

- **remember**, **forget**, and **output** at each timestep

Images source:
Understanding LSTM Networks by Christopher Olah
http://colah.github.io/posts/2015-08-Understanding-LSTMs/

Mark Carman

Aside: LSTMs and handling context



PDF compiled from automatically generated Latex using multi-layer LSTM by Andrej Karpathy
http://karpathy.github.io/2015/05/21/rnn-effectiveness/

LSTMs can be stacked on top of each other

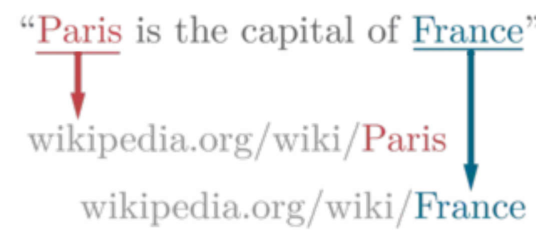
- have uncanny ability to handle nested contexts
- useful for natural language:
  - for example, complete sentences with: *he/she/his/her*

    My mother was taking on the phone to ___ friend Jim.
    Jim said that ___ favourite game was confusing ___ students.
    Replying, ___ said that ___ should find a better hobby.
  - gender of subject changes for each subsequent sentence
  - another example, this time with negation, complete with: *friendly/self-absorbed*

    I get along well with her brother. He's always ___
    I can not get along well with her brother. He's always ___
    I can not help but get along well with her brother. He's always ___
  - LSTMs are able to switch between sentence and negation contexts

Entity Linkage

What is entity linkage?



Source: https://commons.wikimedia.org/wiki/File:Entity_Linking_-_Short_Example.pn

Determining that a named-entity has been mentioned in text
- often **only the first part** of problem
- second part: determine **which real-word entity** was referred to
- not as easy as it sounds!

Linkage techniques make use of:
- relative importance of entities
- context within text (other entities present)

> I grew up in a small town just out of Paris.
> Currently driving from Dallas to Paris.
> Paris Hilton was photographed leaving the Paris Hilton.

> Just had my photo taken with Michael Jordan!!
> Just had my photo taken with Michael Jordan at EMNLP!!

Ontology/Knowledge Base
- generally **Wikipedia/DBPedia** is used
  - but many individuals/objects have no Wikipedia page
  - so custom custom ontologies can be used

Aside: Taxonomies and Ontologies

What are taxonomies and ontologies?

Taxonomy = hierarchy of concepts (e.g. types of products with is-a or part-of relationships)

Ontology = formal definition of concepts belonging to a domain

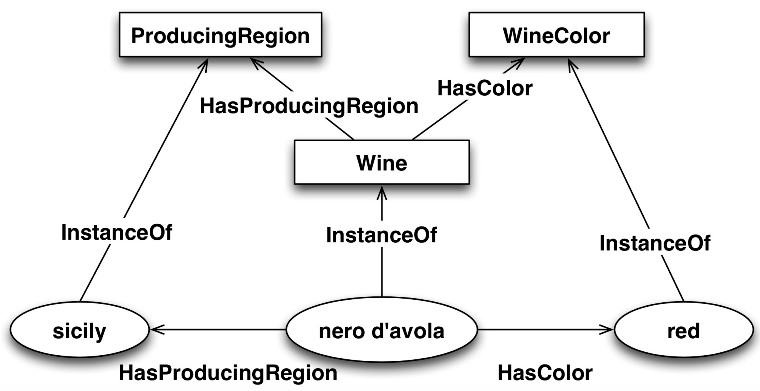
- Abstract definition of concepts that does not depend on the language

Most ontologies are composed of:

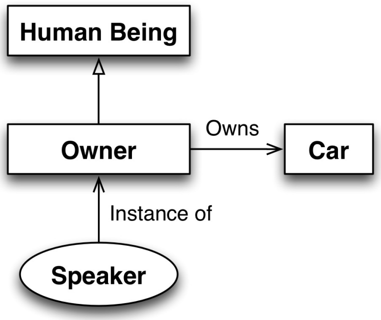
- Classes (e.g. **`wine,winery`**): A set of elements; a type
- Individuals (e.g. **`champagne`**): An element/objectd
- Attributes (e.g. **`price`**): property with primitive data type (e.g. **`string/integer`**) allowing for restrictions on values (e.g., “>0”)
- Relationships (e.g. **`winery produces wine`**): characterization of relationships among classes or individuals
- Logical rules, e.g.:
  `hasParent(?x1,?x2) ^ hasBrother(?x2,?x3) → hasUncle(?x1,?x3)`

Ontologies as graphs

The relationships between concepts in an ontology/knowledge base form a graph:



- Ontologies/knowledge bases can be used to represent information (called "facts") contained in sentences
  - e.g. for the sentence *"I have a car"*



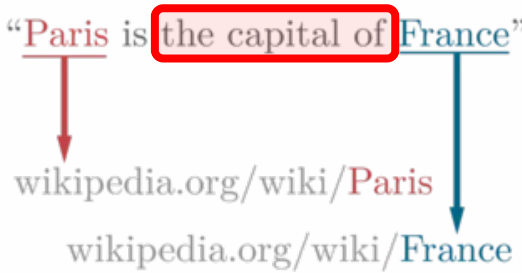
Web Ontologies and Knowledge Bases

OWL (Web Ontology Language)

- based on RDF (triple: *subject predicate object*), i.e. Description Logic
- uses SPARQL query language to allow inference over KB such as DBPedia
- KBs have **open world** semantics:
  - any statement that is not known to be true is unknown
  - as opposed to closed world assumption used in SQL:
    - any statement that is not known to be true is false (*negation as failure*)
  - example if KB contains propositions:
    "Giovanni is an architect", "Giovanni is not a physicist"
    - query: *"Is Giovanni an engineer?"*
      open world answer: *unknown,* closed world answer: *no* (proposition not in KB)
    - query: *"Is Giovanni a physicist?"*
      open world answer: *no* (negated proposition found), closed world answer: *no* (proposition not in KB)

Relation Extraction

What is relation extraction?



Source: https://commons.wikimedia.org/wiki/File:Entity_Linking_-_Short_Example.png

Once entity mentions have been linked to unique entities
- relationships between entities can be mined
- and used to populate a knowledge graph / knowledge base

Handled as a problem of predicting **missing links** in a graph
- entity embeddings can be leveraged for this purpose
  - since translations in space naturally encode relationships
  - ongoing research topic, see e.g. https://arxiv.org/pdf/2002.00388.pdf

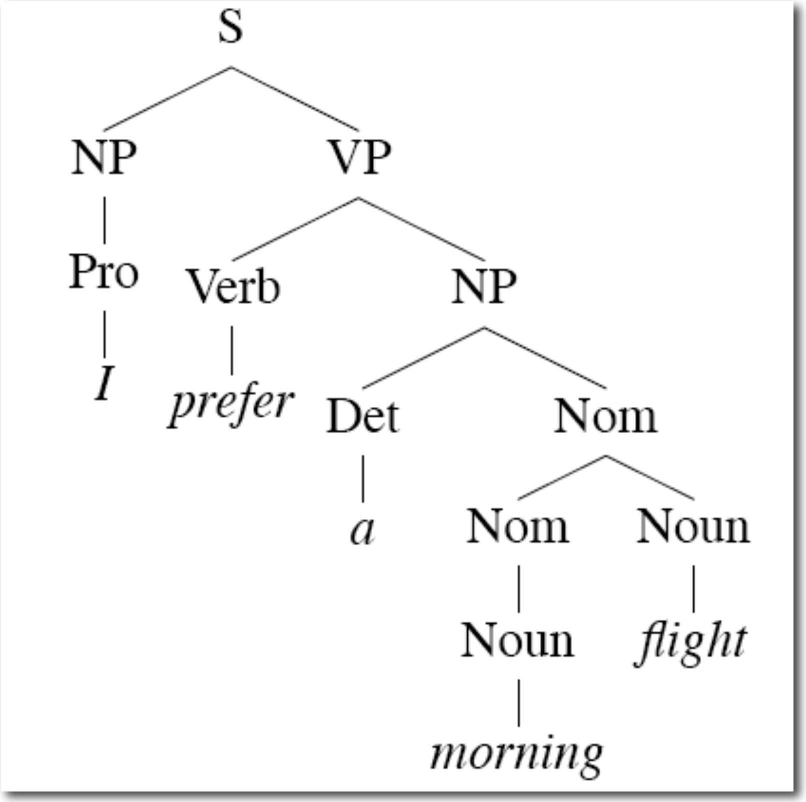
Mark Carman

Parse Trees

Parse Tree

Parse Trees result from applying a **formal grammer** to understand how a sentence was generated

- used to be very popular in NLP



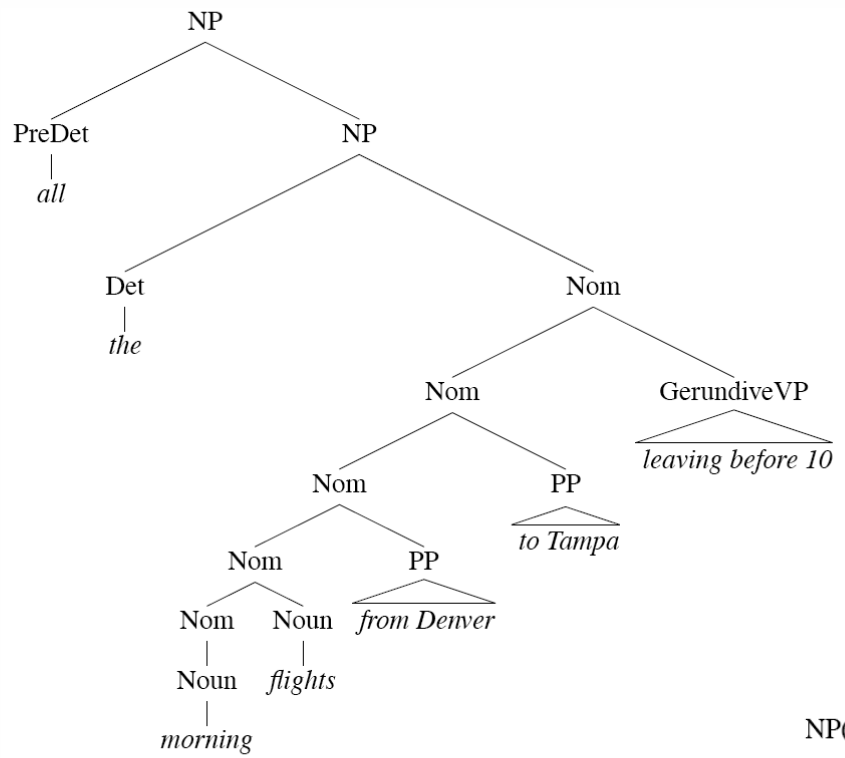
Penn Treebank

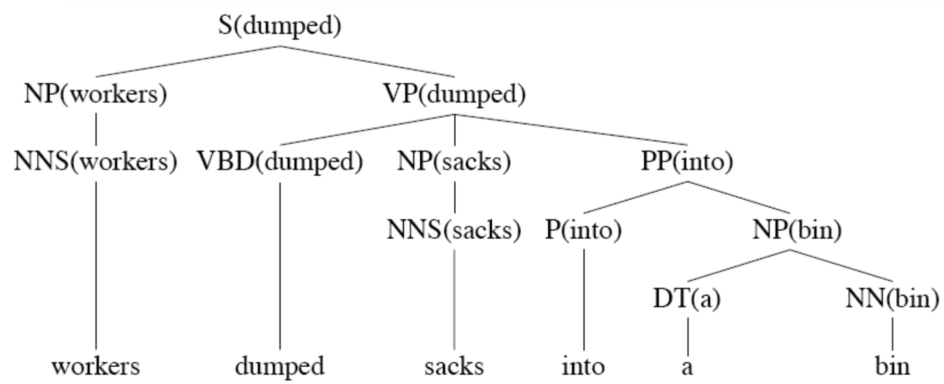
Treebank

- corpora in which sentences are paired with a parse tree
- Most famous is the Wall Street Journal section of the Penn TreeBank.
  - One million words from Wall Street Journal.

```
( (S (‘‘ ‘‘)
    (S-TPC-2
      (NP-SBJ-1 (PRP We) )
      (VP (MD would)
        (VP (VB have)
          (S
            (NP-SBJ (-NONE- *-1) )
            (VP (TO to)
              (VP (VB wait)
                (SBAR-TMP (IN until)
                  (S
                    (NP-SBJ (PRP we) )
                    (VP (VBP have)
                      (VP (VBN collected)
                        (PP-CLR (IN on)
                          (NP (DT those)(NNS assets)))))))))))))
    (, ,) (’’ ’’)
    (NP-SBJ (PRP he) )
    (VP (VBD said)
      (S (-NONE- *T*-2) ))
    (. .) ))
```

Examples of Parse Trees:





Co-reference resolution

"No, ***I*** *could never vote for* ***Trump****.* ***He*** *doesn't share* ***my*** *morals"*, **she** explained.

Co-reference Resolution

Problem of determining who or what is being referenced across (or sometimes within) sentences:

> ***John*** *went to* ***Bill's car dealership*** *to check out* ***an Acura Integra****.*
> ***He*** *looked at* ***it*** *for half an hour*

- In the second sentence, who is being referred to by the word ***he***?
  - *He* refers to *John*
- In the second sentence, what is ***it***?
  - Is it *Bill's car dealership?*
  - Or *An Acura Integra?*

Co-reference Resolution (cont.)

- Sometimes pronoun comes after referent (anaphora):

  John went to the dealership to see a car that he was interested to purchase

- Sometimes pronoun comes before referent (cataphora):

  Before he bought it, John checked over the Integra very carefully

Why resolve co-references to entities from earlier/later in the text?

- In order to be able to understand what is being said about those entities when pronouns are being used
- So needed for chatbots and information extraction …

Types of Reference Phenomena

Pronouns:

I saw no less than 6 Acura Integras today. They are the coolest cars.

Demonstratives

I bought an Integra yesterday, similar to the one I bought five years ago. That one was nice, but I like this one even more

- *This* and *that* often refer metaphorically to time

A non-pronominal anaphora

I saw no less that 6 Acura Integra today. I want one

- … one (of them)

Inferable

I almost bought an Acura Integra today, but the engine seemed noisy.

- The engine of…? Easy to infer: *the Acura Integra*

Usually resolve references to entities, but other things can be referenced in text:

According to John, Bob bought Sue an Integra, and Sue bought Fred a Legend

- *But that turned out to be a lie* (a speech act)
- *But that was false* (proposition)
- *That struck me as a funny way to describe the situation* (manner of description)
- *That caused Sue to become rather poor* (event)

Conclusions

Conclusions

TODO