

Part of Speech Tagging

Word Classes: Parts of Speech

- 8 (ish) traditional parts of speech
 - Noun, verb, adjective, preposition, adverb, article, interjection, pronoun, conjunction, etc
 - Called: parts-of-speech, lexical categories, word classes, morphological classes, lexical tags...

POS examples

- N noun *chair, bandwidth, pacing*
- V verb *study, debate, munch*
- ADJ adjective *purple, tall, ridiculous*
- ADV adverb *unfortunately, slowly*
- P preposition *of, by, to*
- PRO pronoun *I, me, mine*
- DET determiner *the, a, that, those*

POS Tagging

- The process of assigning a part-of-speech or lexical class marker to each word in a collection.

WORD

tag

the

DET

boy

N

put

V

the

DET

keys

N

on

P

the

DET

table

N

Why is POS Tagging Useful?

- First step of a vast number of practical tasks
- Speech synthesis
 - How to pronounce "lead"?
 - INsult inSULT
 - OBject obJECT
 - OVERflow overFLOW
 - DIScount disCOUNT
 - CONtent conTENT
- Parsing
 - Helpful to know parts of speech before you start parsing
 - Analogy to lex/yacc (flex/bison)
- Information extraction
 - Finding names, relations, etc.
- Machine Translation

Open and Closed Classes

- Closed class: a small(ish) fixed membership
 - Usually **function words** (short common words which play a role in grammar)
- Open class: new ones can be created all the time
 - English has 4: Nouns, Verbs, Adjectives, Adverbs
 - Many languages have these 4, but not all!
 - Nouns are typically where the bulk of the action is with respect to new items

Open Class Words

- Nouns

- Proper nouns (Bansthali, Aditya, Manpreet Kaur)
 - English capitalizes these.
- Common nouns (the rest).
- Count nouns and mass nouns
 - Count: have plurals, get counted: goat/goats, one goat, two goats
 - Mass: don't get counted (snow, salt, communism) (*two snows)

- Adverbs: tend to modify things

- Unfortunately, John walked home extremely slowly yesterday
- Directional/locative adverbs (here, home, downhill)
- Degree adverbs (extremely, very, somewhat)
- Manner adverbs (slowly, slinkily, delicately)

- Verbs

- In English, have morphological affixes (eat/eats/eaten)

Closed Class Words

Examples:

- prepositions: *on, under, over, ...*
- particles: *up, down, on, off, ...*
- determiners: *a, an, the, ...*
- pronouns: *she, who, I, ..*
- conjunctions: *and, but, or, ...*
- auxiliary verbs: *can, may should, ...*
- numerals: *one, two, three, third, ...*

Prepositions from CELEX

of	540,085	through	14,964	worth	1,563	pace	12
in	331,235	after	13,670	toward	1,390	nigh	9
for	142,421	between	13,275	plus	750	re	4
to	125,691	under	9,525	till	686	mid	3
with	124,965	per	6,515	amongst	525	o'er	2
on	109,129	among	5,090	via	351	but	0
at	100,169	within	5,030	amid	222	ere	0
by	77,794	towards	4,700	underneath	164	less	0
from	74,843	above	3,056	versus	113	midst	0
about	38,428	near	2,026	amidst	67	o'	0
than	20,210	off	1,695	sans	20	thru	0
over	18,071	past	1,575	circa	14	vice	0

English Particles

aboard	aside	besides	forward(s)	opposite	through
about	astray	between	home	out	throughout
above	away	beyond	in	outside	together
across	back	by	inside	over	under
ahead	before	close	instead	overhead	underneath
alongside	behind	down	near	past	up
apart	below	east, etc.	off	round	within
around	beneath	eastward(s),etc.	on	since	without

Conjunctions

and	514,946	yet	5,040	considering	174	forasmuch as	0
that	134,773	since	4,843	lest	131	however	0
but	96,889	where	3,952	albeit	104	immediately	0
or	76,563	nor	3,078	providing	96	in as far as	0
as	54,608	once	2,826	whereupon	85	in so far as	0
if	53,917	unless	2,205	seeing	63	inasmuch as	0
when	37,975	why	1,333	directly	26	insomuch as	0
because	23,626	now	1,290	ere	12	insomuch that	0
so	12,933	neither	1,120	notwithstanding	3	like	0
before	10,720	whenever	913	according as	0	neither nor	0
though	10,329	whereas	867	as if	0	now that	0
than	9,511	except	864	as long as	0	only	0
while	8,144	till	686	as though	0	provided that	0
after	7,042	provided	594	both and	0	providing that	0
whether	5,978	whilst	351	but that	0	seeing as	0
for	5,935	suppose	281	but then	0	seeing as how	0
although	5,424	cos	188	but then again	0	seeing that	0
until	5,072	supposing	185	either or	0	without	0

POS Tagging: Choosing a Tagset

- There are many potential distinctions we can draw leading to potentially large tagsets
- To do POS tagging, we need to choose a standard set of tags to work with
- Could pick very coarse tagsets
 - N, V, Adj, Adv.
- More commonly used set is the finer grained, “Penn TreeBank tagset”, 45 tags
 - PRP\$, WRB, WP\$, VBG
- Even more fine-grained tagsets exist

Penn TreeBank POS Tagset

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

Using the Penn Tagset

- The/DT grand/JJ jury/NN commmented/VBD on/IN a/DT number/NN of/IN other/JJ topics/NNS ./.
- Prepositions and subordinating conjunctions marked IN (“although/IN I/PRP..”)
- Except the preposition/complementizer “to” is just marked “TO”.

POS Tagging

- Words often have more than one POS: *back*
 - The *back* door = JJ
 - On my *back* = NN
 - Win the voters *back* = RB
 - Promised to *back* the bill = VB
- The POS tagging problem is to determine the POS tag for a particular instance of a word.

How Hard is POS Tagging? Measuring Ambiguity

		87-tag Original Brown	45-tag Treebank Brown
Unambiguous (1 tag)		44,019	38,857
Ambiguous (2–7 tags)		5,490	8844
Details:	2 tags	4,967	6,731
	3 tags	411	1621
	4 tags	91	357
	5 tags	17	90
	6 tags	2 (<i>well, beat</i>)	32
	7 tags	2 (<i>still, down</i>)	6 (<i>well, set, round, open, fit, down</i>)
	8 tags		4 (<i>'s, half, back, a</i>)
	9 tags		3 (<i>that, more, in</i>)

Methods for POS Tagging

1. Rule-based tagging
2. Stochastic
 1. Probabilistic sequence models
 - HMM (Hidden Markov Model) tagging
 - MEMMs (Maximum Entropy Markov Models)
3. Hybrid tagging
 - Partially Rule-based, Partially Probabilistic
 - aka Transformation Based Learning

Hidden Markov Model Tagging

- Using an HMM to do POS tagging is a special case of *Bayesian inference*
 - Foundational work in computational linguistics
 - Bledsoe 1959: OCR
 - Mosteller and Wallace 1964: authorship identification
- It is also related to the “noisy channel” model that’s the basis for ASR, OCR and MT

POS Tagging as Sequence Classification

- We are given a sentence (an “observation” or “sequence of observations”)
 - *Secretariat is expected to race tomorrow*
- What is the best sequence of tags that corresponds to this sequence of observations?
- Probabilistic view:
 - Consider all possible sequences of tags
 - Out of this universe of sequences, choose the tag sequence which is most probable given the observation sequence of n words $w_1 \dots w_n$.

Getting to HMMs

- We want, out of all sequences of n tags $t_1 \dots t_n$ the single tag sequence such that $P(t_1 \dots t_n | w_1 \dots w_n)$ is highest.

$$\hat{t}_1^n = \operatorname{argmax}_{t_1^n} P(t_1^n | w_1^n)$$

- Hat ^ means “our estimate of the best one”
- $\operatorname{Argmax}_x f(x)$ means “the x such that $f(x)$ is maximized”

Getting to HMMs

- This equation is guaranteed to give us the best tag sequence

$$\hat{t}_1^n = \operatorname{argmax}_{t_1^n} P(t_1^n | w_1^n)$$

- But how to make it operational? How to compute this value?
- Intuition of Bayesian inference:
 - Use Bayes rule to transform this equation into a set of other probabilities that are easier to compute

Using Bayes Rule

$$P(x|y) = \frac{P(y|x)P(x)}{P(y)}$$

$$\hat{t}_1^n = \operatorname{argmax}_{t_1^n} \frac{P(w_1^n | t_1^n) P(t_1^n)}{P(w_1^n)}$$

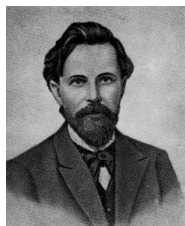
$$\hat{t}_1^n = \operatorname{argmax}_{t_1^n} P(w_1^n | t_1^n) P(t_1^n)$$

Likelihood and Prior



$$\hat{t}_1^n = \operatorname{argmax}_{t_1^n} \overbrace{P(w_1^n | t_1^n)}^{\text{likelihood}} \overbrace{P(t_1^n)}^{\text{prior}}$$

$$P(w_1^n | t_1^n) \approx \prod_{i=1}^n P(w_i | t_i)$$



$$P(t_1^n) \approx \prod_{i=1}^n P(t_i | t_{i-1})$$

$$\hat{t}_1^n = \operatorname{argmax}_{t_1^n} P(t_1^n | w_1^n) \approx \operatorname{argmax}_{t_1^n} \prod_{i=1}^n P(w_i | t_i) P(t_i | t_{i-1})$$

Two Kinds of Probabilities

- Tag transition probabilities $p(t_i|t_{i-1})$
 - Determiners likely to precede adjs and nouns
 - That/DT flight/NN
 - The/DT yellow/JJ hat/NN
 - So we expect $P(NN|DT)$ and $P(JJ|DT)$ to be high
 - But $P(DT|JJ)$ to be:
 - Compute $P(NN|DT)$ by counting in a labeled corpus:

$$P(t_i|t_{i-1}) = \frac{C(t_{i-1}, t_i)}{C(t_{i-1})}$$

$$P(NN|DT) = \frac{C(DT, NN)}{C(DT)} = \frac{56,509}{116,454} = .49$$

Two Kinds of Probabilities

- Word likelihood probabilities $p(w_i|t_i)$
 - VBZ (3sg Pres verb) likely to be “is”
 - Compute $P(\text{is}|\text{VBZ})$ by counting in a labeled corpus:

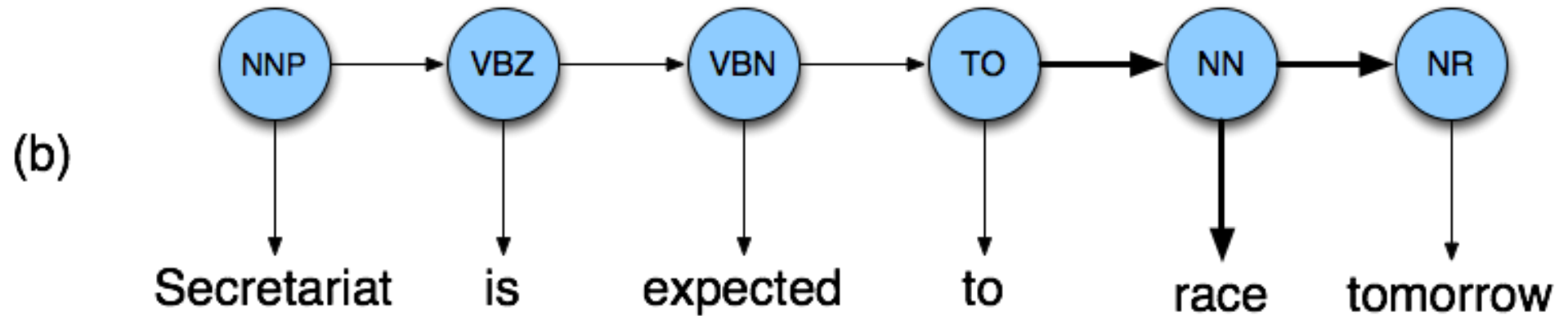
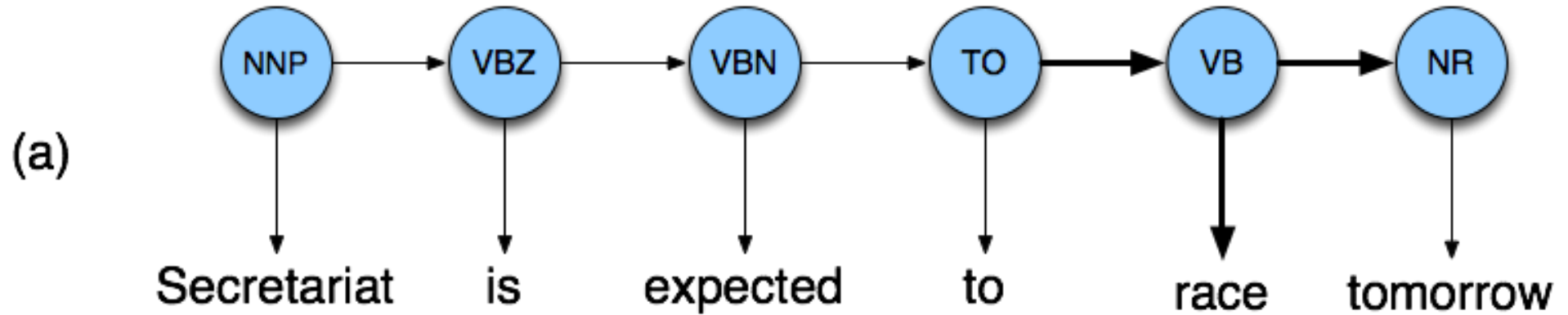
$$P(w_i|t_i) = \frac{C(t_i, w_i)}{C(t_i)}$$

$$P(\text{is}|\text{VBZ}) = \frac{C(\text{VBZ}, \text{is})}{C(\text{VBZ})} = \frac{10,073}{21,627} = .47$$

Example: The Verb “race”

- Secretariat/**NNP** is/**VBZ** expected/**VBN** to/**TO** **race**/**VB**
tomorrow/**NR**
- People/**NNS** continue/**VB** to/**TO** inquire/**VB** the/**DT** reason/**NN**
for/**IN** the/**DT** **race**/**NN** for/**IN** outer/**JJ** space/**NN**
- How do we pick the right tag?

Disambiguating “race”



Example

- $P(\text{NN}|\text{TO}) = .00047$
- $P(\text{VB}|\text{TO}) = .83$
- $P(\text{race}|\text{NN}) = .00057$
- $P(\text{race}|\text{VB}) = .00012$
- $P(\text{NR}|\text{VB}) = .0027$
- $P(\text{NR}|\text{NN}) = .0012$
- $P(\text{VB}|\text{TO})P(\text{NR}|\text{VB})P(\text{race}|\text{VB}) = .00000027$
- $P(\text{NN}|\text{TO})P(\text{NR}|\text{NN})P(\text{race}|\text{NN}) = .00000000032$
- So we (correctly) choose the verb reading,

Hidden Markov Models

- What we've described with these two kinds of probabilities is a Hidden Markov Model (HMM)
- This is a *generative* model.
 - There is a *hidden* underlying generator of observable events
 - The hidden generator can be modeled as a set of states
 - We want to infer the underlying state sequence from the observed events