POS Tagging

Background

- **Part of speech**:
  - Noun, verb, pronoun, preposition, adverb, conjunction, particle, and article
- Recent lists of **POS** (also know as **word classes**, **morphological class**, or **lexical tags**) have much larger numbers of word classes.
  - 45 for Penn Treebank
  - 87 for the Brown corpus, and
  - 146 for the C7 tagset
- The significance of the POS for language processing is that it gives a significant amount of information about the word and its neighbors.
- POS can be used in stemming for IR, since
  - Knowing a word's POS can help tell us which morphological affixes it can take.
  - They can help an IR application by helping select out nouns or other important words from a document.

English Word Classes

- Give a more complete definition of the classes of POS.
  - Traditionally, the definition of POS has been based on morphological and syntactic function.
  - While, it has tendencies toward semantic coherence (e.g., nouns describe people, places, or things and adjectives describe properties), this is not necessarily the case.
- Two broad subcategories of POS:
  1. **Closed class**
  2. **Open class**

English Word Classes

1. **Closed class**
   - Having relatively fixed membership, e.g., prepositions
   - **Function words:**
     - Grammatical words like *of*, *and*, or *you*, which tend to be very short, occur frequently, and play an important role in grammar.
2. **Open class**
   - Four major open classes occurring in the languages of the world: **nouns**, **verbs**, **adjectives**, and **adverbs.**
     - Many languages have no adjectives, e.g., the native American language Lakhota, and Chinese

Tagsets for English

- There are a small number of popular tagsets for English, many of which evolved from the 87-tag tagset used for the Brown corpus.
  - Three commonly used
    - **The small 45-tag Penn Treebank tagset**
    - The medium-sized 61 tag C5 tageset used by the Lancaster UCREL project's CLAWS tagger to tag the British National Corpus, and
    - The larger 146-tag C7 tagset

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 ")
PP	Personal pronoun	*I, you, he*	(	Left parenthesis	([, (, {, <)
PP$	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 POS tags

The grand jury commented on a number of other topic
DT JJ NN

Tagsets for English

The grand jury commented on a number of other topics.

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

There are 70 children there

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

Although preliminary findings were reported more than a year ago, the latest results appear in today's New English Journal of Medicine

Although/IN preliminary/JJ findings/NNS were/VBD reported/VBN more/RBR than/IN a/DT year/NN ago/NN ,/, the/DT latest/JJS results/NNS appear/VBP in/IN today/NN 's/POS New/NNP English/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, 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 ")
PP	Personal pronoun	*I, you, he*	(	Left parenthesis	([, (, {, <)
PP$	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*			

Hidden Markov Model

that

- Give me that pen. ← determin

- Who gave you that? ← demonstrative pronoun

- I did not know that she was married. → Complex.

- It's a song that my mother taught me → relative pronoun

This game?! I cant want

that long.

adv

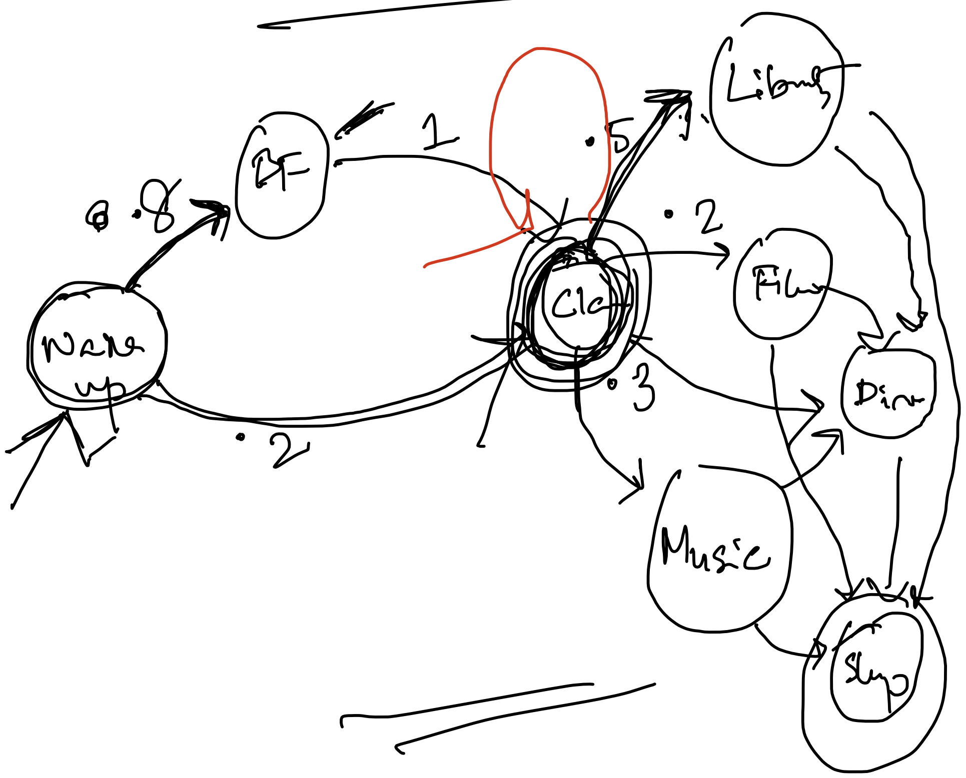
Sequence labelling task

Hidden Markov Models

Outline

- **Markov Chains**
- **Hidden Markov Models**
- **Three Algorithms for HMMs**
  - **The Forward Algorithm**
  - **The Viterbi Algorithm**
- **Applications:**
  - **The Ice Cream Task**
  - **Part of Speech Tagging**

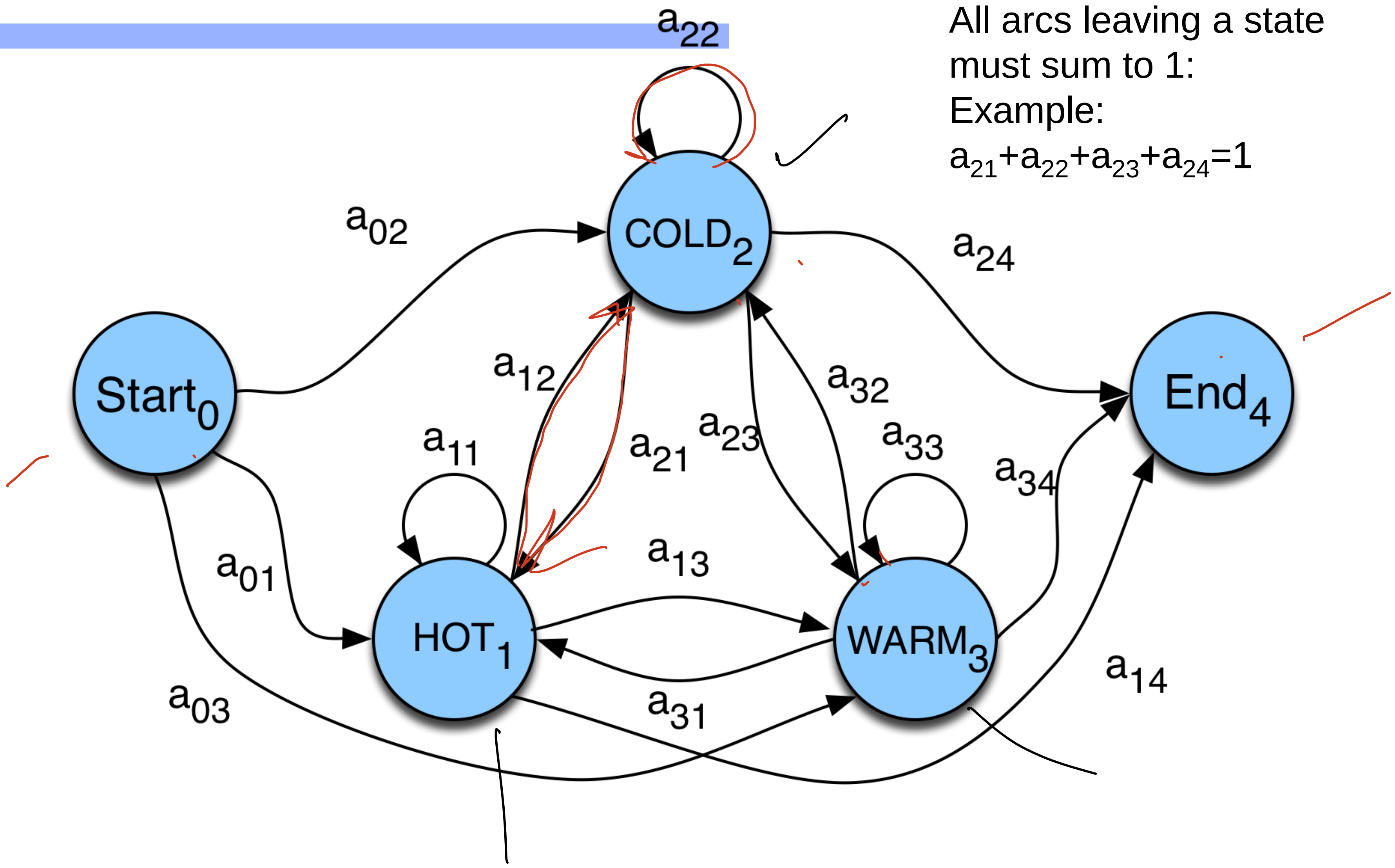
Finite State Machine



Definitions

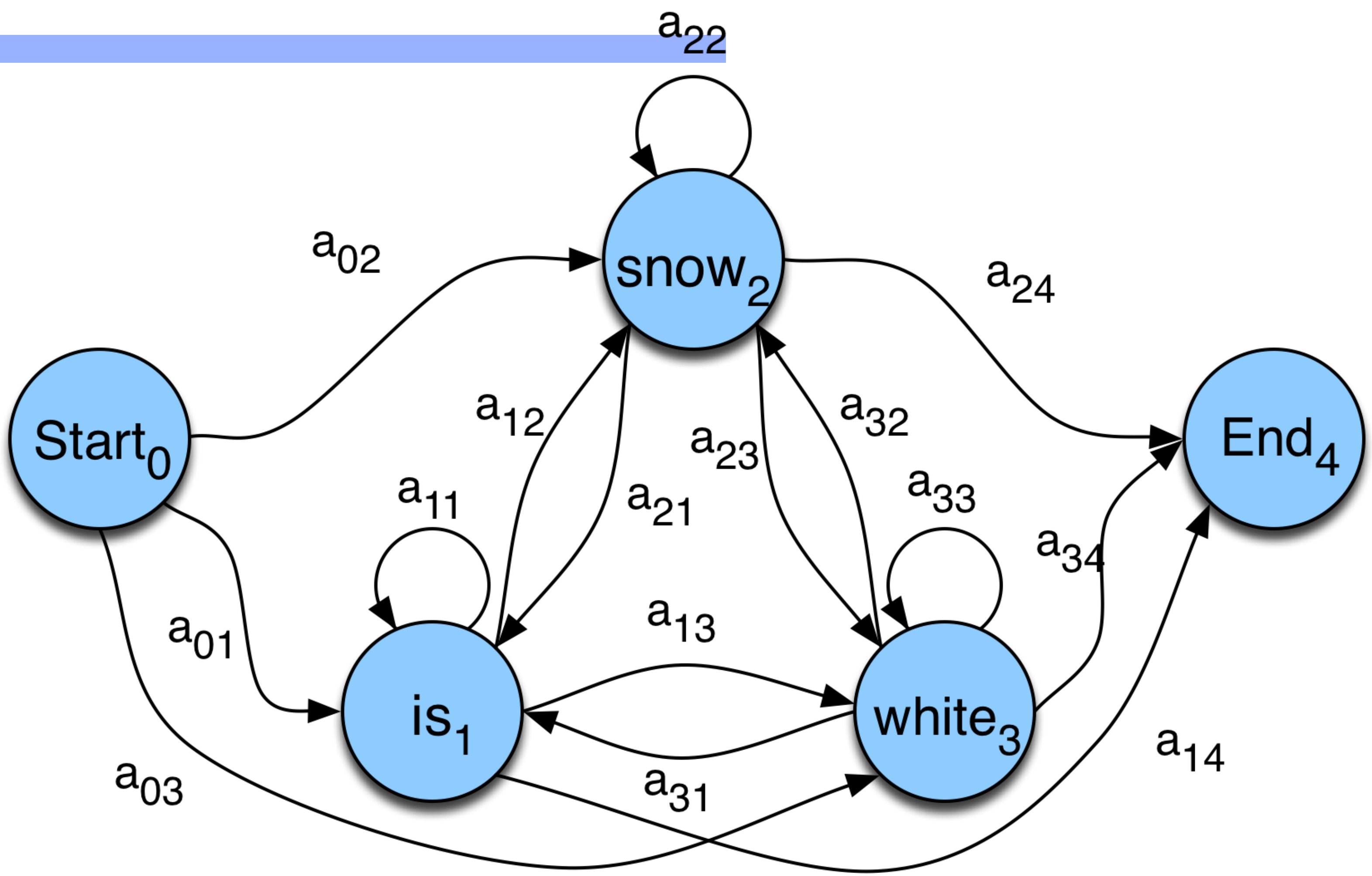
- A **weighted finite-state automaton**
  - An FSA with probabilities on the arcs
  - The sum of the probabilities leaving any arc <u>must sum to one</u>
- A **Markov chain (or observable Markov Model)**
  - a special case of a WFST in which the input sequence uniquely determines which states the automaton will go through
- **Markov chains can't represent inherently ambiguous problems**
  - Useful for assigning probabilities to unambiguous sequences

Markov chain for weather



All arcs leaving a state must sum to 1:
Example:
$a_{21}+a_{22}+a_{23}+a_{24}=1$

Markov chain for words



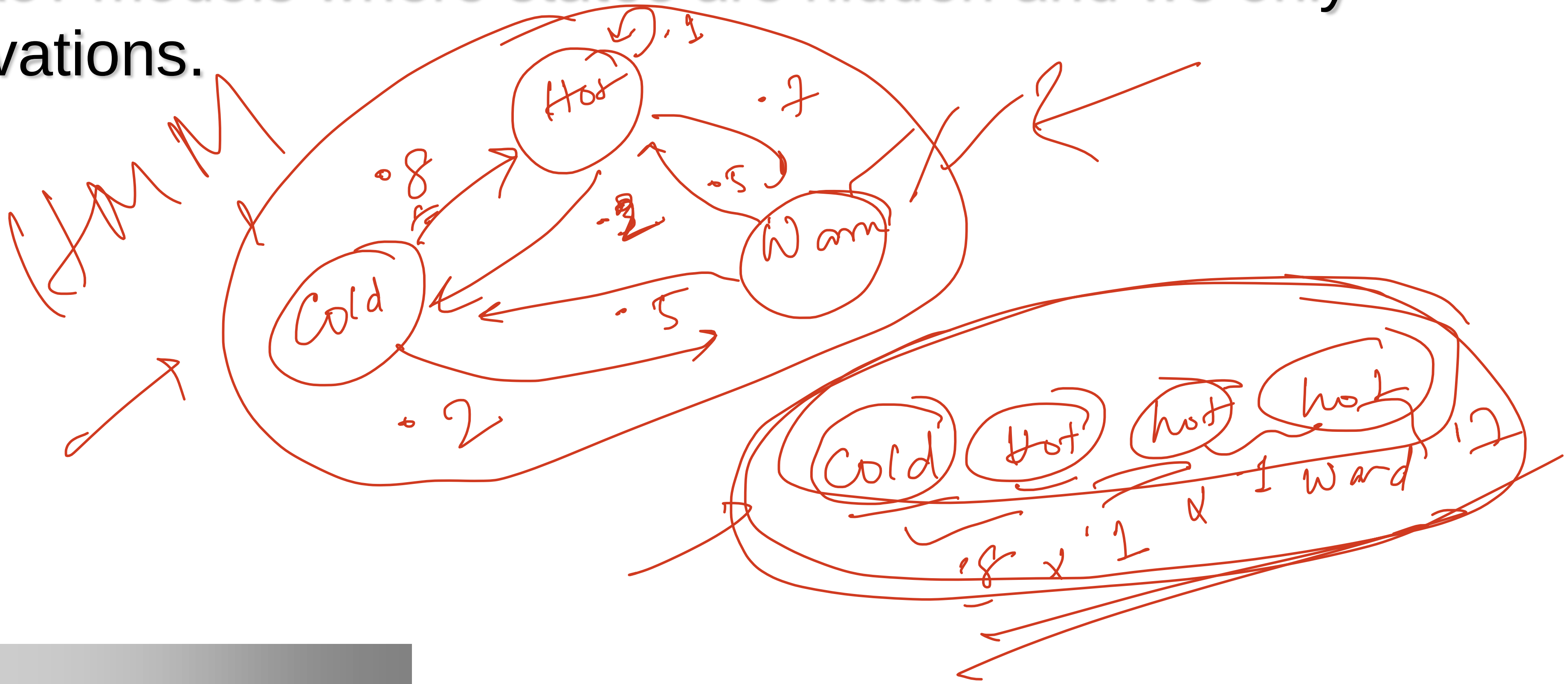
Markov chain = "First-order observable Markov Model"

- Markov Assumption:

  Current state only depends on previous state

$$P(q_i \mid q_1 \ldots q_{i-1}) = P(q_i \mid q_{i-1})$$ – First order MP

- So far the states are visible
- To model more complex transitions we might need to use hidden markov models where states are hidden and we only make observations.



HMM for Ice Cream

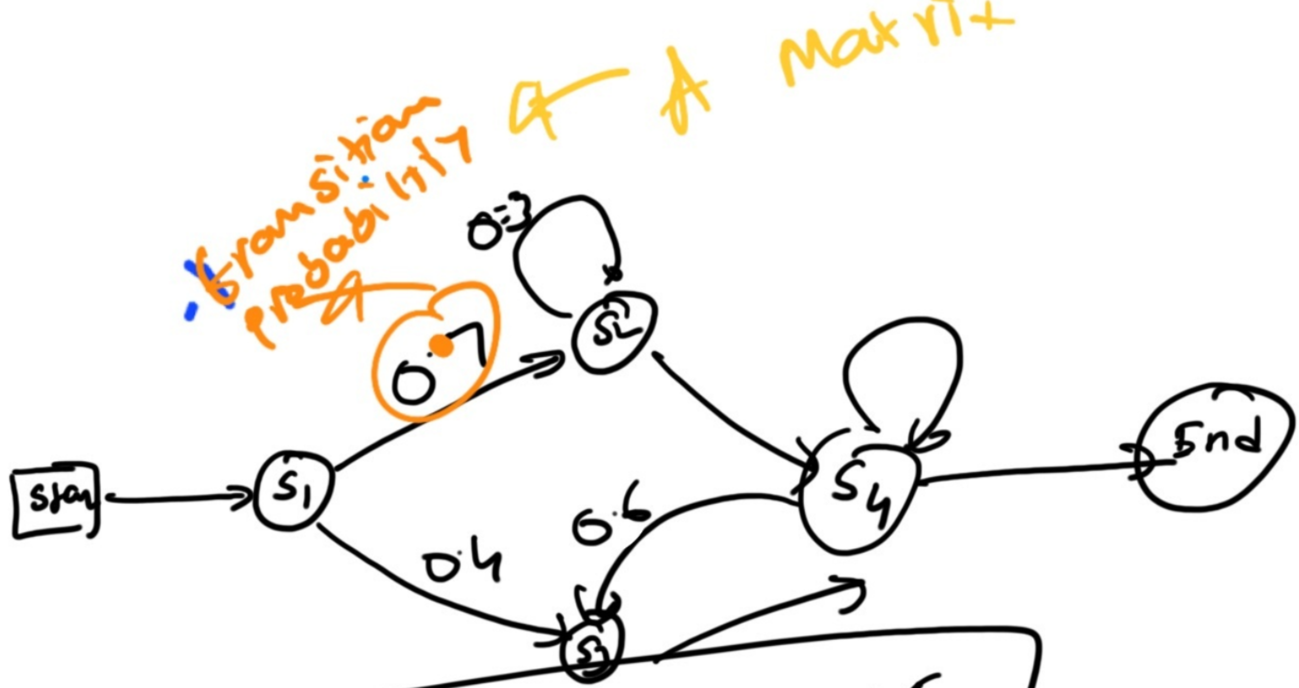
- You are a climatologist in the year 2799
- Studying global warming
- You can't find any records of the weather in Delhi for summer of 2025
- But you find Rahul's diary
- Which lists how many ice-creams Rahul ate every date that summer
- Our job: figure out how hot it was

Hidden Markov Model

- For Markov chains, the output symbols are the same as the states.
  - See **hot** weather: we're in state **hot**
- But in named-entity or part-of-speech tagging (and speech recognition and other things)
  - The output symbols are **words**
  - But the hidden states are something else
    - **Part-of-speech tags**
    - **Named entity tags**
- So we need an extension!
- A Hidden Markov Model is an extension of a Markov chain in which the input symbols are not the same as the states.
- This means we don't know which state we are in.

Hidden Markov Models

$Q = q_1 q_2 \ldots q_N$	a set of N **states**
$A = a_{11} a_{12} \ldots a_{n1} \ldots a_{nn}$	a **transition probability matrix** A, each a_{ij} representing the probability of moving from state i to state j, s.t. $\sum_{j=1}^{n} a_{ij} = 1 \quad \forall i$
$O = o_1 o_2 \ldots o_T$	a sequence of T **observations**, each one drawn from a vocabulary $V = v_1, v_2, ..., v_V$
$B = b_i(o_t)$	a sequence of **observation likelihoods**, also called **emission probabilities**, each expressing the probability of an observation o_t being generated from a state i
q_0, q_F	a special **start state** and **end (final) state** that are not associated with observations, together with transition probabilities $a_{01} a_{02} \ldots a_{0n}$ out of the start state and $a_{1F} a_{2F} \ldots a_{nF}$ into the end state



$p(O_1 | S_3) = 0.05$

$p(O_2 | S_3) = 0.01$

$p(O_3 | S_3) = 0.06$

$p(O_4 | S_3) = 0.7$

$= 0.8$

Observation probability
or Emission probability

B matrix

Assumptions

- **Markov assumption:**

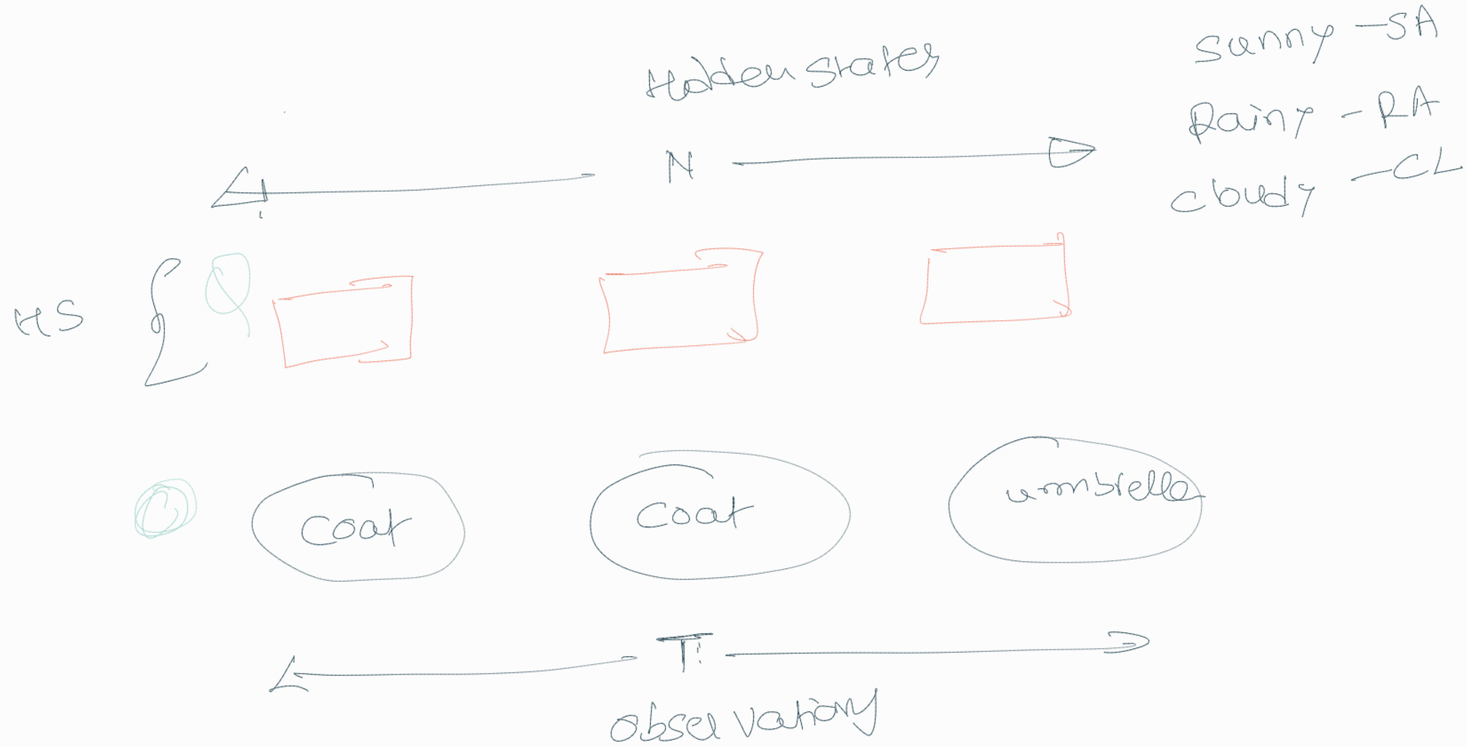
$$P(q_i \mid q_1 \ldots q_{i-1}) = P(q_i \mid q_{i-1})$$

- the current state is dependent only on the previous state.
- this represents the memory of the model

- **Output-independence assumption**

$$P(o_t \mid O_1^{t-1}, q_1^t) = P(o_t \mid q_t)$$

- the output observation at time t is dependent only on the current state
- it is independent of previous observations and states



Number of combination = N^T

complexity = exponential

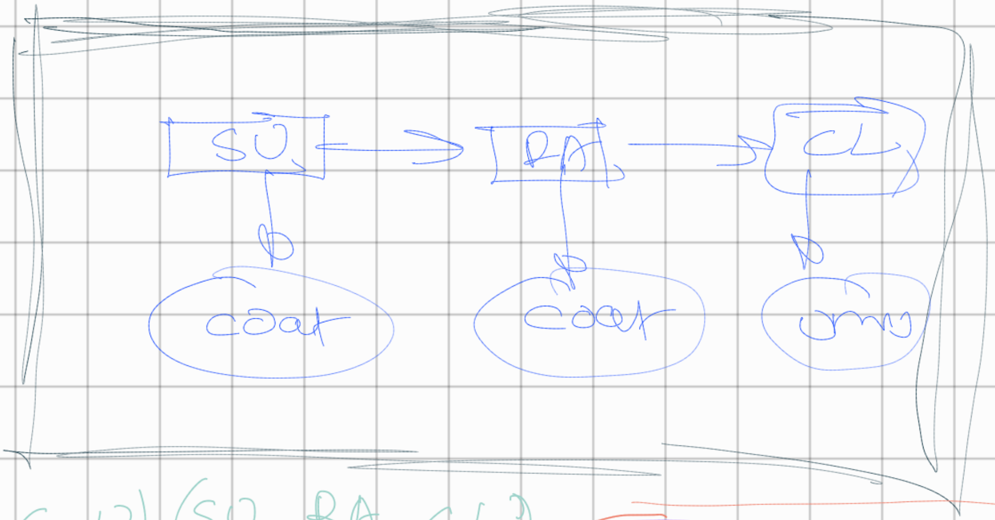
Joint probability

$$P(A, B) = P(A|B)\,P(B)$$

$$P(\text{coat}, SU) = P(\text{coat}|SU)\,P(SU)$$

$$P(O, Q) = P(O|Q)\,P(Q)$$

$\pi = [0.75, 0.2, 0.05]$ ← How to compute initial probabilities



$P((C,C,U),(SU,RA,CL))$

$= P(C|SU)\,P(C|RA)\,P(U|CL)$

$\quad P(SU)\,P(RA|SU)\,P(CL|RA)$

Emission matrix

$= 0.3 \times 0.3 \times 0.5 \times 0.75 \times 0.15 \times 0.02 =$

	short	coat	umbrella
SU	0.6	0.3	0.1
RA	0.05	0.3	0.65
CL	0	0.5	0.5

transition matrix

	SU	RA	CL
SU	0.8	0.15	0.05
RA	0.38	0.6	0.02
CL	0.75	0.05	0.2

$$\boxed{P(O,Q) = \prod P(O_i|q_i) \times \prod P(q_i|q_{i-1})}$$

The school is closed today.

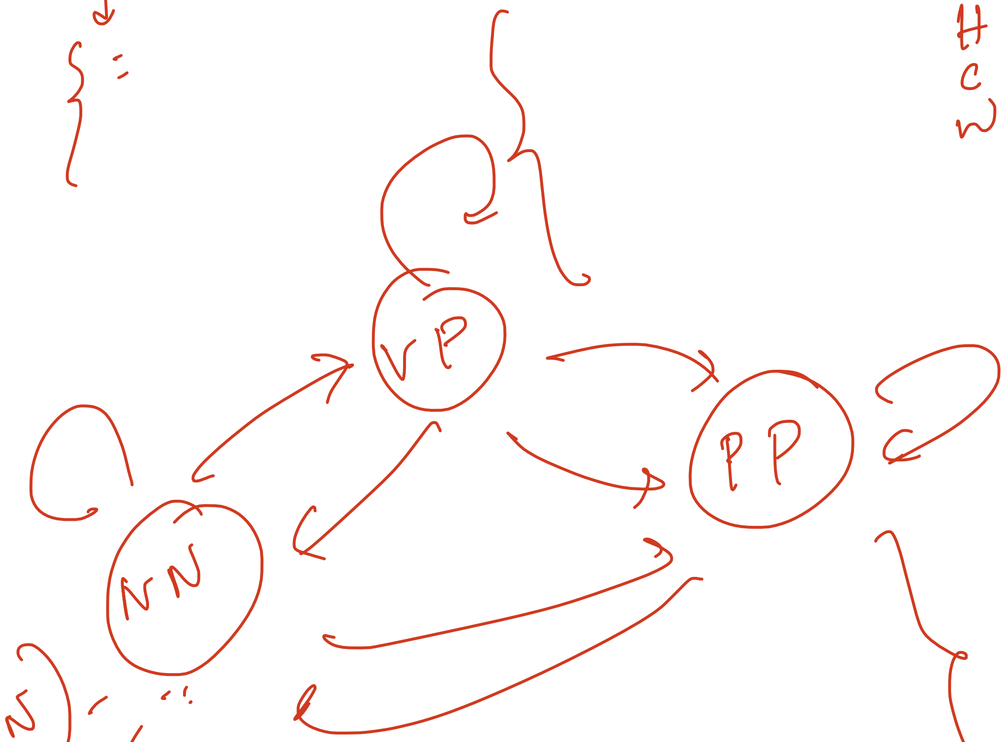
2 5 8 7 9

H H H

C C C

W W V

LM



$P(\text{The} \mid NN)$

$P(\ldots \mid NN)$

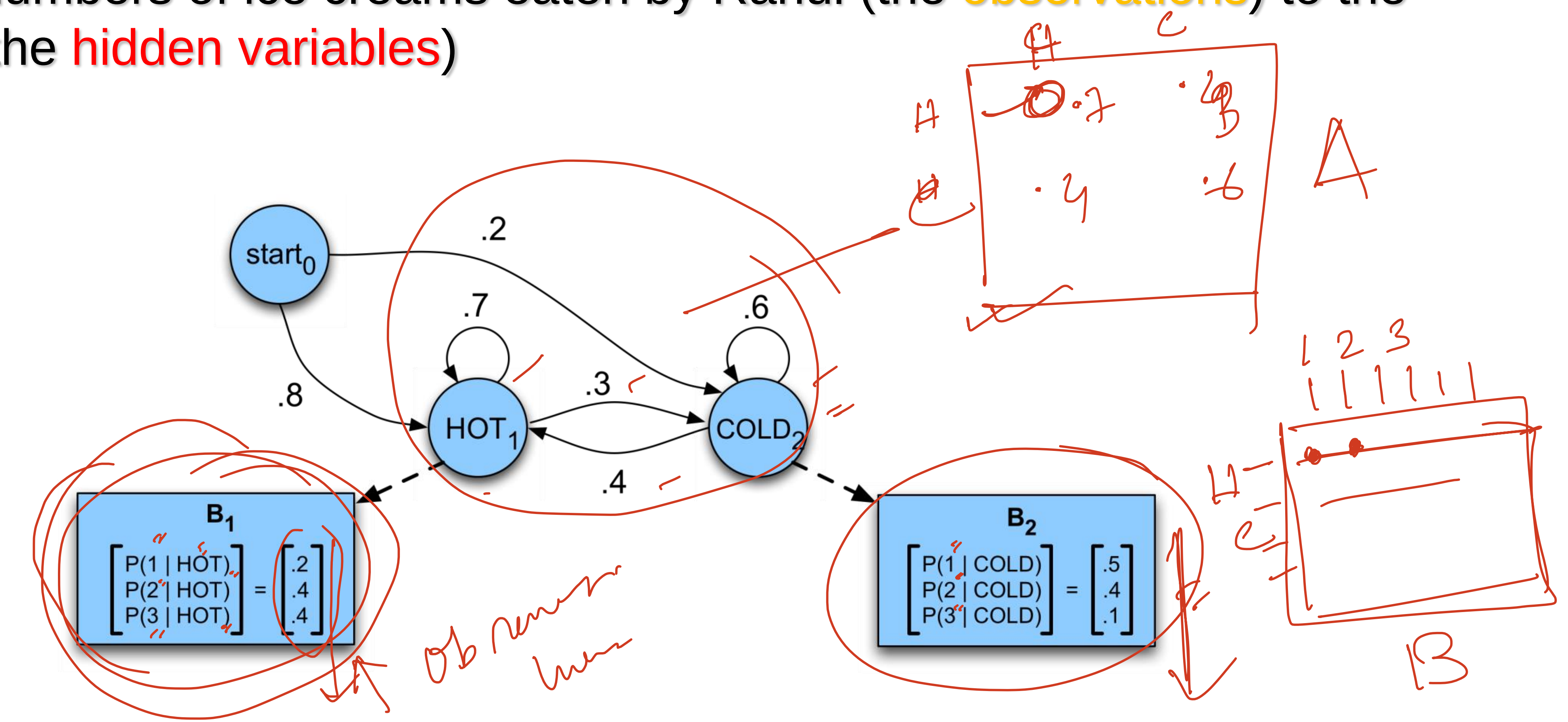
The Ice Cream task (cont.)

- Given a sequence of observations O,
  - each observation an integer = number of ice creams eaten
  - Figure out correct hidden sequence Q of weather states (H or C) which caused Rahul to eat the ice cream

In other words:
Given:
 Ice Cream Observation Sequence: 1,2,3,2,2,2,3…
Produce:
 Weather Sequence: H,C,H,H,H,C…

An HMM for this task

- Relating numbers of ice creams eaten by Rahul (the observations) to the weather (the hidden variables)



The Three Basic Problems for HMMs : more formally

- Problem 1 (**Evaluation):** Given the observation sequence $O=(o_1 o_2 \ldots o_T)$, and an HMM model $\Phi = (A,B)$, how do we efficiently compute $P(O|\Phi)$, the probability of the observation sequence, given the model

$P(O|\Phi)$

A = [tran pr]

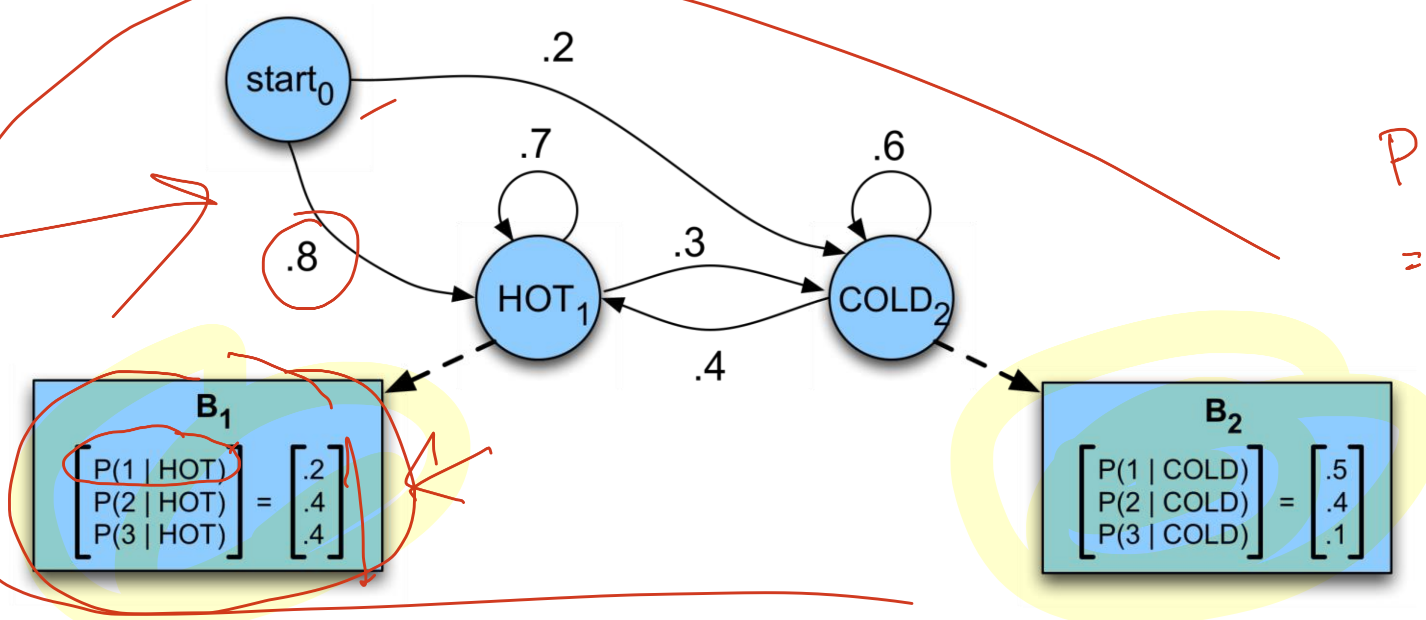
B = [Obs pr.]

3|3

Problem 1: computing the observation likelihood

Computing Likelihood: Given an HMM $\lambda = (A, B)$ and an observation sequence O, determine the likelihood $P(O|\lambda)$.

- Given the following HMM:



P(H C C)
= P(H|<start>) P(C|H) P(C|C)

- How likely is the sequence 3 1 3? P(3 1 3) P(3 1 3 | H C C)

3 1 3
H C C

$P(313 \mid HCC)$

$= P(3 \mid H) \times P(1 \mid C) \times P(3 \mid C)$
$\times P(H \mid START) \times P(C \mid H) \times P(C \mid C)$

313
HH H
HH C
...
8

CC
HHH

$\max P(313, HCC) = P(313 \mid HCC) \times P(HCC)$

$= P(3 \mid H)\, P(1 \mid H)\, P(3 \mid C) \times P(H \mid S)\, P(C \mid H)\, P(C \ldots)$

$P(x, y)$

$y \in \ldots$

$P(313) = \sum_{y \in \ldots} P(313, y)$

$= P(313, HCC) + P(313, HC\ldots) + \ldots$

313
HHC

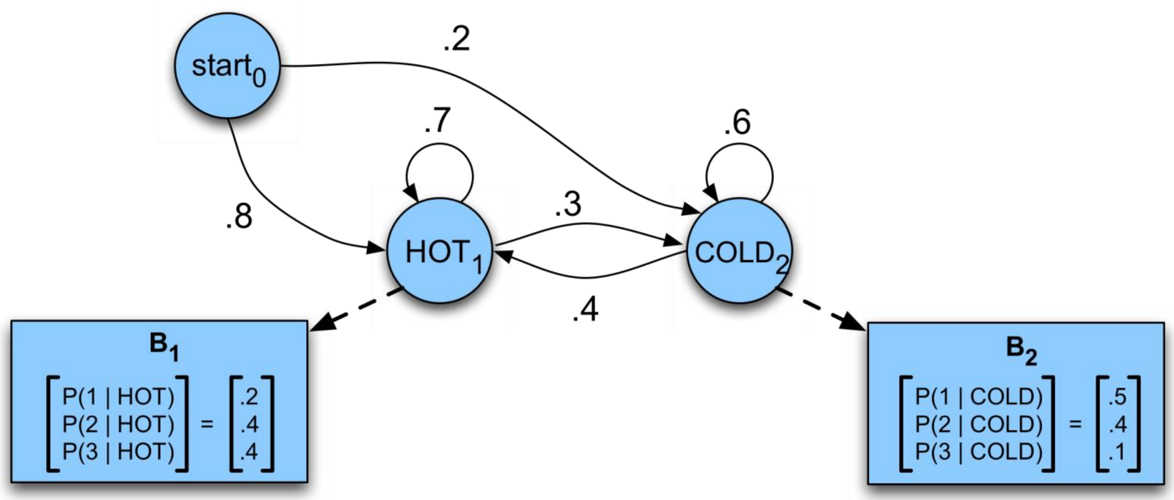
How to compute likelihood

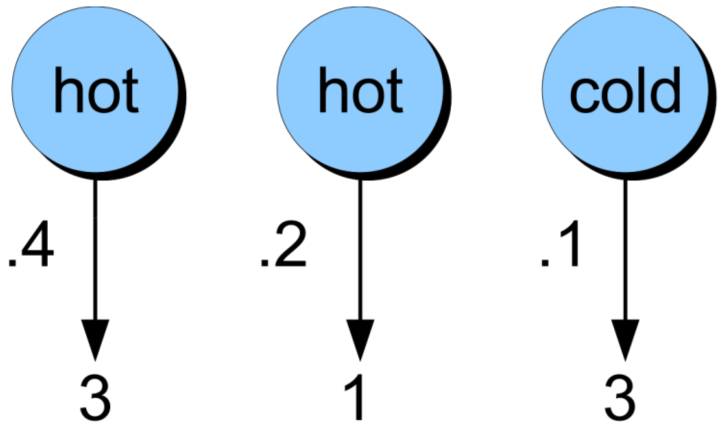
- For a Markov chain, we just follow the states 3 1 3 and multiply the probabilities
- But for an HMM, we don't know what the states are!
- So let's start with a simpler situation.
- Computing the observation likelihood for a **given** hidden state sequence
  - Suppose we knew the weather and wanted to predict how much ice cream Rahul would eat.
  - i.e. $P(3\ 1\ 3 \mid \mathrm{H\ H\ C})$

Computing likelihood of 3 1 3 given hidden state sequence

$$P(O|Q) = \prod_{i=1}^{T} P(o_i|q_i)$$

$$P(3\ 1\ 3|\text{hot hot cold}) \;=\; P(3|\text{hot}) \times P(1|\text{hot}) \times P(3|\text{cold})$$

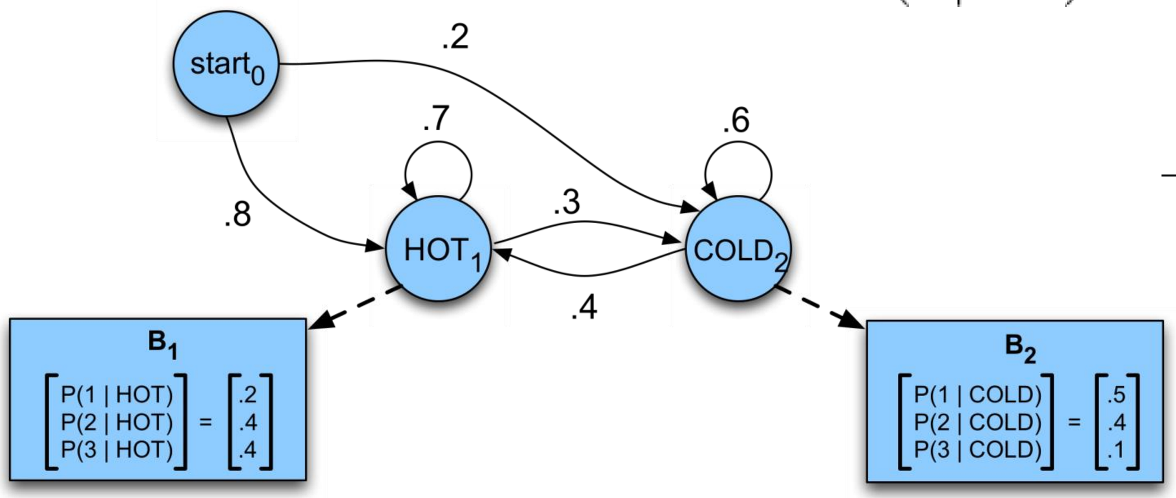


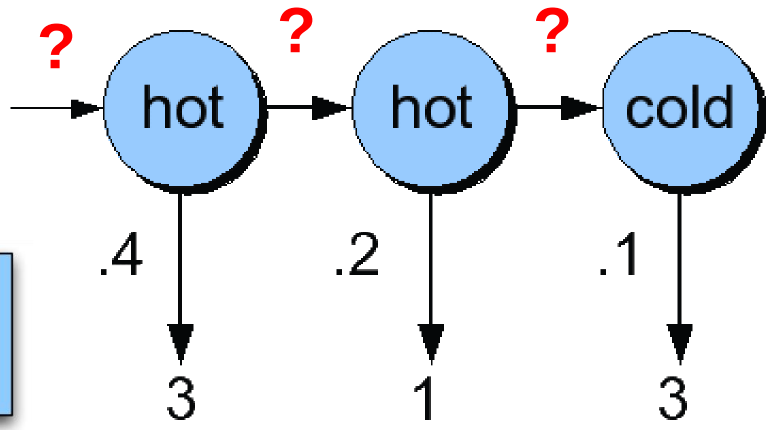


Computing joint probability of observation and state sequence

$$P(O,Q) = P(O|Q) \times P(Q) = \prod_{i=1}^{n} P(o_i|q_i) \times \prod_{i=1}^{n} P(q_i|q_{i-1})$$

$$\begin{aligned} P(3\ 1\ 3, \text{hot hot cold}) &= P(\text{hot}|\text{start}) \times P(\text{hot}|\text{hot}) \times P(\text{cold}|\text{hot}) \\ &\times P(3|\text{hot}) \times P(1|\text{hot}) \times P(3|\text{cold}) \end{aligned}$$





Computing total likelihood of 3 1 3

- We would need to sum over
  - Hot hot cold
  - Hot hot hot
  - Hot cold hot
  - ….

$$P(O) = \sum_{Q} P(O,Q) = \sum_{Q} P(O|Q)P(Q)$$

N^T

- How many possible hidden state sequences are there for this sequence?

$$P(3\ 1\ 3) = P(3\ 1\ 3, \text{cold cold cold}) + P(3\ 1\ 3, \text{cold cold hot}) + P(3\ 1\ 3, \text{hot hot cold}) + \ldots$$

- How about in general for an HMM with N hidden states and a sequence of T observations?
  - N^T
- So we can't just do separate computation for each hidden state sequence.

Instead: the Forward algorithm

- A kind of **dynamic programming** algorithm
  - Just like Minimum Edit Distance
  - Uses a table to store intermediate values
- Idea:
  - Compute the likelihood of the observation sequence
  - By summing over all possible hidden state sequences
  - But doing this efficiently
    - By folding all the sequences into a single **trellis**

The forward algorithm

- The goal of the forward algorithm is to compute

$$P(o_1, o_2 \dots o_T, q_T = q_F \mid \lambda)$$

A =

B =

- We'll do this by recursion

The forward algorithm

- Each cell of the forward algorithm trellis $\alpha_t(j)$
  - Represents the probability of being in state j
  - After seeing the first t observations
  - Given the automaton
- Each cell thus expresses the following probability

$$\alpha_t(j) = P(o_1, o_2 \ldots o_t, q_t = j \mid \lambda)$$

The Forward Recursion

1. Initialization:

$$\alpha_1(j) \;=\; a_{0j} b_j(o_1) \;\; 1 \leq j \leq N$$

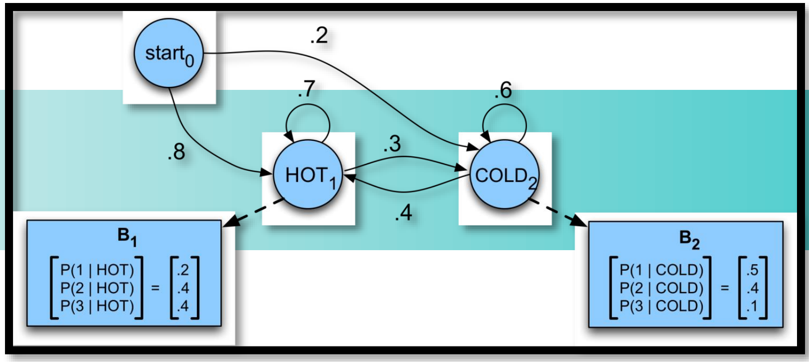
2. Recursion (since states 0 and F are non-emitting):

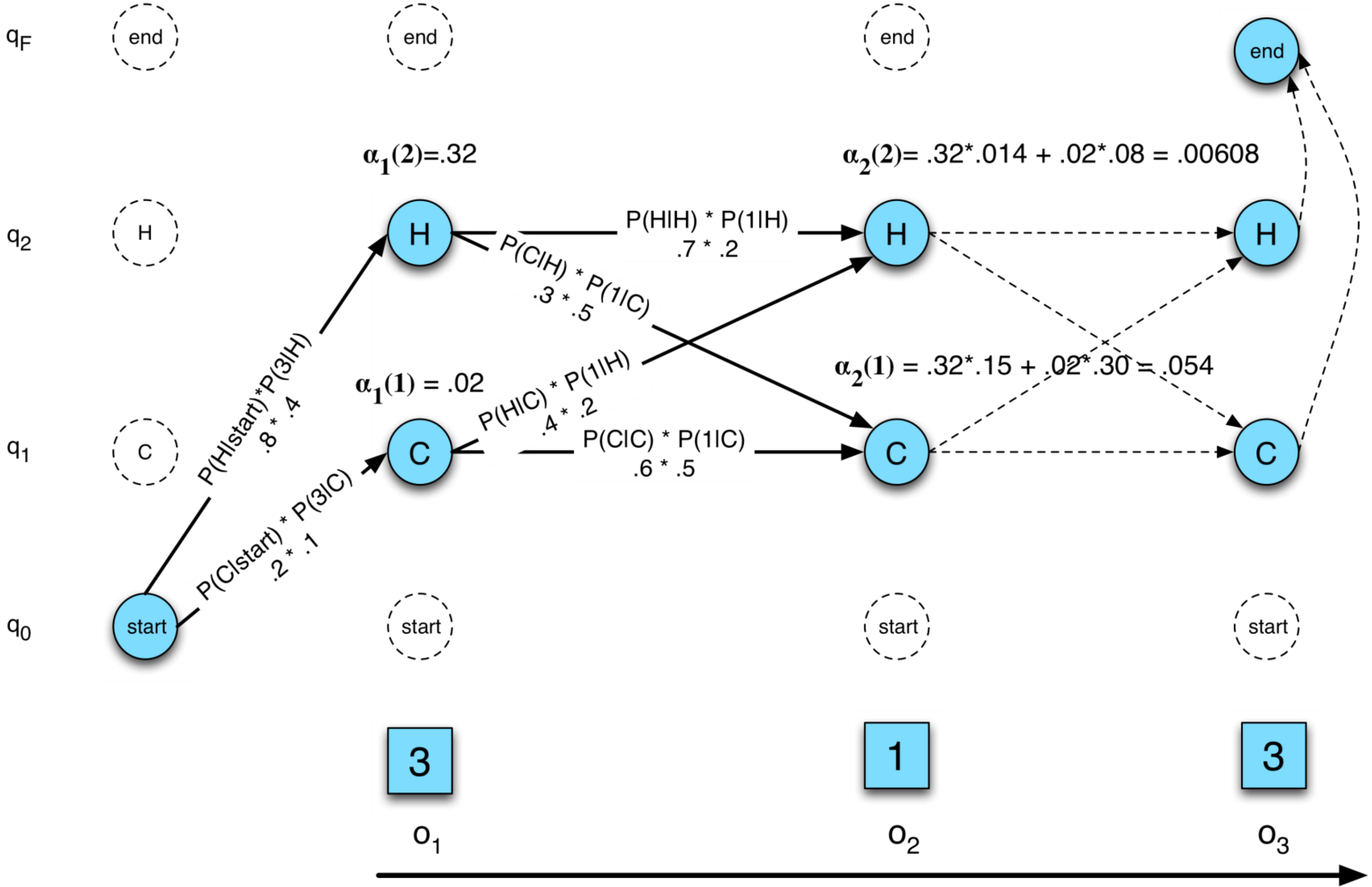
$$\alpha_t(j) = \sum_{i=1}^{N} \alpha_{t-1}(i) a_{ij} b_j(o_t); \quad 1 \leq j \leq N, 1 < t \leq T$$

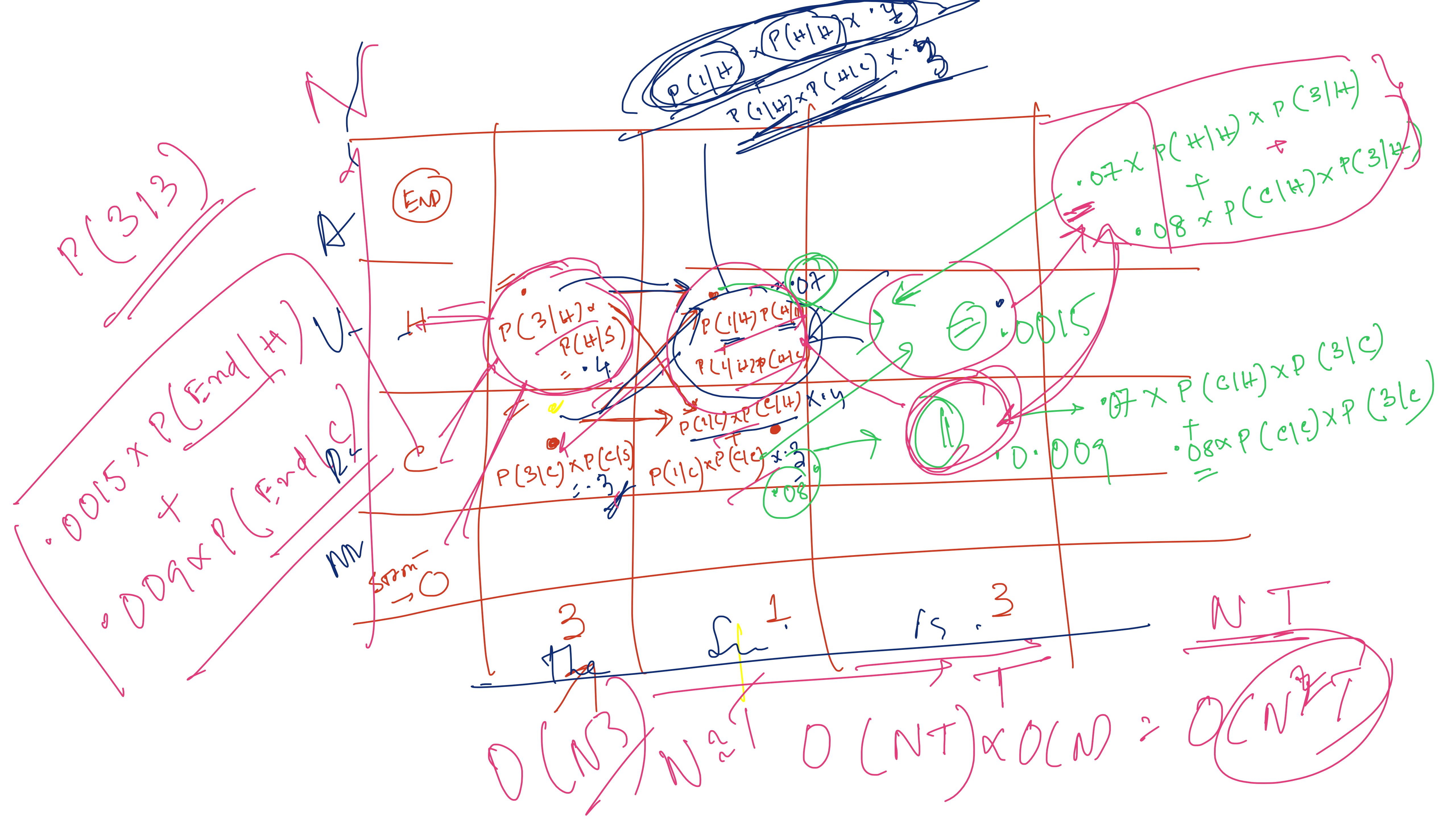
3. Termination:

$$P(O|\lambda) = \alpha_T(q_F) = \sum_{i=1}^{N} \alpha_T(i) a_{iF}$$

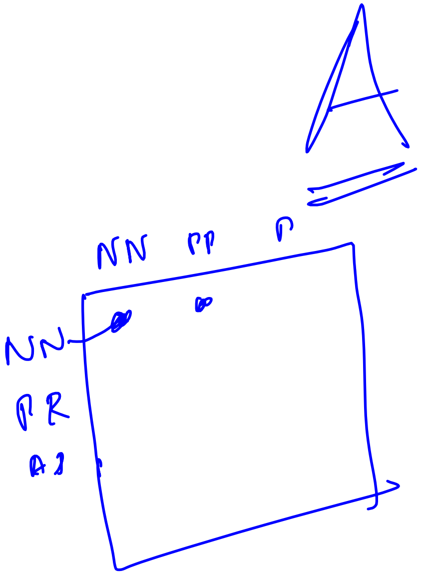
The Forward Trellis

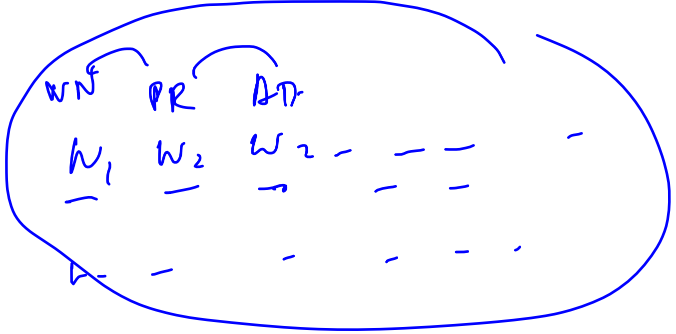


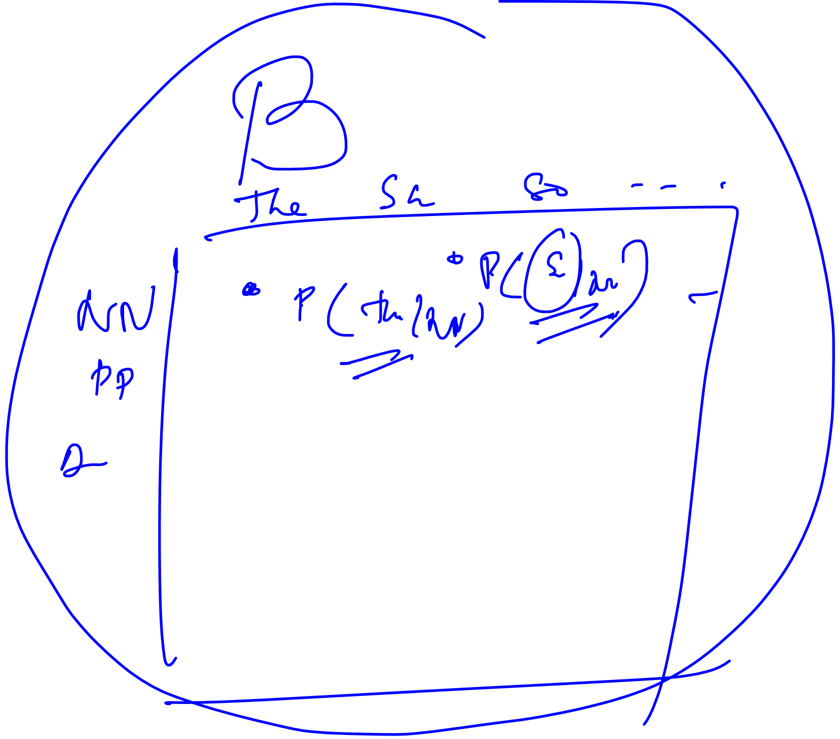




P(1|H) x P(H|H) x .4
+
P(1|H) x P(H|C) x .3
END
N
V
H
C
.07
.08
P(3|H) x P(H|S) = .4
P(3|C) x P(C|S) = .3
P(1|H) P(H|H)
P(1|H) P(H|C)
P(1|C) x P(C|H) x .4
+
P(1|C) x P(C|C) x .3
.0015
.009
.07 x P(H|H) x P(3|H)
+
.08 x P(C|H) x P(3|H)
.07 x P(C|H) x P(3|C)
+
.08 x P(C|C) x P(3|C)
P(3|3)
.0015 x P(End|H)
+
.009 x P(End|C)
Start
3
1
3
The
Sun
is
N T
O(N3)
N ≈ 1
O(NT) x O(N) = O(N²T)

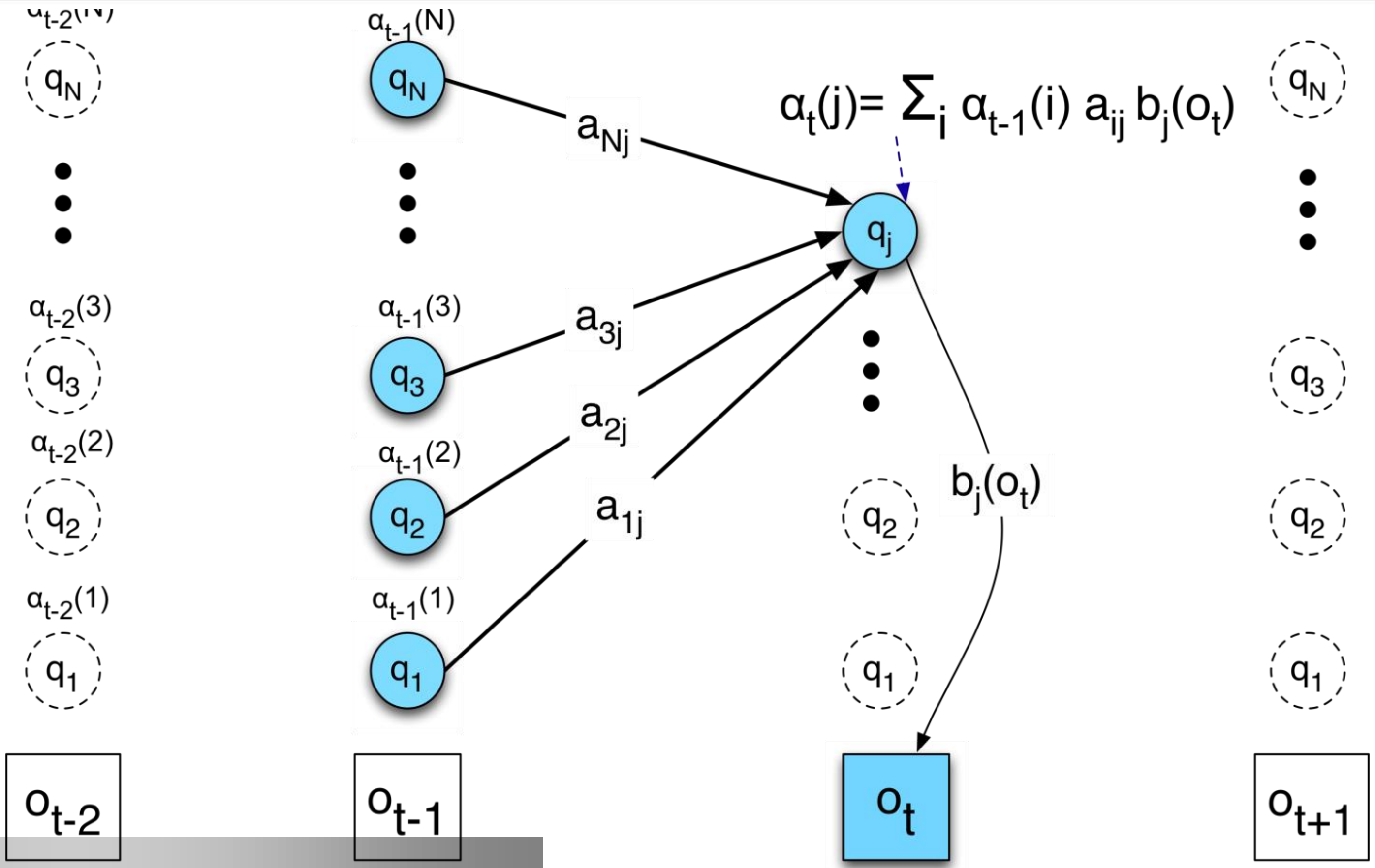

A
NN PP P
NN
PR
AJ


NN PR AJ
w1 w2 w2


B
The Sa So
NN
PP
D
P(The|NN) P(Sa|NN)

We update each cell

$\alpha_{t-1}(i)$	the **previous forward path probability** from the previous time step
a_{ij}	the **transition probability** from previous state q_i to current state q_j
$b_j(o_t)$	the **state observation likelihood** of the observation symbol o_t given the current state j



The Forward Algorithm

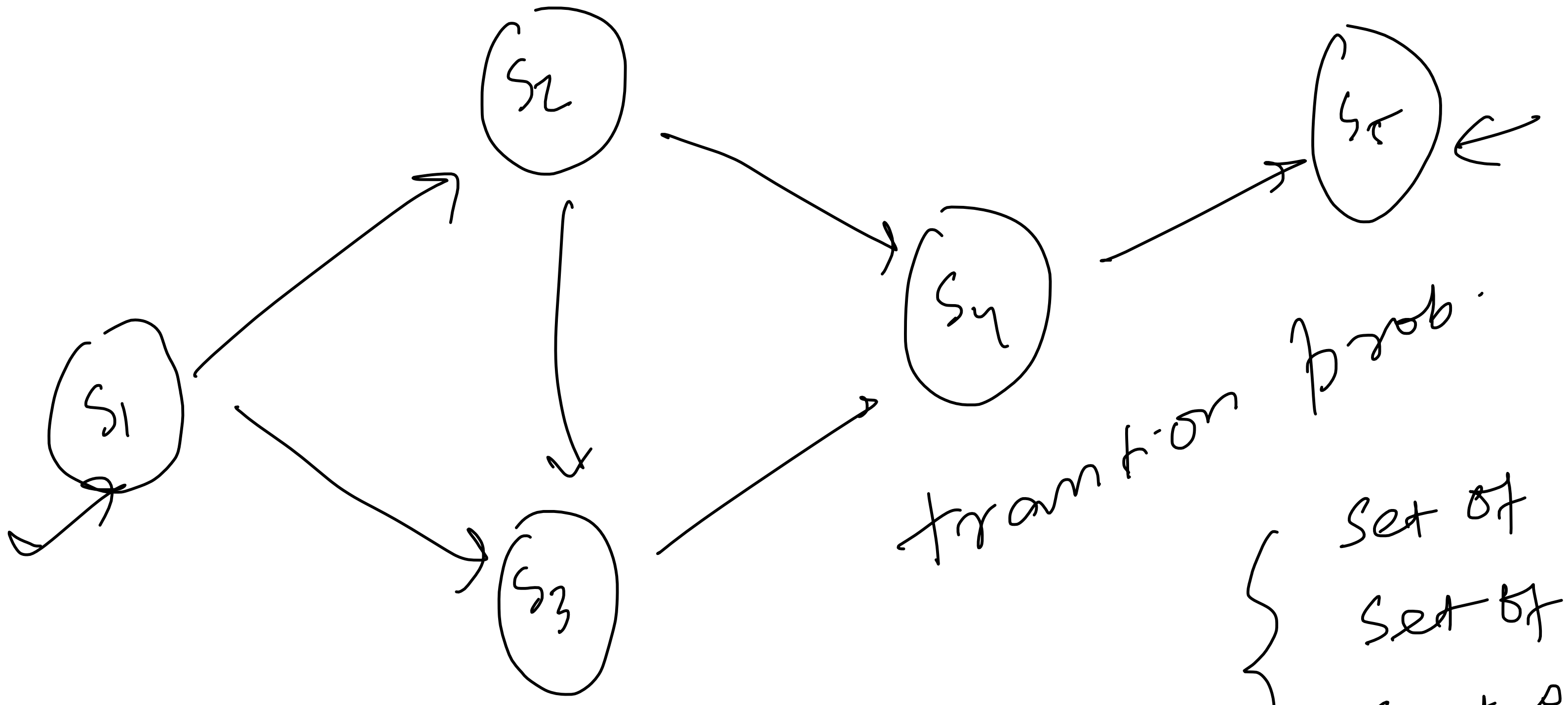
function FORWARD(*observations* of len T, *state-graph* of len N) **returns** *forward-prob*

create a probability matrix *forward[N+2,T]*
for each state s **from** 1 **to** N **do** ; initialization step
$forward[s,1] \leftarrow a_{0,s} * b_s(o_1)$
for each time step t **from** 2 **to** T **do** ; recursion step
for each state s **from** 1 **to** N **do**

$$forward[s,t] \leftarrow \sum_{s'=1}^{N} forward[s',t-1] * a_{s',s} * b_s(o_t)$$

$$forward[q_F,\mathrm{T}] \leftarrow \sum_{s=1}^{N} forward[s,T] * a_{s,q_F}$$; termination step

return $forward[q_F,T]$



transition prob.

Set of States
Set of tran prob.
Start state & end state