Computational Linguistics

5. Hidden Markov Models and Part-Of-Speech Tagging

Xiaojing Bai

Tsinghua University

https://bxjthu.github.io/CompLing

Recap: conditional probability and Joint probability

Conditional probability is the probability of event A given that the occurrence of event B, written as $P(A|B)$.

Joint probability is the probability of two events in conjunction, i.e. the probability of both events together, written as $P(A \cap B)$ or $P(A, B)$.

If A and B are independent, i.e. knowing the outcome of A does not change the probability of B, or $P(B|A) = P(B)$, then $P(A \cap B) = P(A)P(B)$.

If A and B are not independent, e.g. knowing the outcome of A does change the probability of B, or $P(B|A) \neq P(B)$, then $P(A \cap B) = P(A)P(B|A)$.

Recap

Probabilities of bigrams

$$P(w_n|w_{n-1}) = \frac{C(w_{n-1}w_n)}{\sum_w C(w_{n-1}w)} = \frac{C(w_{n-1}w_n)}{C(w_{n-1})}$$

Probabilities of sequences

$$\begin{aligned} P(w_1^n) &= P(w_1)P(w_2|w_1)P(w_3|w_1^2)\dots P(w_n|w_1^{n-1}) \\ &= \prod_{k=1}^{n} P(w_k|w_1^{k-1}) \\ &\approx \prod_{k=1}^{n} P(w_k|w_{k-1}) \end{aligned}$$

$\mathbf{w_{n-1}}$	$\mathbf{w_n}$	**count**	**probability**
<s>	welcome	3	0.60
<s>	what	1	0.20
<s>	you	1	0.20
a	welcome	2	1.00
are	a	1	1.00
back	</s>	1	1.00
home	</s>	2	1.00
sight	</s>	1	1.00
welcome	home	2	0.40
welcome	back	1	0.20
welcome	sight	1	0.20
welcome	</s>	1	0.20
what	a	1	1.00
you	are	1	1.00

The bigram counts and probabilities for the toy corpus

Recap

Probabilities of trigrams

$$P(w_n|w_{n-2}w_{n-1}) = \frac{C(w_{n-2}w_{n-1}w_n)}{C(w_{n-2}w_{n-1})}$$

Probabilities of sequences

$$\begin{aligned} P(w_1^n) &= P(w_1)P(w_2|w_1)P(w_3|w_1^2)\ldots P(w_n|w_1^{n-1}) \\ &= \prod_{k=1}^{n} P(w_k|w_1^{k-1}) \\ &\approx \prod_{k=1}^{n} P(w_k|w_{k-1}w_{k-2}) \end{aligned}$$

```
<s>welcome back</s>

<s>welcome home</s>

<s>you are a welcome sight</s>

<s>what a welcome</s>

<s>what a lovely day</s>

<s>you are so lovely</s>
```

w_{n-2}	w_{n-1}	w_n	count	probability
you	are	so	1	0.5
what	a	welcome	1	0.5
are	so	lovely	1	1

Recap

- Power of n-grams
- Dependence of n-grams on their training sets
- Evaluation of language models
- N-grams in NLP applications

At the end of this session you will

- learn the difference between Markov models and hidden Markov models;
- know that hidden Markov models can help parsing on different levels;
- understand the purposes of POS tagging;
- know what a tagset is and how tagsets vary;
- know a rule-based method and a probabilistic method of POS tagging;
- work better with REs in structured programs and handle file i/o well.

The Markov model or the Markov chain

- The Markov assumption
  - the probability of a word depends only on the previous word

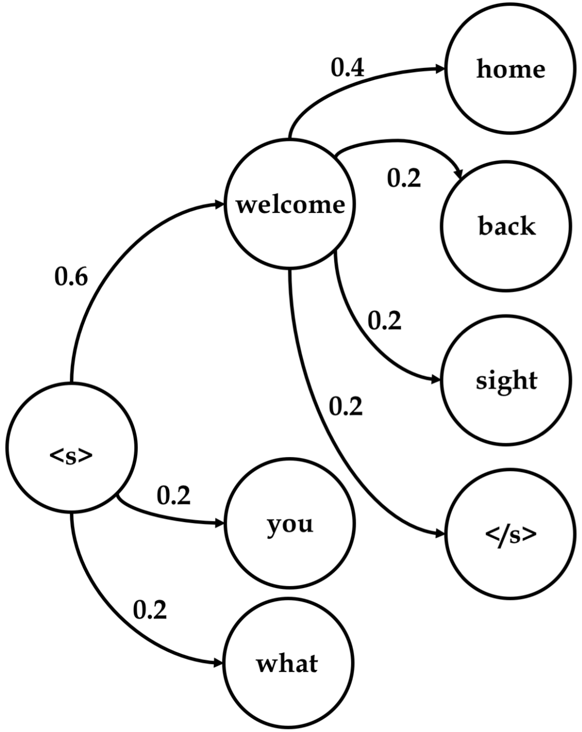
    $$P(q_i|q_1...q_{i-1}) = P(q_i|q_{i-1})$$

- An extension of an FSA: a special case of a weighted FSA
  - the weights being the probabilities
  - the input sequence uniquely determining the states to go through
- Useful for assigning probabilities to unambiguous sequences

The Markov model or the Markov chain

w_{n-1}	w_n	count	probability
<s>	welcome	3	0.60
<s>	what	1	0.20
<s>	you	1	0.20
a	welcome	2	1.00
are	a	1	1.00
back	</s>	1	1.00
home	</s>	2	1.00
sight	</s>	1	1.00
welcome	home	2	0.40
welcome	back	1	0.20
welcome	sight	1	0.20
welcome	</s>	1	0.20
what	a	1	1.00
you	are	1	1.00

The bigram counts and probabilities for the toy corpus



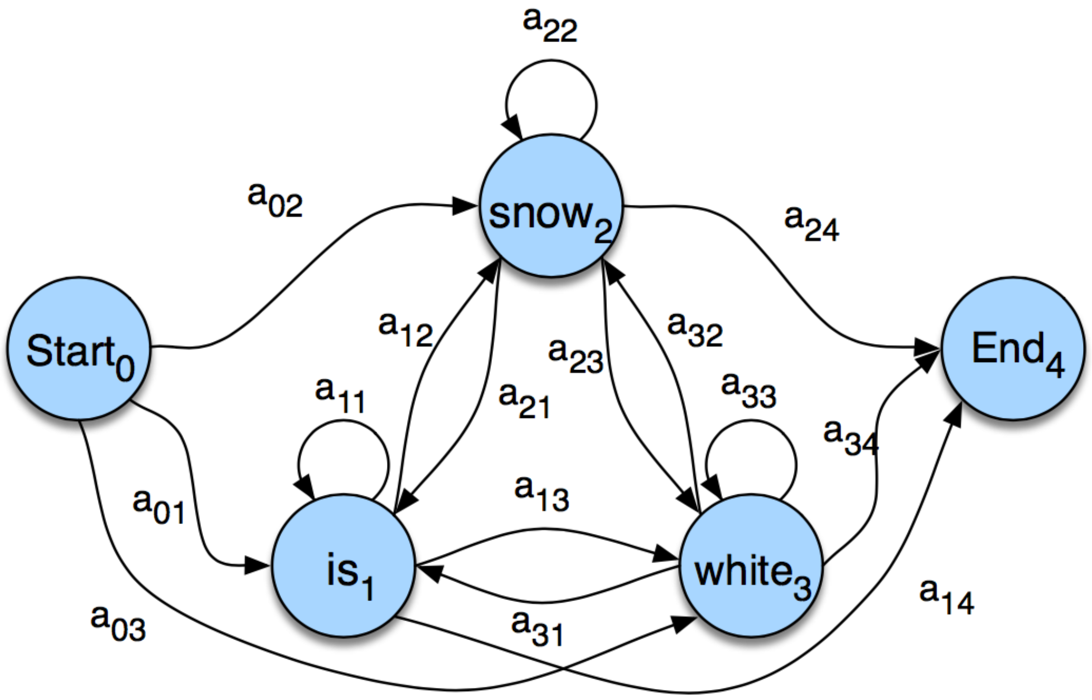
Part of the Markov chain for the toy corpus

The Markov model or the Markov chain

$Q = \{q_1, q_2, \ldots, q_n\}$: a set of n **states**

$A = [a_{ij}]$: a **transition probability matrix** A, each a_{ij} representing the probability of moving from state i to state j, s.t. $\sum_{j=1}^{n} a_{ij} = 1 \;\; \forall i$

$\pi = \{\pi_1, \pi_2, \ldots, \pi_n\}$: an **initial probability distribution** over states, each π_i representing the probability that the Markov chain will start in state i, s.t. $\sum_{i=1}^{n} \pi_i = 1$



A Markov model

Used to compute a probability for a sequence of observable events

A hidden Markov model (HMM)

Used to compute a probability for a sequence of NOT observable events

Example: Jason's ice cream climatology data

	B	C	D	E	F	G
10		p(...|C)	p(...|H)	p(...|START)		
11	p(1|...)	0.7	0.1		If today is cold (C) or	
12	p(2|...)	0.2	0.2		hot (H), how many	
13	p(3|...)	0.1	0.7		cones did I prob. eat?	
14	p(C|...)	0.8	0.1	0.5	If today is cold or hot,	
15	p(H|...)	0.1	0.8	0.5	what will tomorrow	
16	p(STOP|...)	0.1	0.1	0	probably be?	

Figure 2: Initial guesses of parameters.

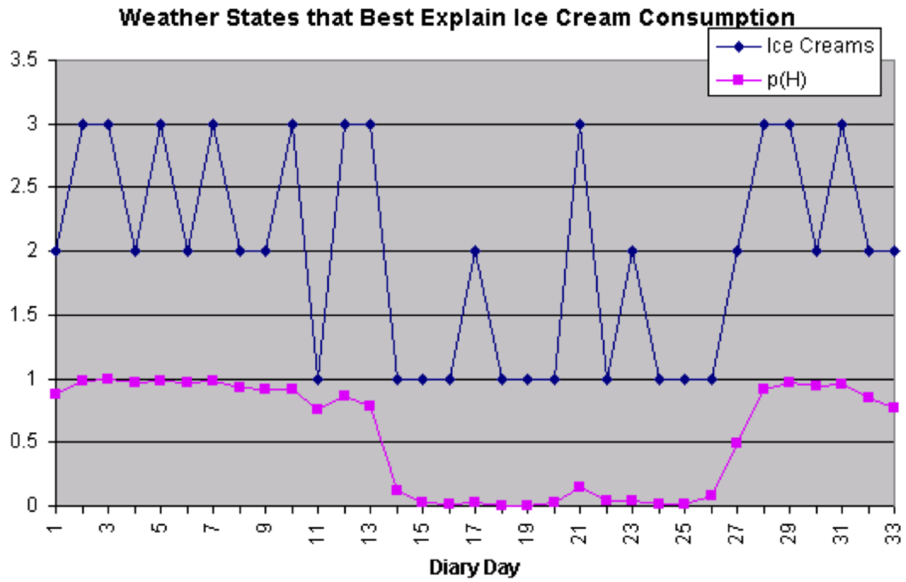
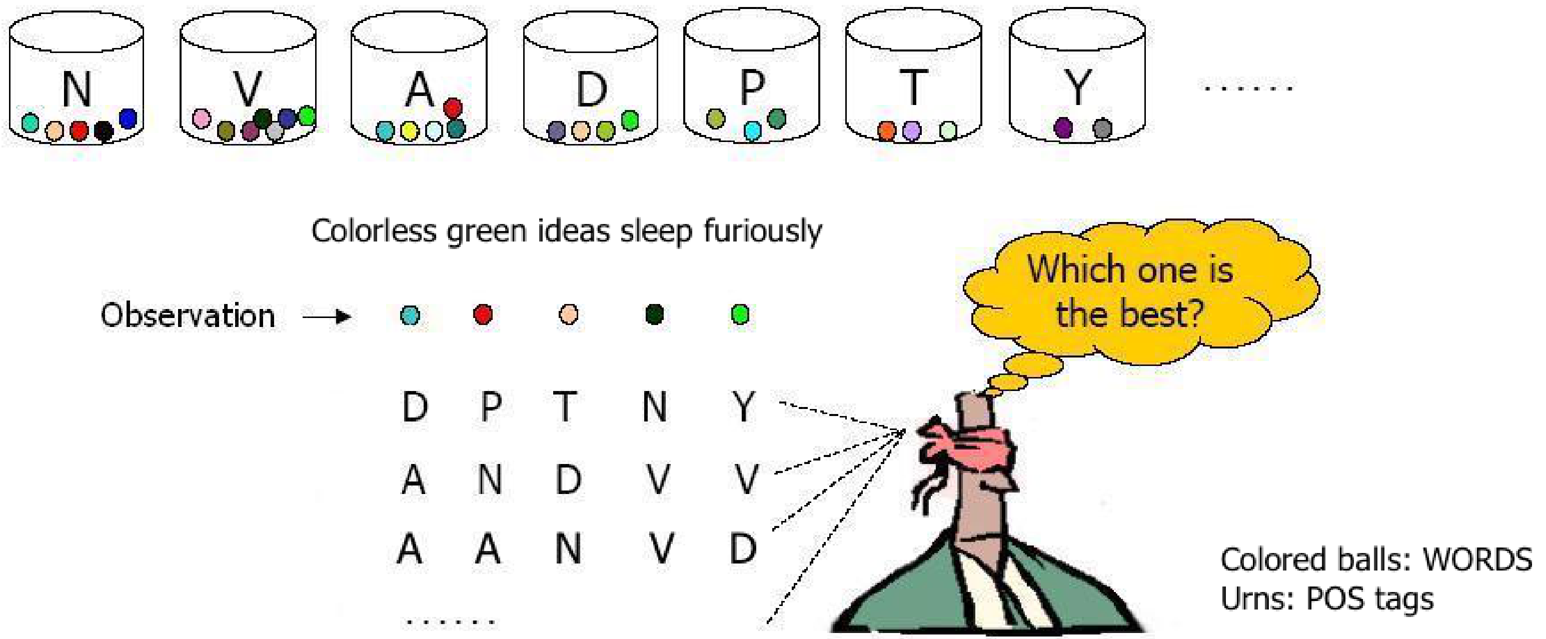


Figure 3: Diary data and reconstructed weather.

HMM: a probabilistic sequence model

Given a sequence of units (words, letters, morphemes, sentences, whatever),
a HMM assigns a label or class to each unit in the sequence,
thus mapping a sequence of observations to a sequence of labels.



The hidden Markov model

$Q = \{q_1, q_2, \ldots, q_n\}$: a set of n **states**

$A = [a_{ij}]$: a **transition probability matrix** A, each a_{ij} representing the probability of moving from state i to state j, s.t. $\sum_{j=1}^{n} a_{ij} = 1 \;\; \forall i$

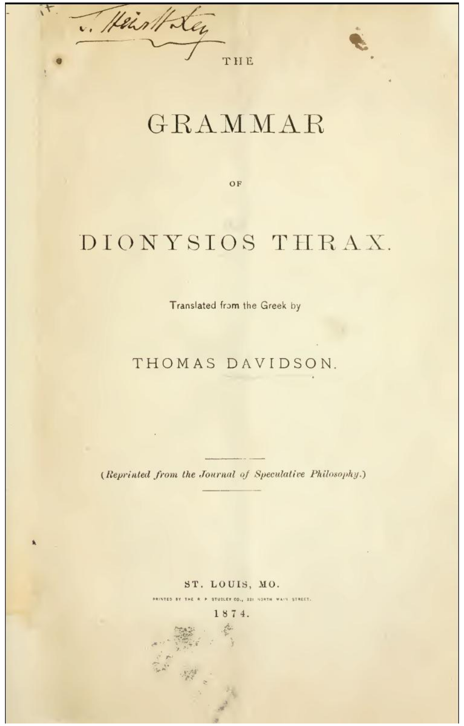
$O = o_1 o_2 \ldots o_T$: a sequence of T **observations**, each one drawn from a vocabulary $V = \{v_1, v_2, \ldots, v_V\}$

$B = b_i(o_t)$: an sequence of **observation probabilities**, each expressing the probability of an observation o_t being generated from a state i

$\pi = \{\pi_1, \pi_2, \ldots, \pi_n\}$: an **initial probability distribution** over states, each π_i representing the probability that the Markov chain will start in state i, s.t. $\sum_{i=1}^{n} \pi_i = 1$

The astonishing durability of POS through two millennia

Terminology: parts-of-speech, word classes, syntactic categories, …



THE

GRAMMAR

OF

DIONYSIOS THRAX.

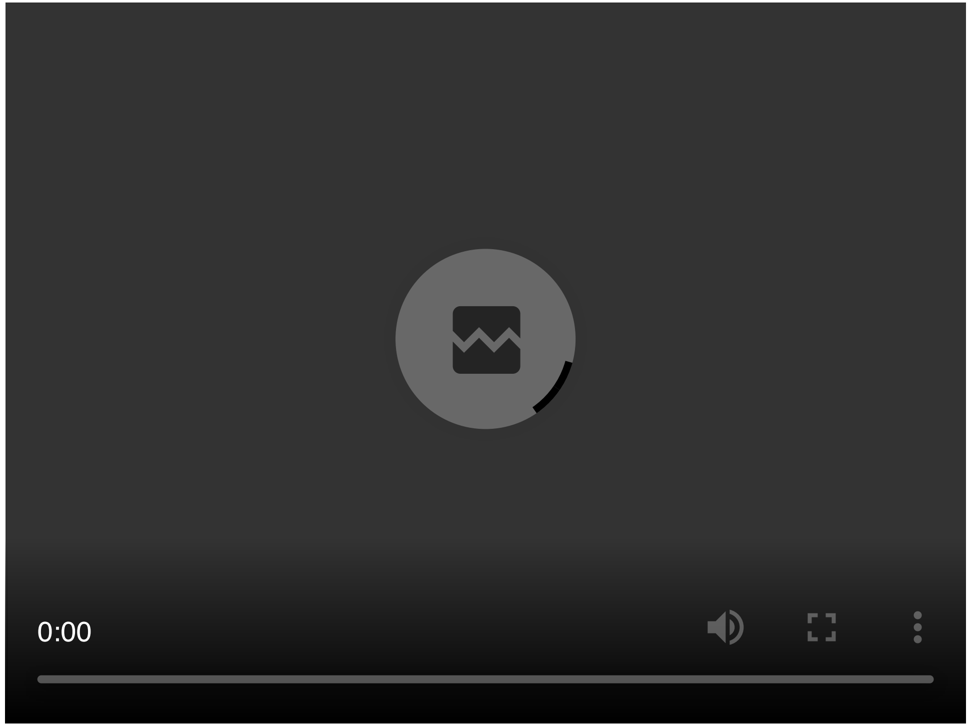
Translated from the Greek by

THOMAS DAVIDSON.

(*Reprinted from the Journal of Speculative Philosophy.*)

ST. LOUIS, MO.

1874.



Why we need to assign parts-of-speech to words?

- Part-of-Speech tagging
    - Input: a sequence of words + a tagset
    - Output: a sequence of tags
- POS features used in
    - Syntactic parsing
    - Information extraction
    - Informational retrieval
    - Automatic summarization
    - Speech synthesis and recognition

Review: English and Chinese Word Classes

Ambiguities in POS tagging

The amount of tag ambiguity for word types in the Brown and the WSJ corpora

		WSJ		Brown	
Types:					
Unambiguous	(1 tag)	44,432	(**86%**)	45,799	(**85%**)
Ambiguous	(2+ tags)	7,025	(**14%**)	8,050	(**15%**)
Tokens:					
Unambiguous	(1 tag)	577,421	(**45%**)	384,349	(**33%**)
Ambiguous	(2+ tags)	711,780	(**55%**)	786,646	(**67%**)

- Differences across the genres
- The most ambiguous frequent words

  that, back, down, put, set

E.g.

earnings growth took a back/JJ seat

a small building in the back/NN

a clear majority of senators back/VBP the bill

Dave began to back/VB toward the door

enable the country to buy back/RP about

debt I was twenty-one back/RB then

Tagged corpora and Tagsets

- POS-tagged corpora as the training and test sets for statistical tagging algorithms and other statistical NLP tasks
- Automatic POS tagger + human annotators hand-correction
- Very commonly used tagsets
    - The 87-tag Brown set
    - The 61-tag CLAWS 5 set
    - The 45-tag Penn Treebank set

The Penn Treebank POS Tagset

- The Brown corpus
- The Wall Street Journal corpus
- The Switchboard corpus

- Tag + slash

E.g.

The/DT grand/JJ jury/NN commented/VBD on/IN a/DT number/NN of/IN other/JJ topics/NNS ./.

There/EX are/VBP 70/CD children/NNS there/RB

Preliminary/JJ findings/NNS were/VBD reported/VBN in/IN today/NN ’s/POS New/NNP England/NNP Journal/NNP of/IN Medicine/NNP ./.

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

Rule-based POS tagging

- A dictionary: to assign each word a list of potential parts-of-speech
- A set of hand-written disambiguation rules: to winnow down this list to a single part-of-speech for each word

E.g.

I consider that odd.

I wouldn't go that far.

Word	POS	Additional POS features
smaller	ADJ	COMPARATIVE
fast	ADV	SUPERLATIVE
that	DET	CENTRAL DEMONSTRATIVE SG
all	DET	PREDETERMINER SG/PL QUANTIFIER
dog's	N	GENITIVE SG
furniture	N	NOMINATIVE SG NOINDEFDETERMINER
one-third	NUM	SG
she	PRON	PERSONAL FEMININE NOMINATIVE SG3
show	V	PRESENT -SG3 VFIN
show	N	NOMINATIVE SG
shown	PCP2	SVOO SVO SV
occurred	PCP2	SV
occurred	V	PAST VFIN SV

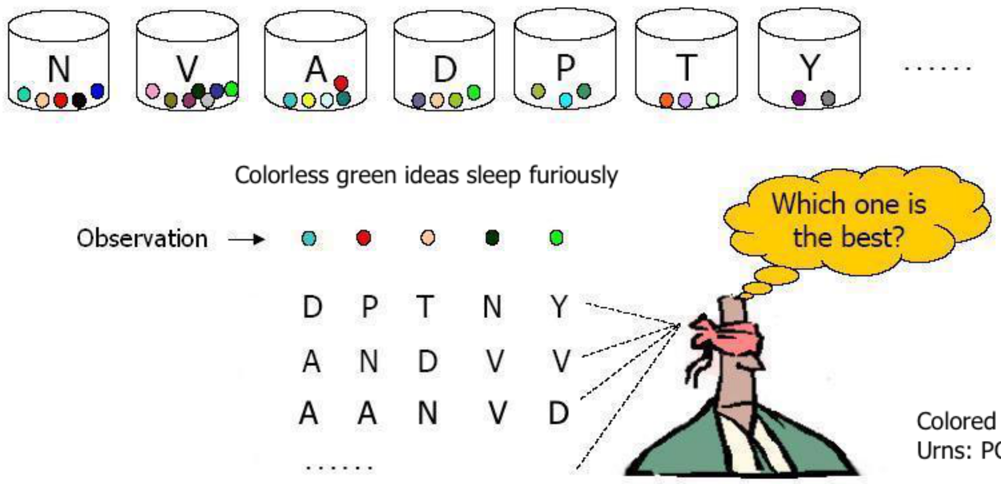
Figure 5.11 Lexical entries in the ENGTWOL lexicon (Voutilainen, 1995; Heikkilä, 1995).

ADVERBIAL-THAT RULE Given input: "that"

```
if
 (+1 A/ADV/QUANT);
          # if next word is adj, adverb, or quantifier
 (+2 SENT-LIM);
          # and following which is a sentence boundary
 (NOT -1 SVOC/A);
          # and the previous word is not a verb
          # like 'consider' which allows adjs as
          # object complements
then eliminate non-ADV tags
else eliminate ADV tag
```

HMM POS tagging: a decoding task

Given as input an HMM $\lambda = (A, B)$ and a sequence of observations $O = o_1 o_2 \ldots o_T$, find the most probable sequence of states $Q = q_1 q_2 q_3 \ldots q_T$.



Colored balls: WORDS
Urns: POS tags

$Q = \{q_1, q_2, \ldots, q_n\}$: a set of n **states**

$A = [a_{ij}]$: a **transition probability matrix** A, each a_{ij} representing the probability of moving from state i to state j, s.t. $\sum_{j=1}^{n} a_{ij} = 1 \;\; \forall i$

$O = o_1 o_2 \ldots o_T$: a sequence of T **observations**, each one drawn from a vocabulary $V = \{v_1, v_2, \ldots, v_V\}$

$B = b_i(o_t)$: an sequence of **observation probabilities**, each expressing the probability of an observation o_t being generated from a state i

$\pi = \{\pi_1, \pi_2, \ldots, \pi_n\}$: an **initial probability distribution** over states, each π_i representing the probability that the Markov chain will start in state i, s.t. $\sum_{i=1}^{n} \pi_i = 1$

Bayes' theorem

Property A = {F,M}

Property B = {FL,CS}

$P(M) = \frac{5}{10} = 0.5 \quad P(F) = \frac{5}{10} = 0.5$

$P(CS) = \frac{4}{10} = 0.4 \quad P(FL) = \frac{6}{10} = 0.6$

$P(CS|M) = \frac{3}{5} = 0.6 \quad P(FL|M) = \frac{2}{5} = 0.4$

$P(CS|F) = \frac{1}{5} = 0.2 \quad P(FL|F) = \frac{4}{5} = 0.8$

$P(M|CS) = \frac{3}{4} = 0.75 \quad P(F|CS) = \frac{1}{4} = 0.25$

$P(M|FL) = \frac{2}{6} = 0.33 \quad P(F|FL) = \frac{4}{6} = 0.66$

Example:
Consider a group of 10 students taking this course: some are male (M) and others female (F); some are enrolled in the Computer Science department (CS) and others in the Foreign Languages department (FL).

Gender	**Dept.**
M	CS
M	CS
M	CS
M	FL
M	FL
F	CS
F	FL
F	FL
F	FL
F	FL

Bayes' theorem

Property A = {F,M}

Property B = {FL,CS}

The interaction between probabilities of the two properties.

$$P(A|B) = \frac{P(B|A)P(A)}{P(B)}$$

Applying Bayes’ theorem to POS tagging:

A = {POS tags in the tagset}

B = {word tokens in the corpus}

Example:
Consider a group of 10 students taking this course: some are male (M) and others female (F); some are enrolled in the Computer Science department (CS) and others in the Foreign Languages department (FL).

Gender	Dept.
M	CS
M	CS
M	CS
M	FL
M	FL
F	CS
F	FL
F	FL
F	FL
F	FL

The basic equation of HMM tagging

The most probable tag sequence given the observation sequence of n words w_1^n:

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

$\hat{t}_1^n$ means 'the estimate of the sequence of n tags'

$\underset{x}{\operatorname{argmax}} P(x)$ means 'the x such that P(x) is maximized'

The basic equation of HMM tagging

The most probable tag sequence given the observation sequence of n words w_1^n:

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

$$\hat{t}_1^n = \underset{t_1^n}{\operatorname{argmax}} \frac{P(w_1^n | t_1^n) P(t_1^n)}{P(w_1^n)} \quad \Leftarrow \text{using the Bayes' rule}$$

$$\hat{t}_1^n = \underset{t_1^n}{\operatorname{argmax}} P(w_1^n | t_1^n) P(t_1^n) \quad \Leftarrow \text{dropping the denominator } P(w_1^n)$$

$$P(w_1^n | t_1^n) \approx \prod_{i=1}^n P(w_i | t_i) \quad P(w_i | t_i) = \frac{\text{Frequency of } w_i \text{ tagged as } t_i \text{ in the training corpus}}{\text{Frequency of } t_i \text{ in the training corpus}}$$

$$P(t_1^n) \approx \prod_{i=1}^n P(t_i | t_{i-1}) \qquad P(t_i | t_{i-1}) = \frac{\text{Frequency of } t_i \text{ after } t_{i-1} \text{ in the training corpus}}{\text{Frequency of } t_{i-1} \text{ in the training corpus}}$$

The basic equation of HMM tagging

The most probable tag sequence given the observation sequence of n words w_1^n:

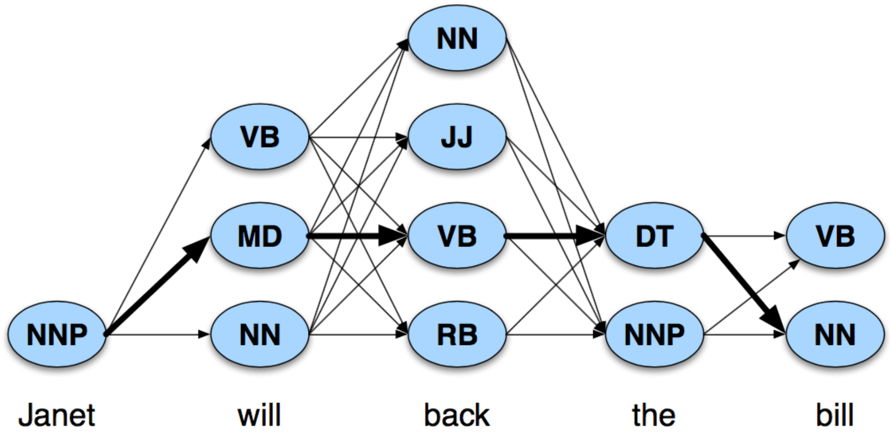
$$\hat{t}_1^n = \underset{t_1^n}{\operatorname{argmax}} P(w_1^n|t_1^n)P(t_1^n) \approx \underset{t_1^n}{\operatorname{argmax}} \prod_{i=1}^{n} P(w_i|t_i)P(t_i|t_{i-1})$$

$$P(w_i|t_i) = \frac{\text{Frequency of } w_i \text{ tagged as } t_i \text{ in the training corpus}}{\text{Frequency of } t_i \text{ in the training corpus}}$$

$$P(t_i|t_{i-1}) = \frac{\text{Frequency of } t_i \text{ after } t_{i-1} \text{ in the training corpus}}{\text{Frequency of } t_{i-1} \text{ in the training corpus}}$$

HMM POS tagging: an example

E.g. Janet will back the bill



Janet/NNP will/MD back/VB the/DT bill/NN

	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	$\boxed{0.7968}$	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 10.5 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.

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

Figure 10.6 Observation likelihoods B computed from the WSJ corpus without smoothing.

At the end of this session you will

- learn the difference between Markov models and hidden Markov models;
- know that hidden Markov models can help parsing on different levels;
- understand the purposes of POS tagging;
- know what a tagset is and how tagsets vary;
- know a rule-based method and a probabilistic method of POS tagging;
- work better with REs in structured programs and handle file i/o well.

Homework

- Read/review (Quiz 5 on Oct. 31, 2018)
  - J+M 8 (8.1 - 8.4; 8.7)

  Question: How might POS features be used in information extraction, informational retrieval, automatic summarization, speech synthesis and recognition, or other NLP applications you can think of?

- Read and Practice
  - http://www.nltk.org/book/ch05.html

Next session

Formal Grammars and Syntactic Parsing