POS tagging

CMSC 723 / LING 723 / INST 725

Marine Carpuat

Parts of Speech

- "Equivalence class" of linguistic entities
  - "Categories" or "types" of words

- Study dates back to the ancient Greeks
  - Dionysius Thrax of Alexandria (*c.* 100 BC)
  - 8 parts of speech: noun, verb, pronoun, preposition, adverb, conjunction, participle, article
  - Remarkably enduring list!

How can we define POS?

- By meaning?
  - Verbs are actions
  - Adjectives are properties
  - Nouns are things
- By the syntactic environment
  - What occurs nearby?
  - What does it act as?
- By what morphological processes affect it
  - What affixes does it take?
- Typically combination of syntactic+morphology

Parts of Speech

- Open class
  - Impossible to completely enumerate
  - New words continuously being invented, borrowed, etc.

- Closed class
  - Closed, fixed membership
  - Reasonably easy to enumerate
  - Generally, short function words that “structure” sentences

Open Class POS

- Four major open classes in English
  - Nouns
  - Verbs
  - Adjectives
  - Adverbs
- All languages have nouns and verbs... but may not have the other two

Nouns

- Open class
  - New inventions all the time: muggle, webinar, ...
- Semantics:
  - Generally, words for people, places, things
  - But not always (bandwidth, energy, ...)
- Syntactic environment:
  - Occurring with determiners
  - Pluralizable, possessivizable
- Other characteristics:
  - Mass vs. count nouns

Verbs

- Open class
  - New inventions all the time: google, tweet, ...
- Semantics
  - Generally, denote actions, processes, etc.
- Syntactic environment
  - E.g., Intransitive, transitive
- Other characteristics
  - Main vs. auxiliary verbs
  - Gerunds (verbs behaving like nouns)
  - Participles (verbs behaving like adjectives)

Adjectives and Adverbs

- Adjectives
  - Generally modify nouns, e.g., *tall* girl
- Adverbs
  - A semantic and formal hodge-podge…
  - Sometimes modify verbs, e.g., sang *beautifully*
  - Sometimes modify adjectives, e.g., *extremely* hot

Closed Class POS

- Prepositions
  - In English, occurring before noun phrases
  - Specifying some type of relation (spatial, temporal, …)
  - Examples: *on* the shelf, *before* noon
- Particles
  - Resembles a preposition, but used with a verb (“phrasal verbs”)
  - Examples: find *out*, turn *over*, go *on*

Particle vs. Prepositions

He came *by* the office in a hurry (by = preposition)
He came *by* his fortune honestly (by = particle)

We ran *up* the phone bill (up = particle)
We ran *up* the small hill (up = preposition)

He lived *down* the block (down = preposition)
He never lived *down* the nicknames (down = particle)

More Closed Class POS

- Determiners
  - Establish reference for a noun
  - Examples: *a*, *an*, *the* (articles), *that*, *this*, *many*, *such*, …
- Pronouns
  - Refer to person or entities: *he*, *she*, *it*
  - Possessive pronouns: *his*, *her*, *its*
  - Wh-pronouns: *what*, *who*

Closed Class POS: Conjunctions

- Coordinating conjunctions
  - Join two elements of “equal status”
  - Examples: cats *and* dogs, salad *or* soup
- Subordinating conjunctions
  - Join two elements of “unequal status”
  - Examples: We’ll leave *after* you finish eating. *While* I was waiting in line, I saw my friend.
  - Complementizers are a special case: I think *that* you should finish your assignment

Beyond English…

Chinese

No verb/adjective distinction!

漂亮: beautiful/to be beautiful

Riau Indonesian/Malay

No Articles

No Tense Marking

3rd person pronouns neutral to both gender and number

No features distinguishing verbs from nouns

Ayam (chicken) *Makan* (eat)

The chicken is eating
The chicken ate
The chicken will eat
The chicken is being eaten
Where the chicken is eating
How the chicken is eating
Somebody is eating the chicken
The chicken that is eating

POS tagging

POS Tagging: What's the task?

- Process of assigning part-of-speech tags to words

- But what tags are we going to assign?
  - Coarse grained: noun, verb, adjective, adverb, …
  - Fine grained: {proper, common} noun
  - Even finer-grained: {proper, common} noun $\pm$ animate

- Important issues to remember
  - Choice of tags encodes certain distinctions/non-distinctions
  - Tagsets will differ across languages!
- For English, Penn Treebank is the most common tagset

Penn Treebank Tagset: 45 Tags

Tag	Description	Example	Tag	Description	Example
CC	coordin. conjunction	*and, but, or*	SYM	symbol	+,%, &
CD	cardinal number	*one, two, three*	TO	“to”	*to*
DT	determiner	*a, the*	UH	interjection	*ah, oops*
EX	existential ‘there’	*there*	VB	verb, base form	*eat*
FW	foreign word	*mea culpa*	VBD	verb, past tense	*ate*
IN	preposition/sub-conj	*of, in, by*	VBG	verb, gerund	*eating*
JJ	adjective	*yellow*	VBN	verb, past participle	*eaten*
JJR	adj., comparative	*bigger*	VBP	verb, non-3sg pres	*eat*
JJS	adj., superlative	*wildest*	VBZ	verb, 3sg pres	*eats*
LS	list item marker	*1, 2, One*	WDT	wh-determiner	*which, that*
MD	modal	*can, should*	WP	wh-pronoun	*what, who*
NN	noun, sing. or mass	*llama*	WP$	possessive wh-	*whose*
NNS	noun, plural	*llamas*	WRB	wh-adverb	*how, where*
NNP	proper noun, singular	*IBM*	$	dollar sign	$
NNPS	proper noun, plural	*Carolinas*	#	pound sign	#
PDT	predeterminer	*all, both*	“	left quote	‘ or “
POS	possessive ending	*’s*	”	right quote	’ or ”
PRP	personal pronoun	*I, you, he*	(	left parenthesis	[, (, {, <
PRP$	possessive pronoun	*your, one’s*	)	right parenthesis	],), }, >
RB	adverb	*quickly, never*	,	comma	,
RBR	adverb, comparative	*faster*	.	sentence-final punc	. ! ?
RBS	adverb, superlative	*fastest*	:	mid-sentence punc	: ; ... – -
RP	particle	*up, off*			

Penn Treebank Tagset: Choices

- Example:
  - The/DT grand/JJ jury/NN commmented/VBD on/IN a/DT number/NN of/IN other/JJ topics/NNS ./.

- Distinctions and non-distinctions
  - Prepositions and subordinating conjunctions are tagged “IN” (“Although/IN I/PRP..”)
  - Except the preposition/complementizer “to” is tagged “TO”

Why do POS tagging?

- One of the most basic NLP tasks
  - Nicely illustrates principles of statistical NLP

- Useful for higher-level analysis
  - Needed for syntactic analysis
  - Needed for semantic analysis

- Sample applications that require POS tagging
  - Machine translation
  - Information extraction
  - Lots more…

Try your hand at tagging…

- The back door
- On my back
- Win the voters back
- Promised to back the bill

Try your hand at tagging…

- I hope that she wins
- That day was nice
- You can go that far

Why is POS tagging hard?

- Ambiguity!

  - Ambiguity in English
    - 11.5% of word types ambiguous in Brown corpus
    - 40% of word tokens ambiguous in Brown corpus
    - Annotator disagreement in Penn Treebank: 3.5%

POS tagging: how to do it?

- Given Penn Treebank, how would you build a system that can POS tag new text?

- Baseline: pick most frequent tag for each word type
  - 90% accuracy if train+test sets are drawn from Penn Treebank

- Can we do better?

How to POS tag automatically?

How can we POS tag automatically?

- POS tagging as multiclass classification
  - What is x? What is y?

- POS tagging as sequence labeling
  - Models sequences of predictions

Linear Models for Classification

Feature function representation

$$\hat{y} = \arg\max_{y} \boldsymbol{\theta}^{\top} \mathbf{f}(\mathbf{x}, y)$$

Weights

Multiclass perceptron

$$\hat{y} = \arg\max_{y} \boldsymbol{\theta}^\top \mathbf{f}(\mathbf{x}, y)$$

Algorithm 1 Perceptron learning algorithm

procedure PERCEPTRON($\boldsymbol{x}_{1:N}, y_{1:N}$)
    repeat
        Select an instance i
        $\hat{y} \leftarrow \arg\max_y \boldsymbol{\theta}_t^\top \boldsymbol{f}(\boldsymbol{x}_i, y)$
        if $\hat{y} \neq y_i$ **then**
            $\boldsymbol{\theta}_{t+1} \leftarrow \boldsymbol{\theta}_t + \boldsymbol{f}(\boldsymbol{x}_i, y_i) - \boldsymbol{f}(\boldsymbol{x}_i, \hat{y})$
        else
            do nothing
    until tired

POS tagging
Sequence labeling with the perceptron

Sequence labeling problem

- Input:
  - sequence of tokens $x = [x_1 \ldots x_K]$
  - Variable length K
- Output (aka label):
  - sequence of tags $y = [y_1 \ldots y_K]$
  - Size of output space?

Structured Perceptron

- Perceptron algorithm can be used for sequence labeling
- But there are challenges
  - How to compute argmax efficiently?
  - What are appropriate features?
- Approach: leverage structure of output space

Feature functions for sequence labeling

$x =$ " monsters eat tasty bunnies "

$y =$ noun verb adj noun

- Example features?
  - Number of times "monsters" is tagged as noun
  - Number of times "noun" is followed by "verb"
  - Number of times "tasty" is tagged as "verb"
  - Number of times two verbs are adjacent
  - …

Feature functions for sequence labeling

$\boldsymbol{x} =$ " monsters eat tasty bunnies "

$\boldsymbol{y} =$ noun verb adj noun

- Standard features of POS tagging
  - **Unary features:** # times word w has been labeled with tag l for all words w and all tags l
  - **Markov features:** # times tag l is adjacent to tag l' in output for all tags l and l'
- Size of feature representation is constant wrt input length

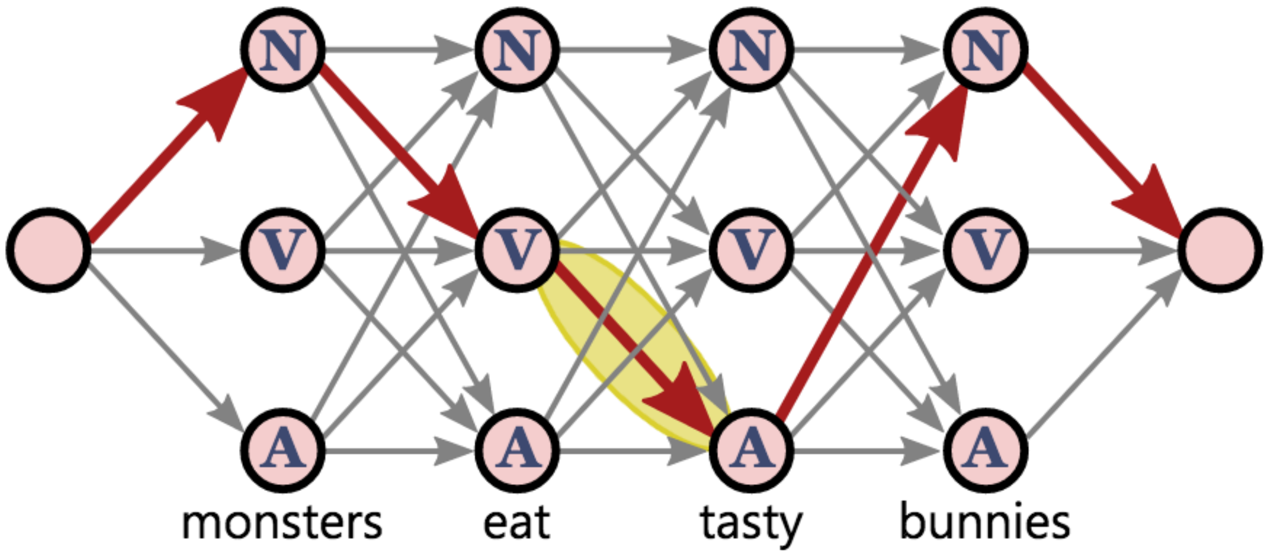
Solving the argmax problem for sequences

$\boldsymbol{x} =$ " monsters eat tasty bunnies "

$\boldsymbol{y} =$ noun verb adj noun

- Efficient algorithms possible if the feature function decomposes over the input
- This holds for unary and markov features

Solving the argmax problem for sequences



- Trellis sequence labeling
  - Any path represents a labeling of input sentence
  - Gold standard path in red
  - Each edge receives a weight such that adding weights along the path corresponds to score for input/ouput configuration
- Any max-weight max-weight path algorithm can find the argmax
  - e.g. Viterbi algorithm $O(LK^2)$

POS tagging

CMSC 723 / LING 723 / INST 725

Marine Carpuat