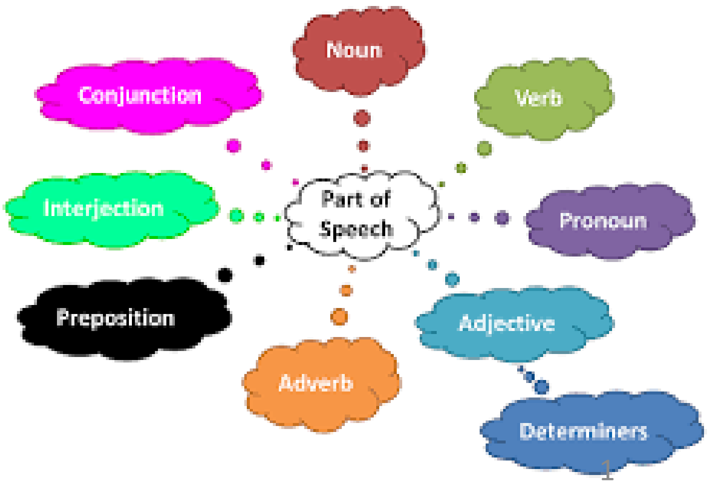
Part of Speech (POS) Tagging

(Natural Language Processing)

Randil Pushpananda
rpn@ucsc.cmb.ac.lk



Written and Spoken Forms

Meaning	Written Form	Spoken Form	Name	Examples
Different	Different	Different	Different	cat,dog
Different	Different	Same	Homophones	bear , bare
Different	Same	Different	Homographs	bass- fish, bass- music
Different	Same	Same	Homonyms	bank
Same	Different	Different	Synonyms	high, tall
Same	Different	Same	Orthographic Variants	labor, labour
Same	Same	Different	Phonetic Variants	either /iy dh er/ , /ay dh er/
Same	Same	Same	Identical	-

Examples

- I *left* my phone on the *left* side of the room.
- The committee *chair* sat in the center *chair*.
- She will *park* the car so we can walk in the *park*.

Examples

- I *left* my phone on the *left* side of the room.

left *adjective* — on the side of your body that is towards the west when you are facing north

left *adverb* — on or to the left side

- *Turn left at the intersection.*

left *verb* — PAST TENSE, PAST PARTICIPLE OF **leave**

03/14/1999 (AFP)… the extremist Harkatul Jihad group, reportedly backed by Saudi dissident Osama bin Laden ...

03/14/1999 (AFP)… the extremist Harkatul Jihad group, reportedly backed by Saudi dissident Osama bin Laden …

Tags:

Determiners - DT, Adjectives - JJ, Proper Nouns - NNP, Nouns - NN, Adverbs - RB, Verbs - VBD, Prepositions - IN

What is Part-Of-Speech Tagging?

- The general purpose of a part-of-speech tagger is to associate each word in a text with its correct lexical-syntactic category (represented by a tag)

03/14/1999 (AFP)… the extremist Harkatul Jihad group, reportedly backed by Saudi dissident Osama bin Laden …

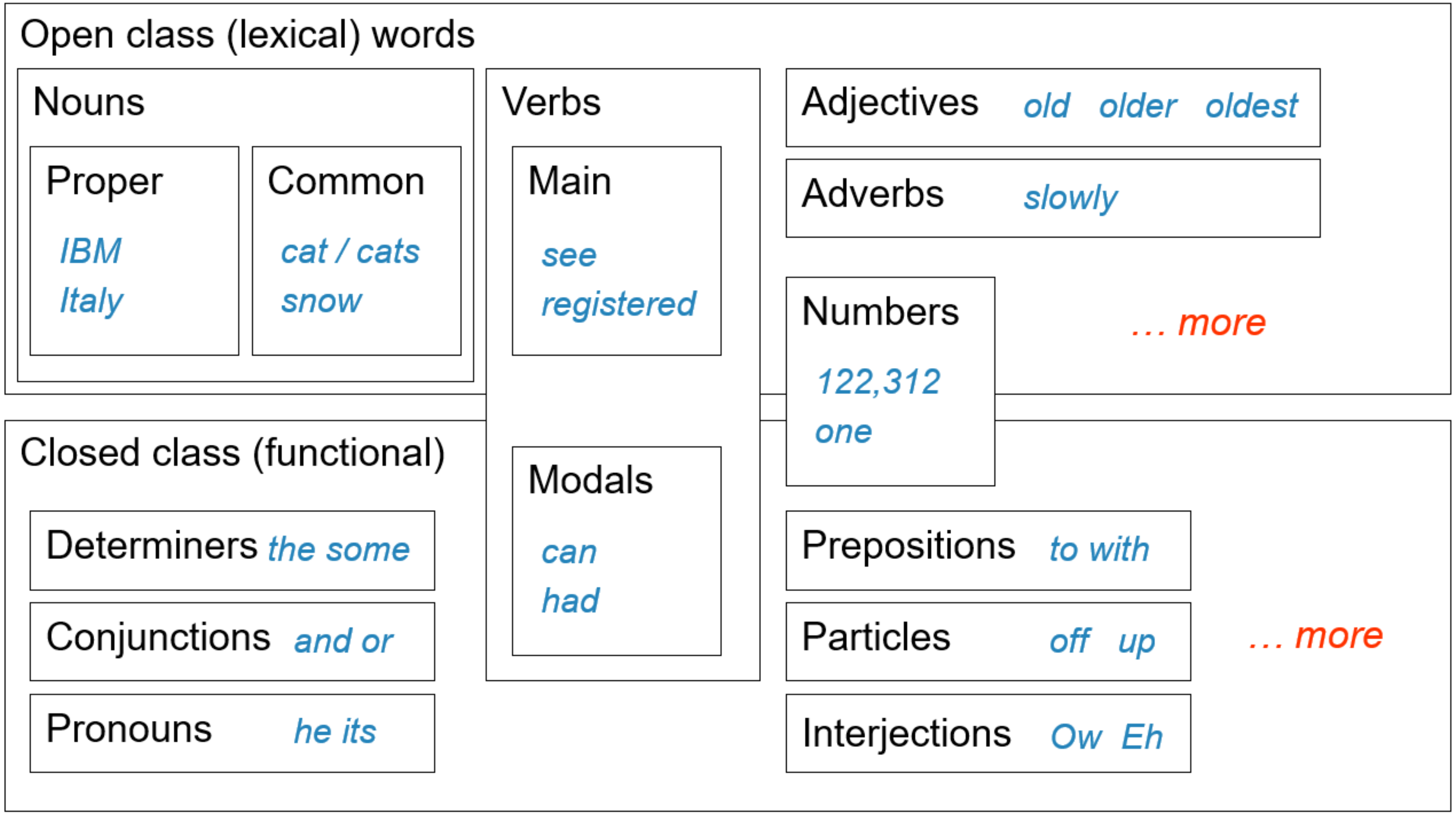
… the|DT extremist|JJ Harkatul|NNP Jihad|NNP group|NN ,|, reportedly|RB backed|VBD by|IN Saudi|NNP dissident|NN Osama|NNP bin|NN Laden|NNP …

What are Parts-of-Speech?

- Approximately 8 traditional basic words classes, sometimes called lexical classes or types
- These are the ones taught in grade school grammar
  - N noun *chair, bandwidth, pacing*
  - V verb *study, debate, munch*
  - ADJ adjective *purple, tall, ridiculous (includes articles)*
  - ADV adverb *unfortunately, slowly*
  - P preposition *of, by, to*
  - CON conjunction *and, but*
  - PRO pronoun *I, me, mine*
  - INT interjection *um*

Open and Closed Classes

- We may want to make more distinctions than 8 classes:



Open vs. Closed classes

- Open vs. Closed classes
  - Closed:
    - determiners: ***a, an, the***
    - pronouns: ***she, he, I***
    - prepositions: ***on, under, over, near, by, …***
    - Why "closed"?
  - Open:
    - Nouns, Verbs, Adjectives, Adverbs.

POS Tagging

	Plays	well	with	others
• Input:	Plays	well	with	others
• Ambiguity:	NNS/VBZ	UH/JJ/NN/RB	IN	NNS
• Output:	Plays/VBZ	well/RB	with/IN	others/NNS

Penn Treebank POS tags

- Uses:
  - Text-to-speech (how do we pronounce "lead"?)
  - Can write regexps like (Det) Adj* N+ over the output for phrases, etc.
  - As input to or to speed up a full parser
  - If you know the tag, you can back off to it in other tasks

POS Tagging

- Words often have more than one POS: *back*
  - The *back* door
  - On my *back*
  - Win the voters *back*
  - Promised to *back* the bill

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

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.

Why is Part-Of-Speech Tagging Hard?

- Words may be ambiguous in different ways:
  - A word may have multiple meanings as the same part-of-speech
    - *Bank* – noun, financial institute
    - *Bank* – noun, edge of a river

- A word may function as multiple parts-of-speech

- a *round* table: **adjective**
- a *round* of applause: **noun**
- to *round* out your interests: **verb**
- to work the year *round*: **adverb**

Word Class Ambiguity (in the Brown Corpus)

- Unambiguous (1 tag): 35,340
- Ambiguous (2-7 tags): 4,100

2 tags	3,760
3 tags	264
4 tags	61
5 tags	12
6 tags	2
7 tags	1

(Derose, 1988)

Possible Tag Sets for English

- Kucera & Brown (Brown Corpus) – 87 POS tags
- C5 (British National Corpus) – 61 POS tags
  - Tagged by Lancaster's UCREL project
- Penn Treebank – 45 POS tags
  - Most widely used of the tag sets today

Possible Tag Sets for English

- Kucera & Brown (Brown Corpus) – 87 POS tags
- C5 (British National Corpus) – 61 POS tags
  - Tagged by Lancaster's UCREL project
- Penn Treebank – 45 POS tags
  - Most widely used of the tag sets today

NN Common noun, neutral for number (e.g. *sheep, co*
'Our *team* has/have lost'.]

NN1 Singular common noun (e.g. *bath, powder, disgr*

NN2 Plural common noun (e.g. *baths, powders, sister*

NNJ Human organization noun (e.g. *council, orchestra*
initial capital, in names of private or public organizations

NNJ2 Plural human organization noun (e.g. *councils, o*

NNL Locative noun, neutral for number (e.g. *Is.* as an a

NNL1 Singular locative noun (e.g. *island, street*). [The

NNL2 Plural locative noun (e.g. *islands, streets*) [Agai

NNO Numeral noun, neutral for number (cf. MC above

NNO2 Plural numeral noun (e.g. *hundreds, thousands*

```
JJ      Adjective
JJR     Adjective, comparative
JJS     Adjective, superlative
```

```
NN      Noun, singular or mass
NNS     Noun, plural
NNP     Proper noun, singular
NNPS    Proper noun, plural
```

Word Classes:

Tag	Description	Example	Tag	Description	Example
CC	coordin. conjunction	*and, but, or*	SYM	symbol	+,%, &
CD	cardinal number	*one, two*	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, sing.	*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*			

Figure 9.1 Penn Treebank part-of-speech tags (including punctuation).

Word Classes:

29 Tags

(2007 – 2010)

UCSC SINHALA TAGSET

TAG	Description	Examples
CC	Conjunction	හා (ha:) , හෝ (ho:) , සමග (saməgə) , එක්ක (ekkə) , නමුත් (namut) , යැයි(yæyi) , දැයි(dæyi) , එහෙත් (ehet) , ඒත් (e:t)
DET	Determiner	මේ (me:) ,ඒ (e:) , මෙම (memə) , බොහෝ (boho:) , එක් (ek) , සමහර (samaharə) , ඔය (oyə) , යම් (yam) , අර (arə) , ඇතැම් (ætæm) , සියලු (siyəlu) , ටික (ţikə) , සියලුම (siyəlumə) , මුළු (muḷu) , කවර (kawərə) , හැම (hæmə) , අන් (an) , කිසි (kisi) , නව (tawə) , වෙන (wenə) , කිසිම (kisimə) , වෙනත්(wenat) , අනෙක් (anek) , වෙන් (wen) , කිසිදු (kisidu) , සෑම (sæ:mə) , එම (ema) කිසියම් (kisiyam)
FRW	Foreign Word	Computer, System, IT
JJ	Adjective	රළු (raḷu) , සුමුදු (sumudu) , වැඩි (wæḍi) , හොඳ (Hoďə) , විශේෂ (viʃe:ʂə) , අලුත් (alut) , සුදු (sudu) , අඩු (aḍu) , විශාල (viʃa:lə) , කුඩා (kuḍa:) , ඉහළ (ihəḷə) , දිග (digə) , පුංචි (puɲci) , ලොකු (loku) , අධික (adhikə) , ඇද (æda) , දියුණු (diyuɳu) , සුළු (suḷu) , යට (yaţə) , සුදුසු (sudusu)
JVB	Adjective in Kriya Müla	කීකරු වෙනවා (ki:kəru venəva:) , එකඟ වෙනවා (ekəğə venava:) , අඩු කරනවා (aḍu kərənəva:)
NNF	Common Noun Feminine	නිළියෝ (niḷiyo:) , ඇතින්න (ætinnə) , මව (mawə) , මිය (miyə)
NNM	Common Noun Masculine	මිනිසා (minisa:) , බල්ලා (balla:) , ශිෂ්‍යයෝ (ʃiʂyayo:) , එළුවන් (eḷuwan) , පියා (piya:) , මිනිසා (minisa:) , මිනිස් (minis) , නිලධාරීන් (nilədha:ri:n) , භික්ෂූන් (bhikʂu:n) , දරුවන් (daruwan) , මිනිසුන් (minisun) , ජනතා (janəta:) , රජ (rajə)

Word Classes:

29 Tags

(2007 – 2010)

Bengali, Hindi, Tamil

UCSC SINHALA TAGSET

TAG	Description	Examples
CC	Conjunction	හා (ha:) , හෝ (ho:) , සමග (saməgə) , එක්ක (ekkə) , නමුත් (namut) , යැයි(yæyi) , දැයි(dæyi) , එහෙත් (ehet) , ඒත් (e:t)
DET	Determiner	මේ (me:) ,ඒ (e:) , මෙම (memə) , බොහෝ (boho:) , එක් (ek) , සමහර (samaharə) , ඔය (oyə) , යම් (yam) , අර (arə) , ඇතැම් (ætæm) , සියලු (siyəlu) , ටික (ţikə) , සියලුම (siyəlumə) , මුළු (muḷu) , කවර (kawərə) , හැම (hæmə) , අන් (an) , කිසි (kisi) , තව (tawə) , වෙන (wenə) , කිසිම (kisimə) , වෙනත්(wenat) , අනෙක් (anek) , වෙන් (wen) , කිසිදු (kisidu) , සෑම (sæ:mə) , එම (ema) කිසියම් (kisiyam)
FRW	Foreign Word	Computer, System, IT
JJ	Adjective	රළු (raḷu) , සුමුදු (sumudu) , වැඩි (wæḍi) , හොඳ (Hoďə) , විශේෂ (viʃe:ʂə) , අලුත් (alut) , සුදු (sudu) , අඩු (aḍu) , විශාල (viʃa:lə) , කුඩා (kuḍa:) , ඉහළ (ihəḷə) , දිග (digə) , පුංචි (puŋci) , ලොකු (loku) , අධික (adhikə) , ඇද (æda) , දියුණු (diyuṇu) , සුළු (suḷu) , යට (yaţə) , සුදුසු (sudusu)
JVB	Adjective in Kriya Müla	කීකරු වෙනවා (ki:kəru venəva:) , එකඟ වෙනවා (ekəğə venava:) , අඩු කරනවා (aḍu kərənəva:)
NNF	Common Noun Feminine	නිළියෝ (niḷiyo:) , ඇතින්න (ætinnə) , මව (mawə) , මිය (miyə)
NNM	Common Noun Masculine	මිනිසා (minisa:) , බල්ලා (balla:) , ශිෂ්‍යයෝ (ʃiʂyayo:) , එළුවන් (eḷuwan) , පියා (piya:) , මිනිසා (minisa:) , මිනිස් (minis) , නිලධාරීන් (nilədha:ri:n) , භික්ෂූන් (bhikʂu:n) , දරුවන් (daruwan) , මිනිසුන් (minisun) , ජනතා (janəta:) ,

Sources of information

- What are the main sources of information for POS tagging?
  - Knowledge of neighboring words
    - saw
    - NN
    - VB(D)

Sources of information

- What are the main sources of information for POS tagging?
  - Knowledge of neighboring words
    - Bill saw that man yesterday
    - NNP NN DT NN NN
    - VB VB(D) IN VB NN
  - Knowledge of word probabilities
    - *man* is rarely used as a verb….
- The latter proves the most useful, but the former also helps

More and Better Features → Feature-based tagger

- Can do surprisingly well just looking at a word by itself:
  - Word — the: *the* → DT
  - Lowercased word — Importantly: *importantly* → RB
  - Prefixes — unfathomable: *un-* → JJ
  - Suffixes — Importantly: *-ly* → RB
  - Capitalization — Meridian: *CAP* → NNP
  - Word shapes — 35-year: *d-x* → JJ

Overview of Approaches

- Rule-based Approach
  - Simple and doesn't require a tagged corpus, but not as accurate as other approaches
- Stochastic Approach
  - Refers to any approach which incorporates frequencies or probabilities
  - Requires a tagged corpus to learn frequencies
  - N-gram taggers and Naïve Bayes taggers
  - Hidden Markov Model (HMM) taggers
  - . . .
- Other Issues: unknown words and evaluation

Rule-Based Tagging

- Uses a dictionary that gives possible tags for words
- Basic algorithm
  - Assign all possible tags to words
  - Remove tags according to set of rules of type:
  - Example rule:
    - if word+1 is an adj, adv, or quantifier and the following is a sentence boundary and word-1 is not a verb like “consider” then eliminate non-adv else eliminate adv.
  - Typically more than 1000 hand-written rules, but may be machine-learned
- This approach not in serious use

N-gram Approach

- N-gram approach to probabilistic POS tagging:
  - calculates the probability of a given sequence of tags occurring for a sequence of words
  - the best tag for a given word is determined by the (already calculated) probability that it occurs with the n previous tags
  - may be bi-gram, tri-gram, etc
    - $word_{n-1}$ … word-2 word-1 word
    - tagn-1 … tag-2 tag-1 ??

N-gram Tagging

- Initialize a tagger by learning probabilities from a tagged corpus

wordn-1	…	word-2	word-1	word
tagn-1	…	tag-2	tag-1	??

  - Probability that the sequence … tag-2 tag-1 given ..word-2 word-1 gives tag XX
  - Note that initial sequences will include a start marker as part of the sequence
- Use the tagger to tag word sequences (usually of length 2-3) with unknown tags
  - Sequence through the words:
  - To determine the POS tag for the next word, use the previous n-1 tags and the word to look up probabilities and use the highest probability tag

Need Longer Sequence Classification

- A more comprehensive approach to tagging considers the entire sequence of words
  - Secretariat is expected to race tomorrow
- What is the best sequence of tags which 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 w1…wn.

Road to HMMs

- We want, out of all sequences of n tags t1…tn the single tag sequence such that P(t1…tn|w1…wn) is highest.
  - i.e. the probability of the tag sequence t1…tn given the word sequence w1…wn

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

- Hat ^ means "our estimate of the best one"
- Argmax f(x) means "the x such that f(x) is maximized"
  - i.e. find the tag sequence that maximizes the probability

Road to HMMs

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

- This equation is guaranteed to give us the best tag sequence
- But how to make it operational? How to compute this value?
- Intuition of Bayesian classification:
  - Use Bayes rule to transform into a set of other probabilities that are easier to compute

Thomas Bayes 1701 - 1761

Using Bayes Rule

- Bayes rule:
$$P(x|y) = \frac{P(y|x)P(x)}{P(y)}$$
- Apply Bayes Rule:
$$\hat{t}_1^n = \underset{t_1^n}{\operatorname{argmax}} \frac{P(w_1^n|t_1^n)P(t_1^n)}{P(w_1^n)}$$
- Note that this is using the conditional probability, given a tag, what is the most likely word with that tag.
  - Eliminate denominator as it is the same for every sequence

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

Likelihood and Prior

- Further simplify

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

- Likelihood: assume that the probability of the word depends only on its tag

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

- Prior: use the bigram assumption that the tag only depends on the previous tag

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

So, the best tag sequence,

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

Two Sets of Probabilities (1)

- Tag transition probabilities p(ti|ti-1) (priors)
  - 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
  - Compute P(NN|DT) by counting in a labeled corpus:

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

Count of DT NN sequence

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

Two Sets of Probabilities (2)

- Word likelihood probabilities p(wi|ti)
  - VBZ (3sg Pres verb) likely to be "is"
  - Compute P(is|VBZ) by counting in a labeled corpus:

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

Count of 'is' tagged with VBZ

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

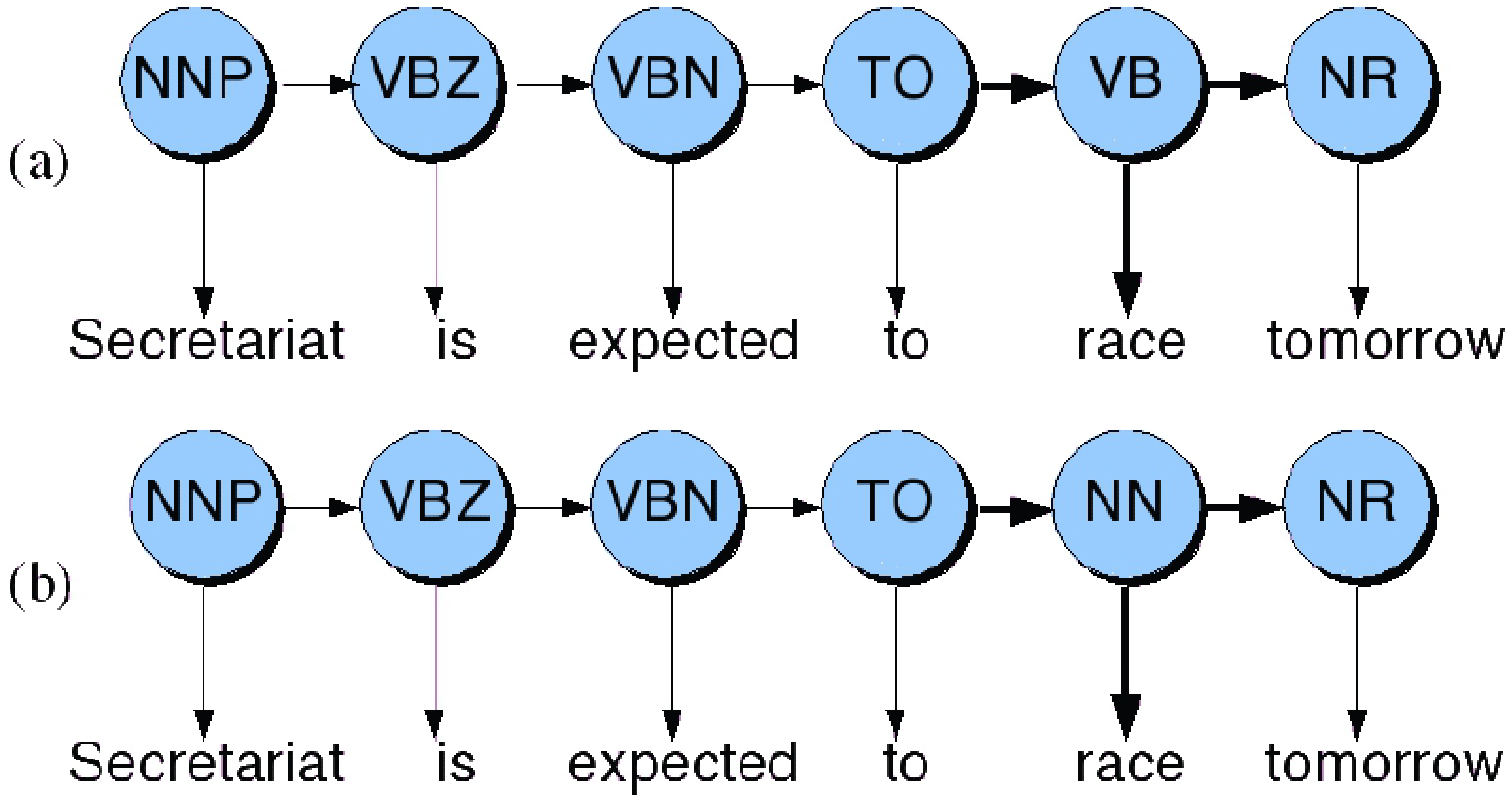
An Example: the verb “race”

- 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"

- Which tag sequence is most likely?



Disambiguating "race"

- *The equations only differ in "to race tomorrow"*
- $P(NN|TO) = .00047$
- $P(VB|TO) = .83$

The tag transition probabilities P(NN|TO) and P(VB|TO)

- $P(race|NN) = .00057$
- $P(race|VB) = .00012$

Lexical likelihoods from the Brown corpus for 'race' given a POS tag NN or VB.

- $P(NR|VB) = .0027$
- $P(NR|NN) = .0012$

Tag sequence probability for the likelihood of an adverb occurring given the previous tag verb or noun

- $P(VB|TO)P(NR|VB)P(race|VB) = .00000027$
- $P(NN|TO)P(NR|NN)P(race|NN)=.00000000032$
- *So we (correctly) choose the verb tag.*

The “A” matrix for the POS HMM

- Example of tag transition (prior) probabilities represented in a matrix, usually called the A matrix in an HMM:
  - The probability that NNP follows <s> is .027, …

	NNP	MD	VB	JJ	NN	RB	DT
$<s>$	0.2767	0.0006	0.0031	0.0453	0.0449	0.0510	0.2026
NNP	0.3777	0.0110	0.0009	0.0084	0.0584	0.0090	0.0025
MD	0.0008	0.0002	0.7968	0.0005	0.0008	0.1698	0.0041
VB	0.0322	0.0005	0.0050	0.0837	0.0615	0.0514	0.2231
JJ	0.0366	0.0004	0.0001	0.0733	0.4509	0.0036	0.0036
NN	0.0096	0.0176	0.0014	0.0086	0.1216	0.0177	0.0068
RB	0.0068	0.0102	0.1011	0.1012	0.0120	0.0728	0.0479
DT	0.1147	0.0021	0.0002	0.2157	0.4744	0.0102	0.0017

Figure 9.5 The A transition probabilities $P(t_i|t_{i-1})$ computed from the WSJ corpus without smoothing. Rows are labeled with the conditioning event; thus $P(VB|MD)$ is 0.7968.

The B matrix for the POS HMM

- Word likelihood probabilities are represented in a matrix, where for each tag, we show the probability that a word has that tag

	Janet	**will**	**back**	**the**	**bill**
NNP	0.000032	0	0	0.000048	0
MD	0	0.308431	0	0	0
VB	0	0.000028	0.000672	0	0.000028
JJ	0	0	0.000340	0.000097	0
NN	0	0.000200	0.000223	0.000006	0.002337
RB	0	0	0.010446	0	0
DT	0	0	0	0.506099	0

Figure 9.6 Observation likelihoods *B* computed from the WSJ corpus without smoothing.

Using HMMs for POS tagging

- From the tagged corpus, create a tagger by computing the two matrices of probabilities, A and B
  - Straightforward for bigram HMM
  - For higher-order HMMs, efficiently compute matrix by the forward-backward algorithm
- To apply the HMM tagger to unseen text, we must find the best sequence of transitions
  - Given a sequence of words, find the sequence of states (POS tags) with the highest probabilities along the path
  - This task is sometimes called "decoding"
  - Use the Viterbi algorithm

Evaluation: Is our POS tagger any good?

- Answer: we use a manually tagged corpus, which we will call the "Gold Standard"
  - We run our POS tagger on the gold standard and compare its predicted tags with the gold tags
  - We compute the accuracy (and other evaluation measures)
- Important: 100% is impossible even for human annotators.
  - We estimate humans can do POS tagging at about 98% accuracy.
  - Some tagging decisions are very subtle and hard to do:
    - Mrs/NNP Shaefer/NNP never/RB got/VBD around/RP to/TO joining/VBG
    - All/DT we/PRP gotta/VBN do/VB is/VBZ go/VB around/IN the/DT corner/NN
    - Chateau/NNP Petrus/NNP costs/VBZ around/RB 250/CD
- The "Gold Standard" will have human mistakes; humans are subject to fatigue, etc.

How can we improve our tagger?

- What are the main sources of information for our HMM POS tagger?
  - Knowledge of tags of neighboring words
  - Knowledge of word tag probabilities
    - *man* is rarely used as a verb….
- The latter proves the most useful, but the former also helps
- Unknown words can be a problem because we don't have this information
- And we are not including information about the features of the words

Features of words

- Can do surprisingly well just looking at a word by itself:

– Word	the: the → DT (determiner)
– Lowercased word	Importantly: importantly → RB (adverb)
– Prefixes	unfathomable: un- → JJ (adjective)
– Suffixes	Importantly: -ly → RB tangential: -al → JJ
– Capitalization	Meridian: CAP → NNP (proper noun)
– Word shapes	35-year: d-x → JJ

- These properties can include information about the previous or the next word(s)
  - The word *be* appears to the left pretty → JJ
- But not information about tags of the previous or next words, unlike HMM

Feature-based Classifiers

- A feature-based classifier is an algorithm that will take a word and assign a POS tag based on features of the word in its context in the sentence.
- Many algorithms are used, just to name a few
  - Naïve Bayes
  - Maximum Entropy (MaxEnt)
  - Support Vector Machines (SVM)

Conclusions

- Part of Speech tagging is a doable task with high performance results
- Contributes to many practical, real-world NLP applications and is now used as a pre-processing module in most systems
- Computational techniques learned at this level can be applied to NLP tasks at higher levels of language processing

Questions!

- Solve the equation according to the sentence;

"I am planning to visit New Delhi to attend Analytics Vidhya Delhi Hackathon".

A = (# of words with Noun as the part of speech tag)
B = (# of words with Verb as the part of speech tag)
C = (# of words with frequency count greater than one)

- **What are the correct values of A, B, and C?**

1. 5, 5, 2
2. 5, 5, 0
3. 7, 5, 1
4. 7, 4, 2
5. 6, 4, 3

Answer:

- Nouns: I, New, Delhi, Analytics, Vidhya, Delhi, Hackathon (7)
- Verbs: am, planning, visit, attend (4)
- Words with frequency counts > 1: to, Delhi (2)

Deep Learning for POS Tagging

- Deep learning approaches have shown great success in various natural language processing tasks, including POS tagging
- Deep Learning Approaches:
  - Recurrent Neural Networks (RNNs)
  - Transformer Models

RNNs for POS Tagging

- Recurrent Neural Networks, particularly Long Short-Term Memory (LSTM) and Gated Recurrent Unit (GRU) networks, have been widely used for POS tagging.
- RNNs are designed to process sequential data by maintaining a hidden state that captures information from previous inputs.
- In POS tagging, an RNN can take a sequence of words as input and predict the corresponding POS tags.
- The hidden state of the RNN captures the contextual information necessary for accurate tagging.
- By training the RNN on annotated data, it can learn to predict the most probable POS tag for each word in a sentence.

Transformer for POS Tagging

- Transformer models, such as the well-known BERT (Bidirectional Encoder Representations from Transformers) and its variants, have revolutionized various NLP tasks, including POS tagging.
- Transformers use self-attention mechanisms to capture contextual dependencies across words in a sentence.
- They can effectively model long-range dependencies and capture the interactions between words.
- In POS tagging, a pre-trained transformer model can be fine-tuned on labeled data to predict the POS tags for input sentences.
- Transformers have shown remarkable performance on POS tagging tasks, often surpassing traditional machine learning methods.