CS626: Speech, Natural Language Processing and the Web

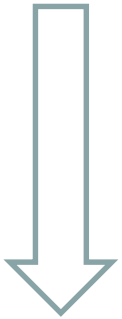
Part of Speech Tagging

Pushpak Bhattacharyya
Computer Science and Engineering Department
IIT Bombay

Week 2 of 1st August, 2022

What is NLP

NLP= Language + Computation

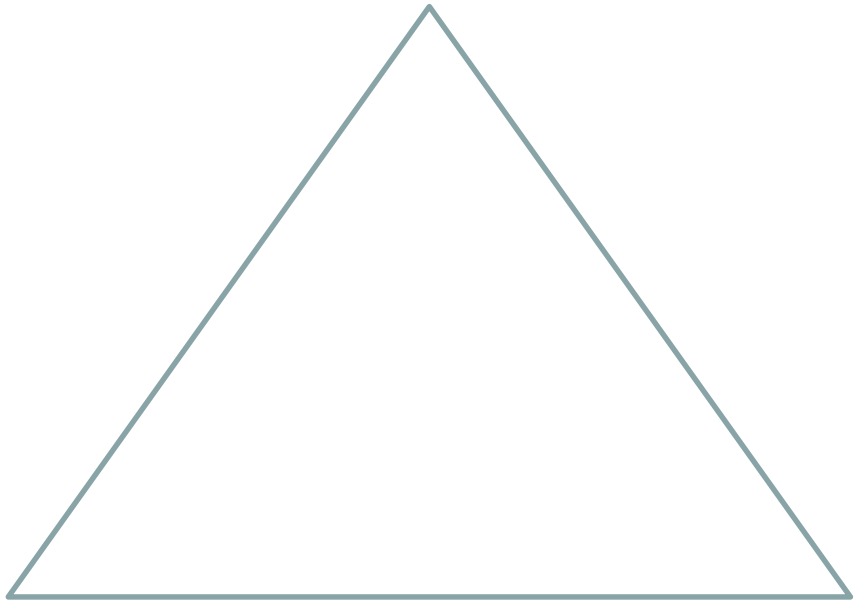


(due to ML)

= Linguistics + Probability

The Trinity

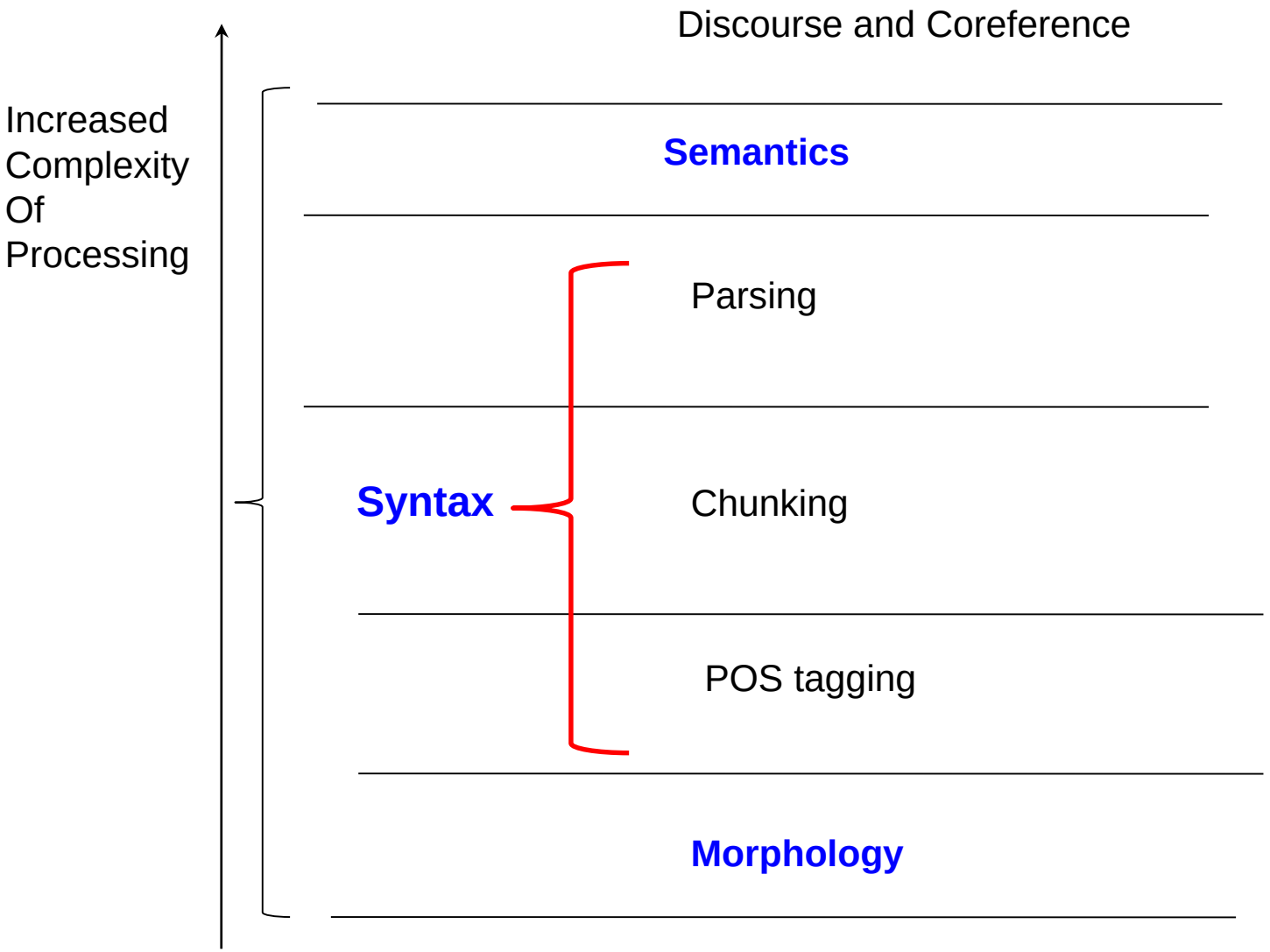
Linguistics



Probability

Coding

NLP Layers


Discourse and Coreference
Increased
Complexity
Of
Processing
Semantics
Parsing
Syntax
Chunking
POS tagging
Morphology

Linguistic Strata *vs.* Languages

	Hindi	**Swahili**	**Tamil**	***etc.***
Sound: Phonetics, Phonology				
Structure: Morphology, Syntax				
Meaning: Semantic, Pragmatics				

Speech-NLP Stack *vs.* Languages

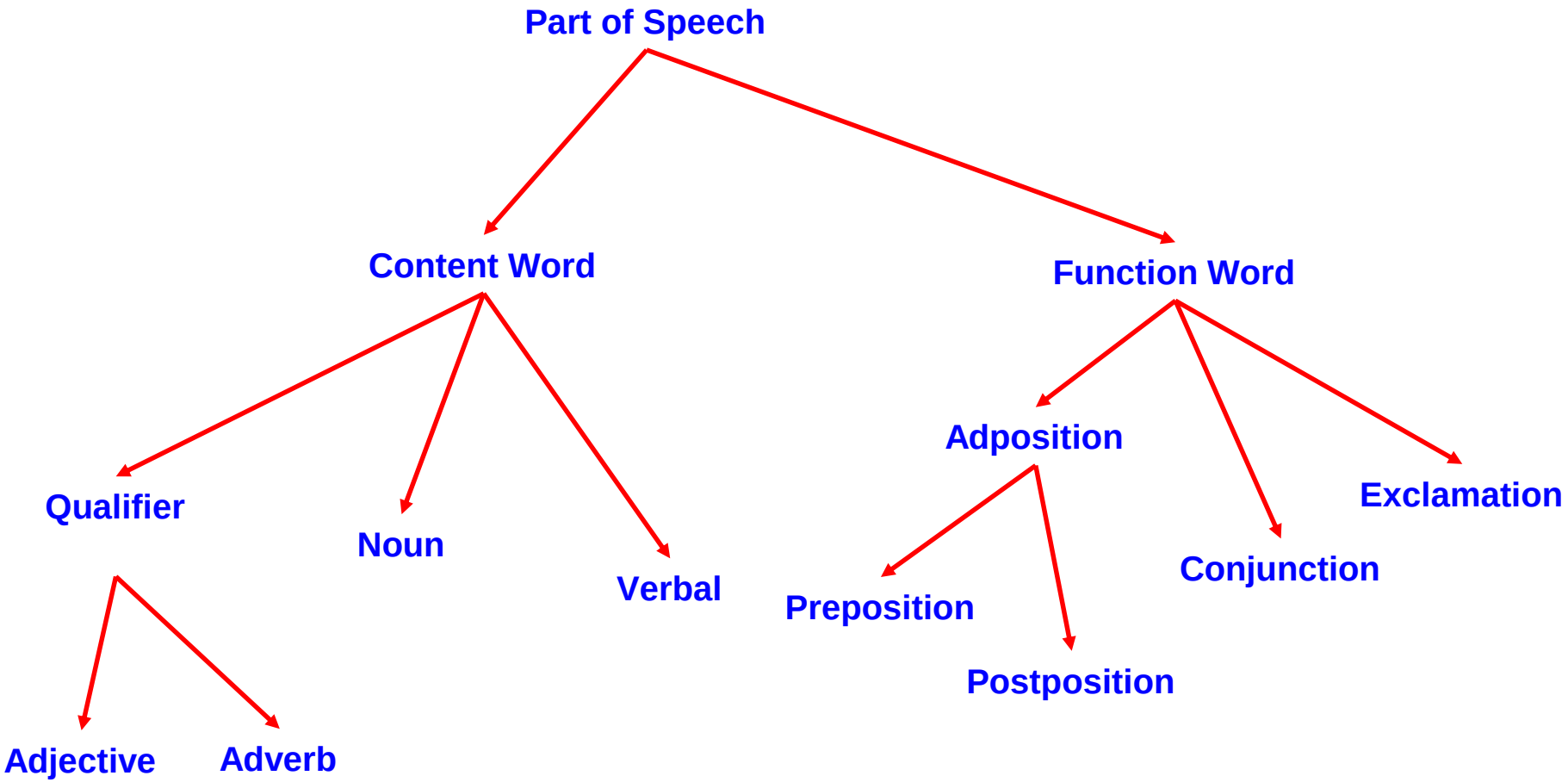
	Hindi	**Swahili**	**Tamil**	***etc.***
Sound: ASR, TTS				
Structure: MA, POS, NE, Chunker, Parser				
Meaning: SRL, Knowledge Nets, SA-EA-OM, QA, Summarizer				

Part of Speech Tagging

What is “Part of Speech”

- Words are divided into different kinds or classes, called Parts of Speech, according to their use; that is, according to the work they do in a sentence.
- The parts of speech are eight in number:
  1. Noun. 2. Adjective. 3. Pronoun. 4. Verb. 5. Adverb. 6. Preposition. 7. Conjunction. 8. Interjection.

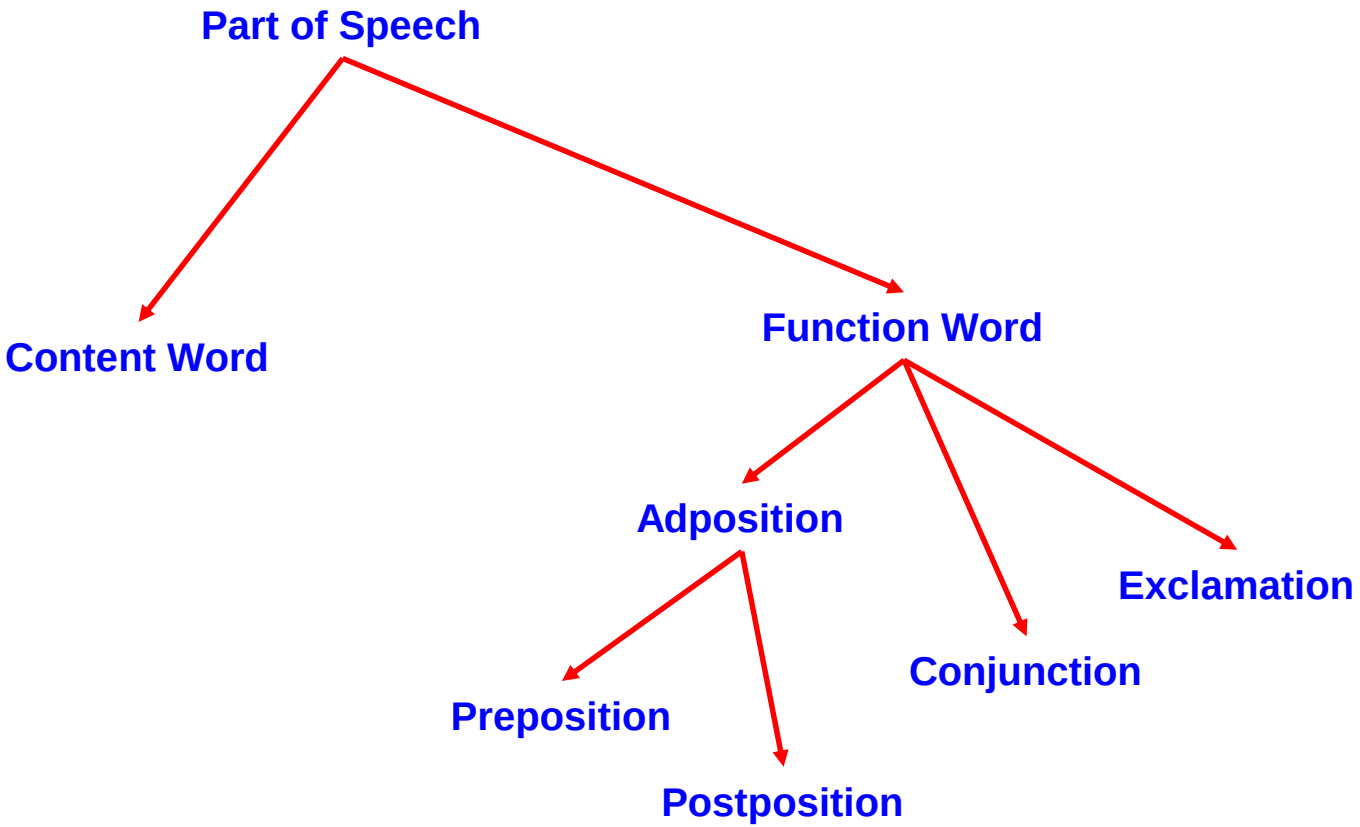
POS hierarchy



Understanding two methods of classification

- Syntactic
  - Important, e.g., for POS tagging
- Semantics
  - Important. E.g., for question answering
- Example: Adjectives
  - Syntactic: normal, comparative, superlative: *good, better, best*
  - Semantic: qualitative, quantitative: *tall man*, *forty horses*

POS hierarchy: Function Words



Problem Statement

- Input: a sequence of words
- Output: a sequence of labels of these words

Example

- *Who is the prime minister of India?*
  - *Who*- WP (who pronoun)
  - *Is*- VZ (auxiliary verb)
  - *The*- DT (determiner)
  - *Prime*- JJ (adjective)
  - *Minister*- NN (noun)
  - *Of*- IN (preposirtion)
  - *India*- NNP (proper noun)
  - *?*- PUNC (punctuation)
- These POS tags are as per the Penn Treebank tagset

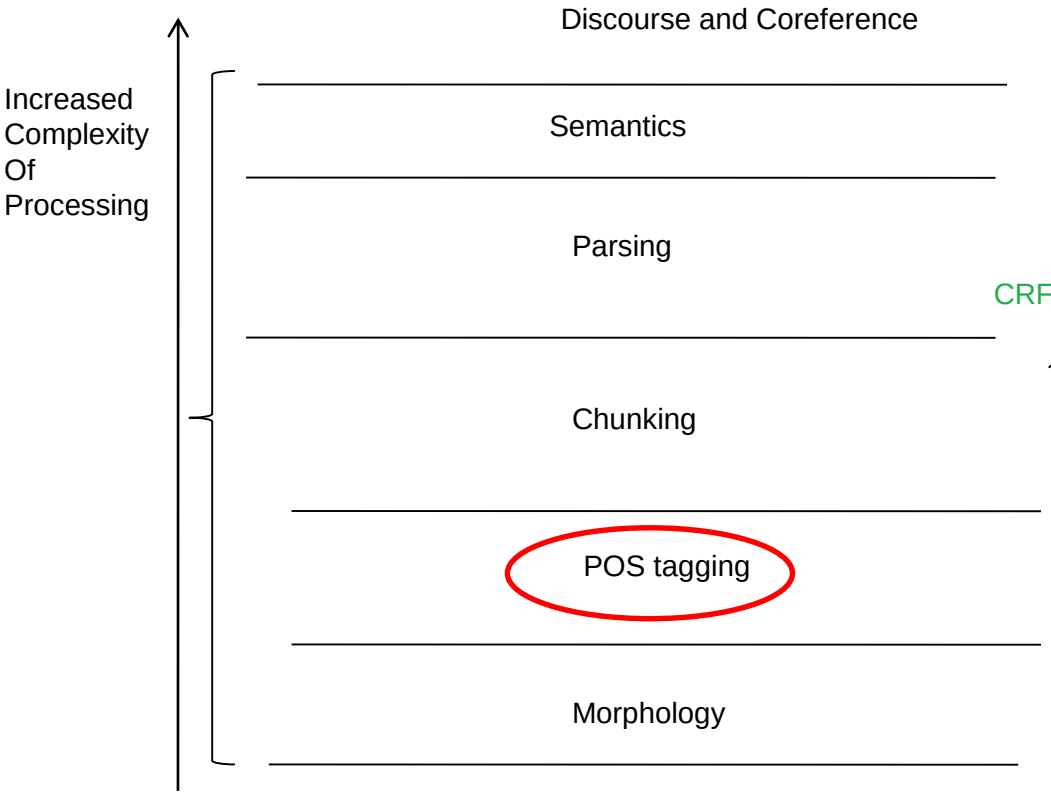
Motivation for POS tagging

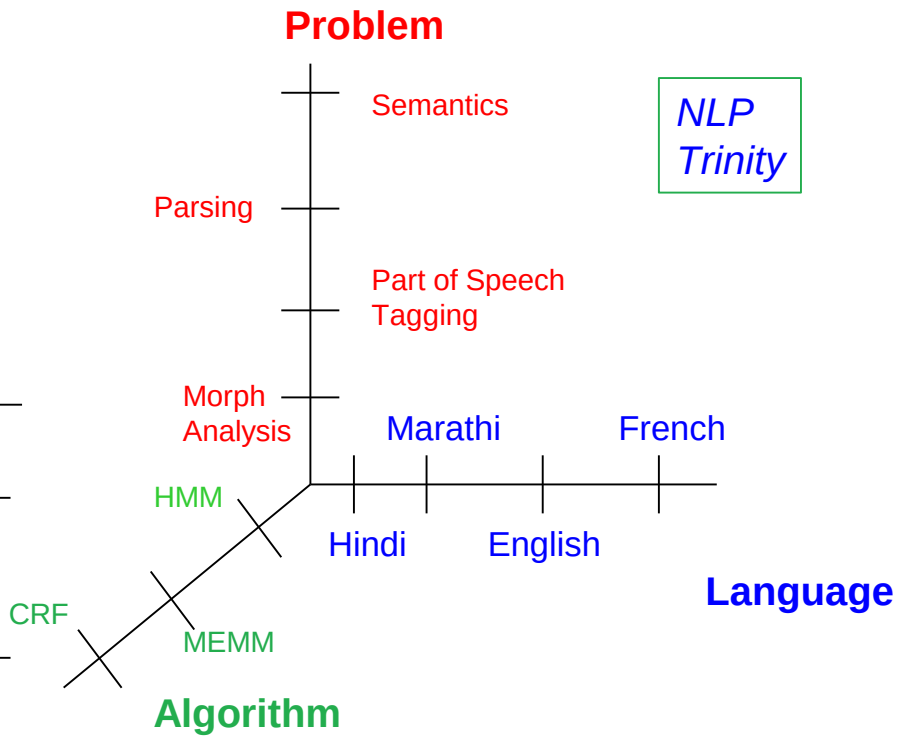
- Question Answering
- Machine Translation
- Summarization
- Entailment
- Information Extraction

Machine Translation

- "*I bank on your moral support*" to Hindi "*main aapake naitik samarthan par nirbhar hUM*"
- needs disambiguation of '*bank*' (verb; POS tag VB) to '*nirbhar hUM*'- as opposed to the noun '*bank*' meaning *the financial organization* or *the side of a water body*.

NLP: multilayered, multidimensional





POS Annotation

- *Who_WP is_VZ the_DT prime_JJ minister_NN of _IN India_NNP ?_PUNC*

- Becomes the training data for ML based POS tagging

3 Generations of POS tagging techniques

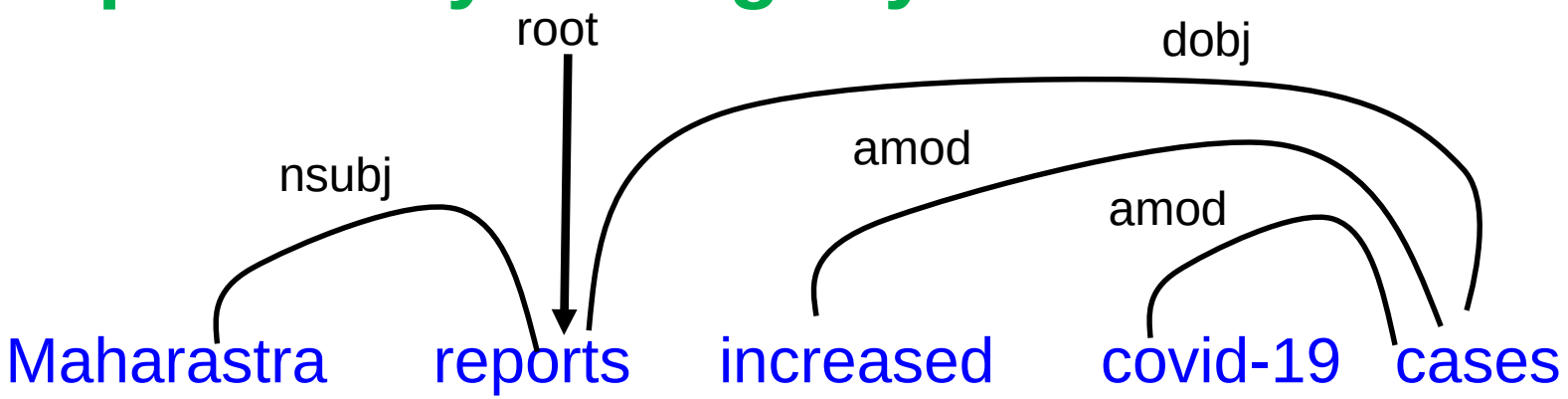
- Rule Based POS Tagging
  - Rule based NLP is also called Model Driven NLP
- Statistical ML based POS Tagging (*Hidden Markov Model, Support Vector Machine*)
- Neural (Deep Learning) based POS Tagging

Necessity of POS Tagging (1/2)

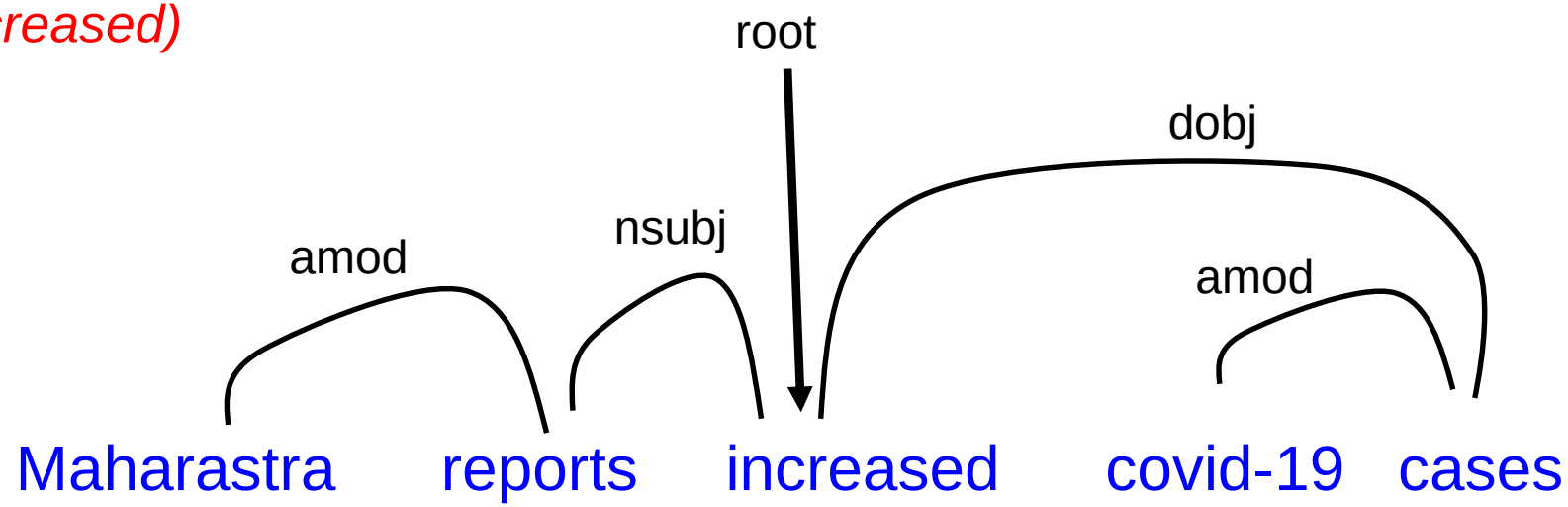
- **Command** Center to Track Best Buses (ToI 30Jan21): POS ambiguity affects

- Elderly with young **face** increased covid 19 risk (ToI Oct 20)

Dependency Ambiguity

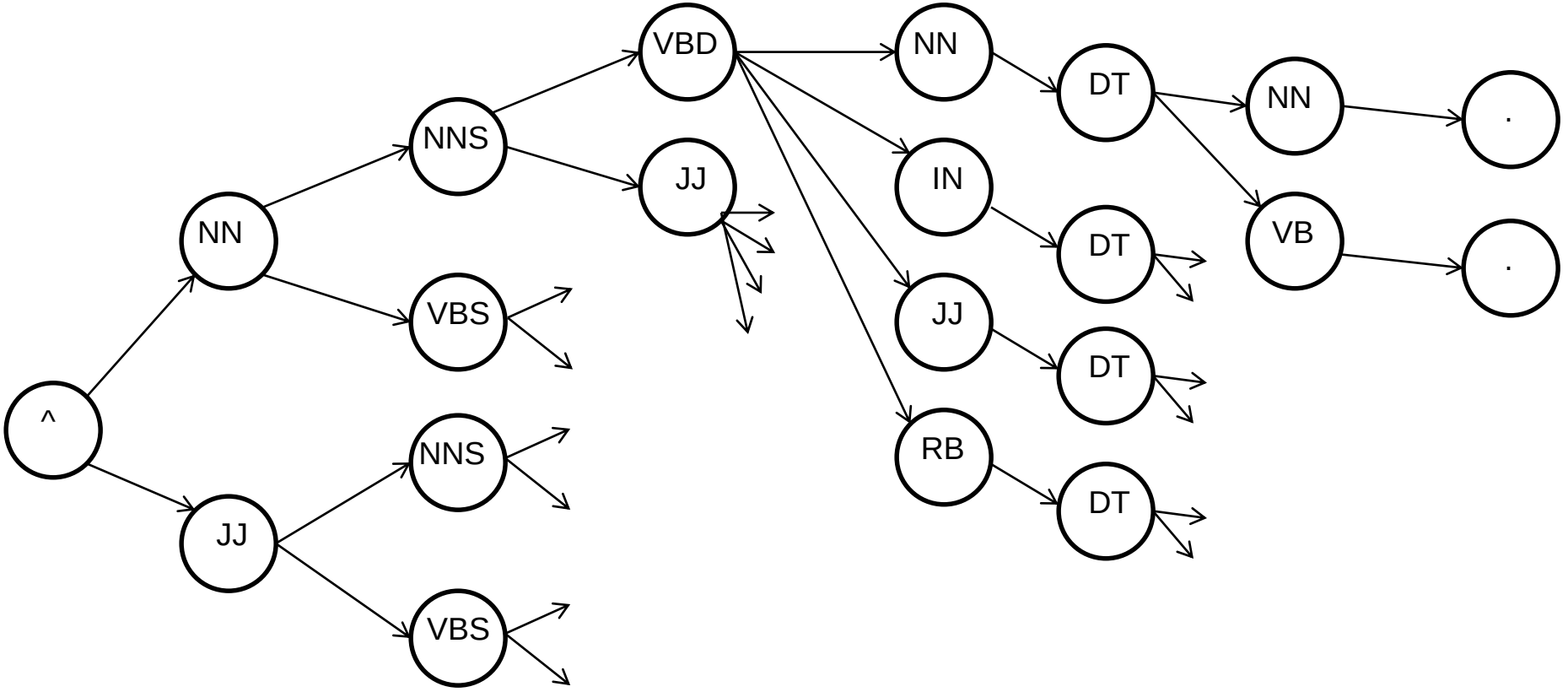


(it is reported by Maharastra Govt. that covid-19 cases have increased)

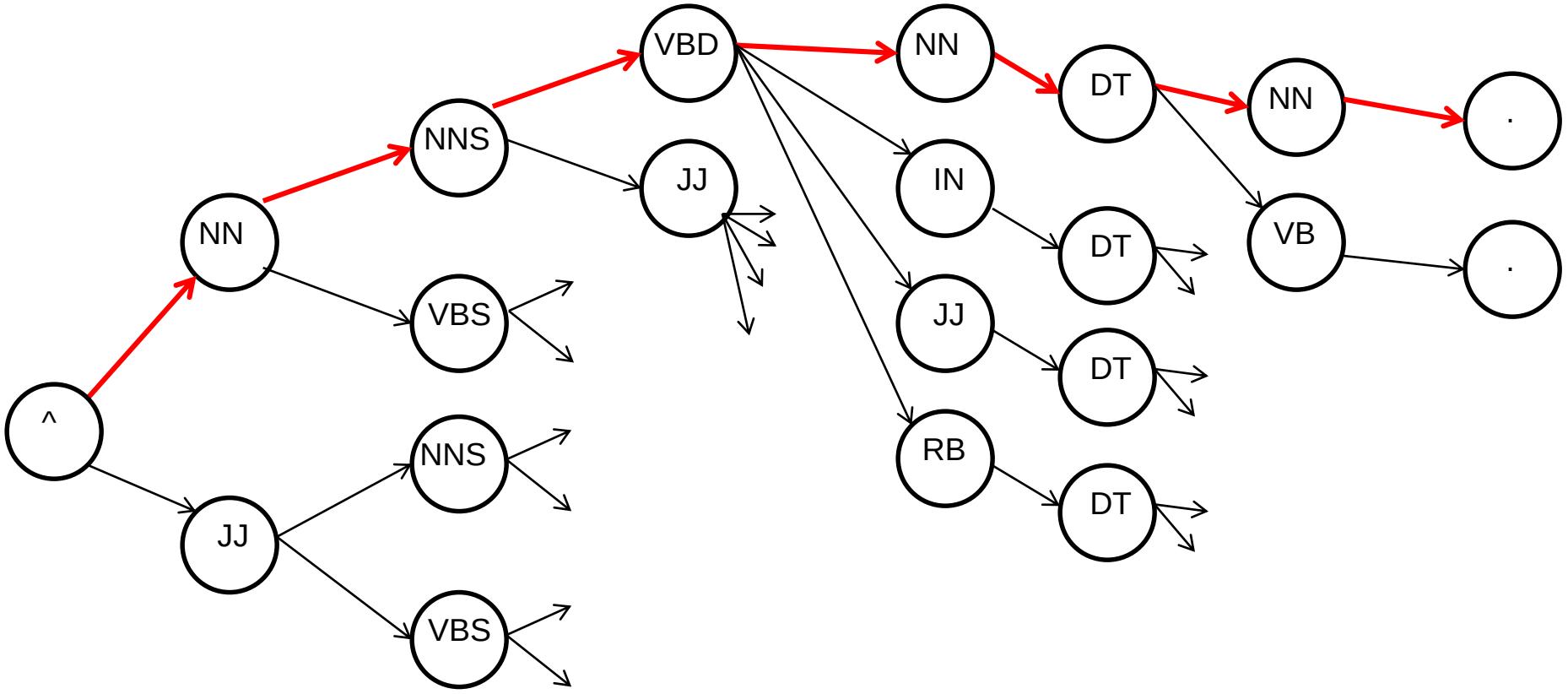


(it is the Maharastra reports that have increased covid-19 cases!!!)

W:	^	Brown	foxes	jumped	over	the	fence	.
T:	^	JJ	NNS	VBD	NN	DT	NN	.
		NN	VBS	JJ	IN		VB	
					JJ			
					RB			



^ Brown foxes jumped over the fence .

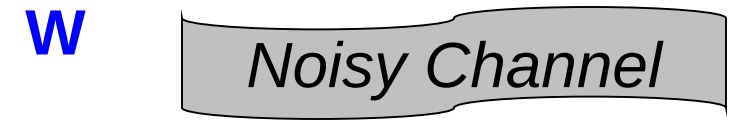


^ Brown foxes jumped over the fence .

Find the PATH with MAX Score.

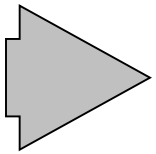
What is the meaning of score?

Noisy Channel Model



$(w_n, w_{n-1}, \ldots, w_1)$ $(t_m, t_{m-1}, \ldots, t_1)$

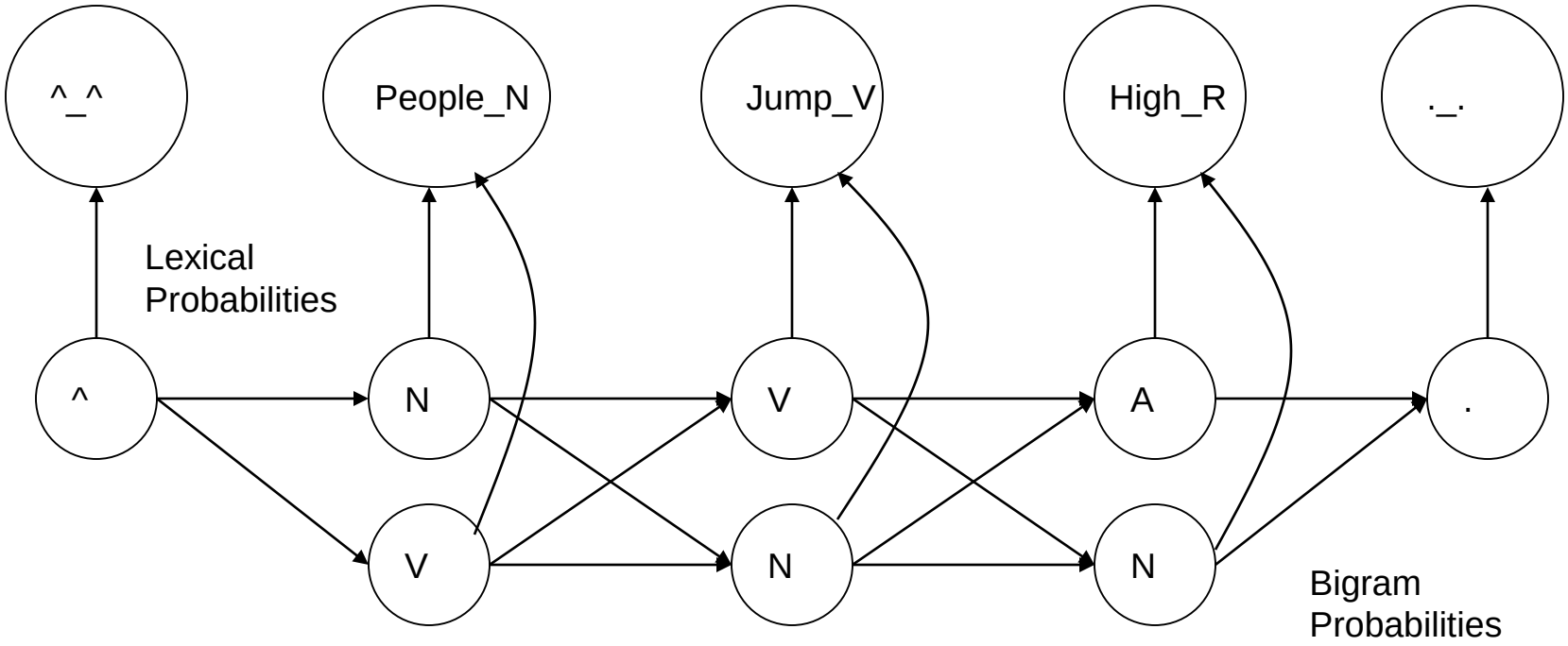
Sequence *W* is transformed into sequence *T*



$$T^* = \underset{T}{\operatorname{argmax}}(P(T|W))$$

$$W^* = \underset{W}{\operatorname{argmax}}(P(W|T))$$

HMM: Generative Model



This model is called Generative model.
Here words are observed from tags as states.
This is similar to HMM.

Tag Set

- Attach to each word a tag from **Tag-Set**
- Standard Tag-set : Penn Treebank (for English).

Penn POS TAG Set

1.	CC	Coordinating conjunction
2.	CD	Cardinal number
3.	DT	Determiner
4.	EX	Existential *there*
5.	FW	Foreign word
6.	IN	Preposition or subordinating conjun
7.	JJ	Adjective
8.	JJR	Adjective, comparative
9.	JJS	Adjective, superlative
10.	LS	List item marker
11.	MD	Modal
12.	NN	Noun, singular or mass
13.	NNS	Noun, plural
14.	NNP	Proper noun, singular
15.	NNPS	Proper noun, plural
16.	PDT	Predeterminer
17.	POS	Possessive ending
18.	PRP	Personal pronoun
19.	PRP$	Possessive pronoun
20.	RB	Adverb
21.	RBR	Adverb, comparative

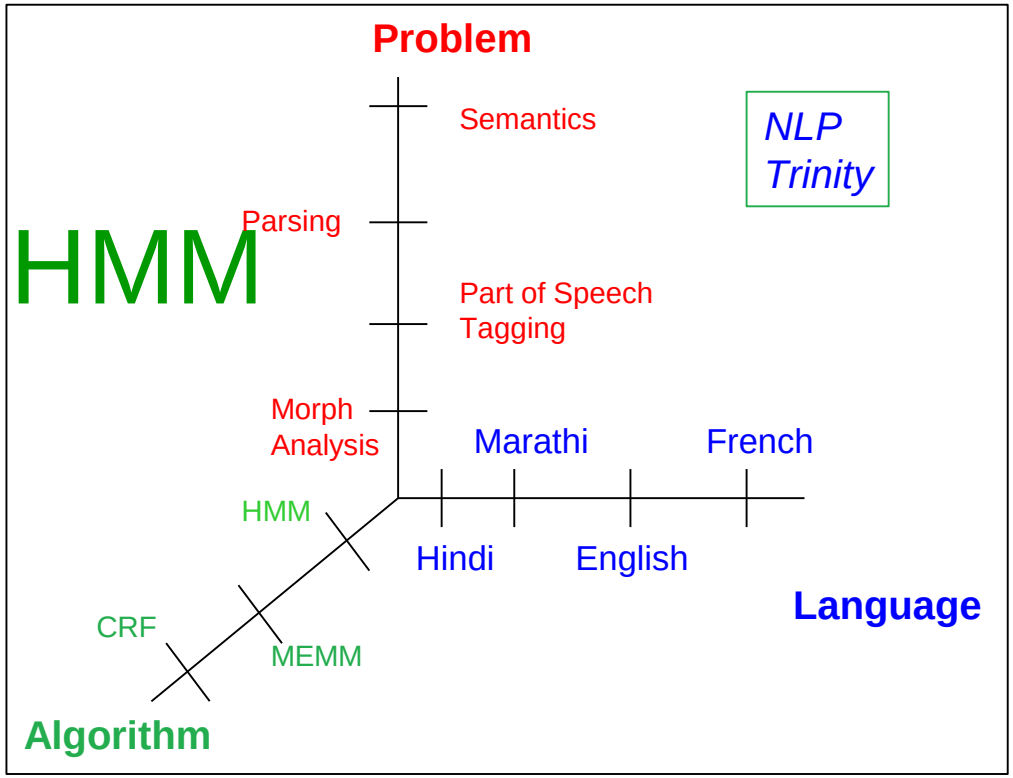
A note on NN and NNS

- Why is there a different tag for plural-NNS?

- Answer: in English most nouns can act as verbs: I watched a *play_noun*; I *play_verb* cricket
- Also, the plural form of nouns coincide with 3rd person, singular number, present tense for verbs

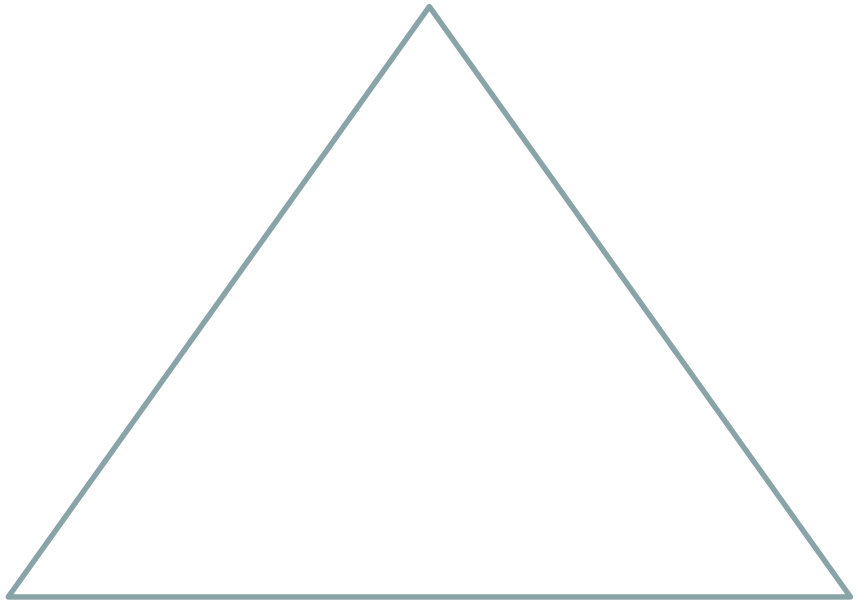
Penn POS TAG Set (cntd)

22.	RBS	Adverb, superlative
23.	RP	Particle
24.	SYM	Symbol
25.	TO	*to*
26.	UH	Interjection
27.	VB	Verb, base form
28.	VBD	Verb, past tense
29.	VBG	Verb, gerund or present participle
30.	VBN	Verb, past participle
31.	VBP	Verb, non-3rd person singular present
32.	VBZ	Verb, 3rd person singular present
33.	WDT	Wh-determiner
34.	WP	Wh-pronoun
35.	WP$	Possessive wh-pronoun
36.	WRB	Wh-adverb


Problem
Semantics
NLP
Trinity
HMM
Parsing
Part of Speech
Tagging
Morph
Analysis
Marathi
French
HMM
Hindi
English
Language
CRF
MEMM
Algorithm

The Trinity

Linguistics

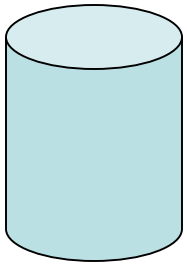


Probability

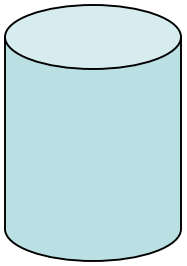
Coding

An Explanatory Example

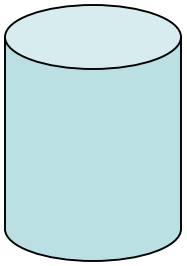
Colored Ball choosing



Urn 1
# of Red = 30
# of Green = 50
# of Blue = 20



Urn 2
# of Red = 10
# of Green = 40
# of Blue = 50



Urn 3
# of Red =60
# of Green =10
# of Blue = 30

Probability of transition to another Urn after picking a ball:

	U_1	U_2	U_3
U_1	0.1	0.4	0.5
U_2	0.6	0.2	0.2
U_3	0.3	0.4	0.3

Example (contd.)

Given :

	U_1	U_2	U_3
U_1	0.1	0.4	0.5
U_2	0.6	0.2	0.2
U_3	0.3	0.4	0.3

and

	R	G	B
U_1	0.3	0.5	0.2
U_2	0.1	0.4	0.5
U_3	0.6	0.1	0.3

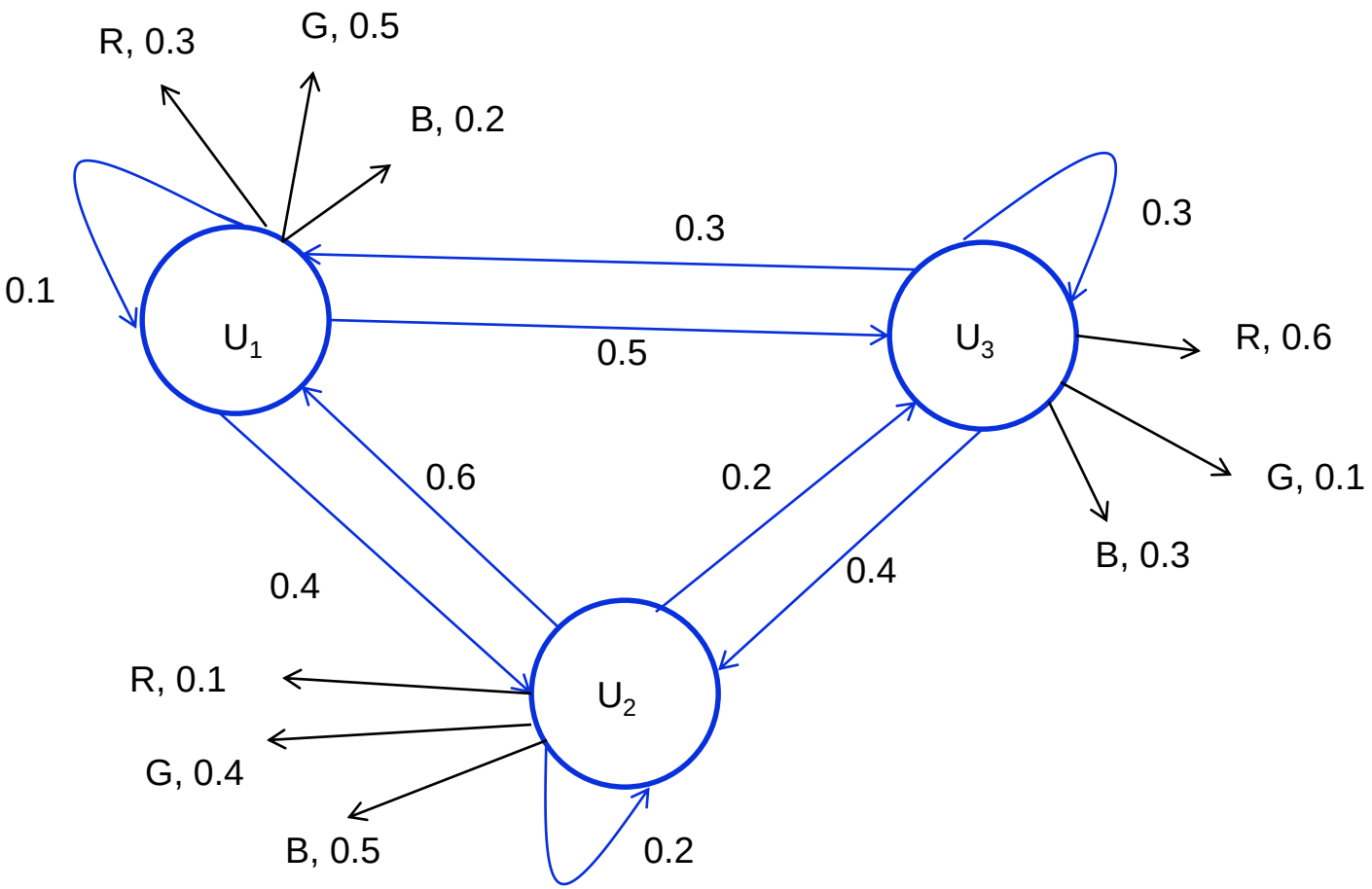
Observation : RRGGBRGR

State Sequence : ??

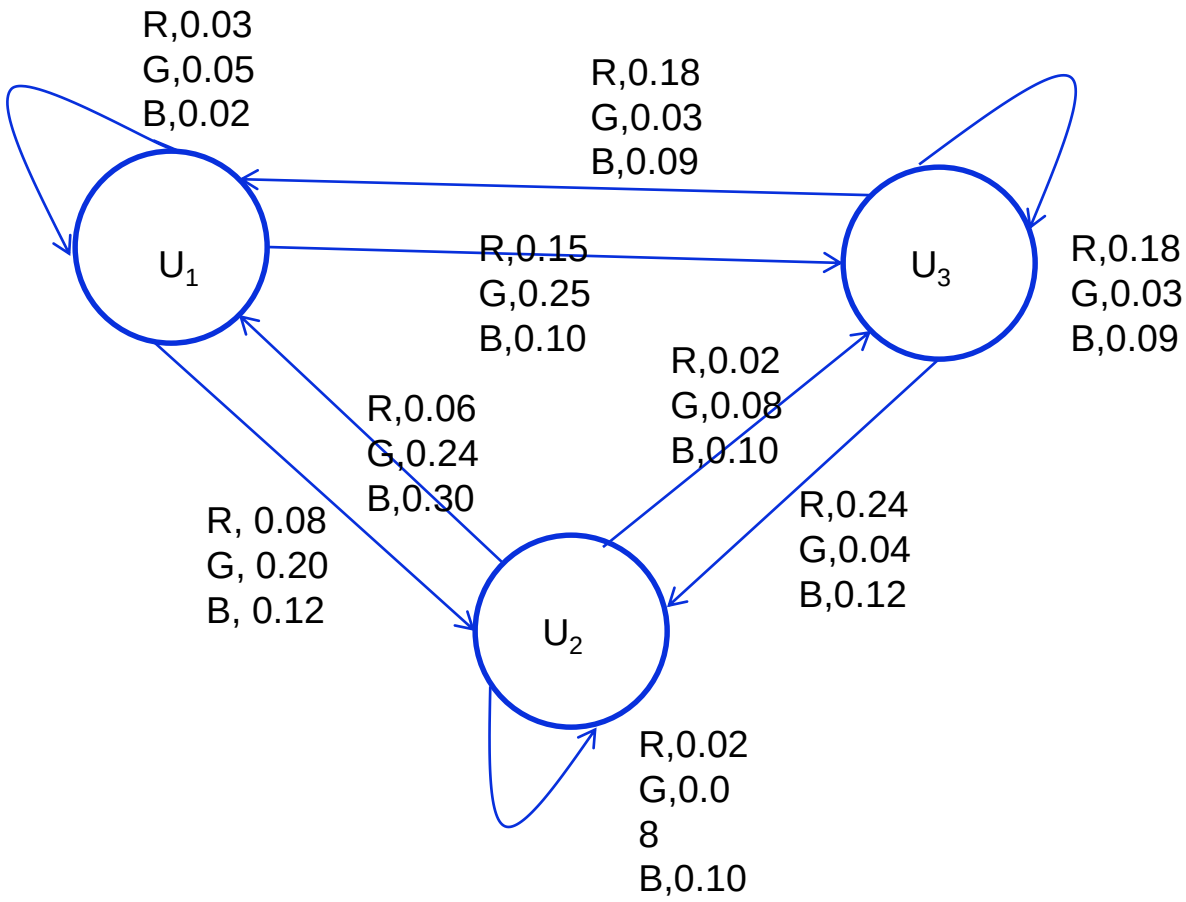
Not so Easily Computable.

There are also initial probabilities of starting A particular urn: 3 probabilities

Diagrammatic representation (1/2)



Diagrammatic representation (2/2)



Classic problems with respect to HMM

1. Given the observation sequence, find the possible state sequences- Viterbi
2. Given the observation sequence, find its probability- forward/backward algorithm
3. Given the observation sequence find the HMM prameters.- Baum-Welch algorithm

Illustration of Viterbi

- The “start” and “end” are important in a sequence.
- Subtrees get eliminated due to the Markov Assumption.

POS Tagset

- N(noun), V(verb), O(other) [simplified]
- ^ (start), . (end) [start & end states]

Illustration of Viterbi

Lexicon

people: N, V

laugh: N, V

.

.

.

Corpora for Training

^ $w_{11}\_t_{11}$ $w_{12}\_t_{12}$ $w_{13}\_t_{13}$ ..... $w_{1k\_1}\_t_{1k\_1}$.

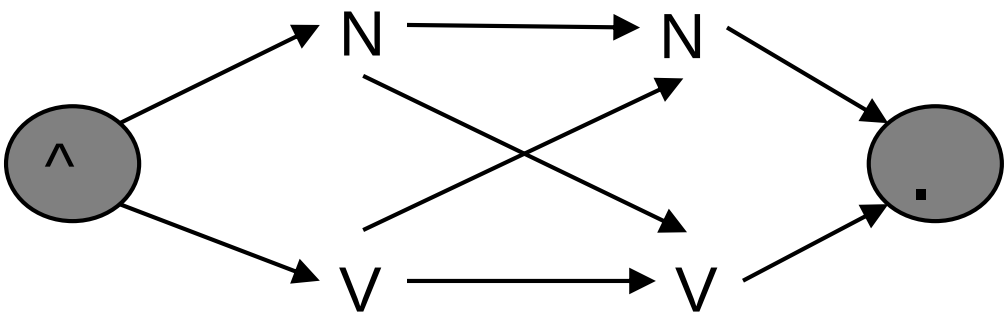
^ $w_{21}\_t_{21}$ $w_{22}\_t_{22}$ $w_{23}\_t_{23}$ ..... $w_{2k\_2}\_t_{2k\_2}$.

.

.

^ $w_{n1}\_t_{n1}$ $w_{n2}\_t_{n2}$ $w_{n3}\_t_{n3}$ ..... $w_{nk\_n}\_t_{nk\_n}$.

Inference



Partial sequence graph

	^	N	V	O	.
^	0	0.6	0.2	0.2	0
N	0	0.1	0.4	0.3	0.2
V	0	0.3	0.1	0.3	0.3
O	0	0.3	0.2	0.3	0.2
.	1	0	0	0	0

This transition table will change from language to language due to language divergences.

Lexical Probability Table

	Є	people	laugh	…	…
^	1	0	0	…	0
N	0	$1x10^{-3}$	$1x10^{-5}$	…	…
V	0	$1x10^{-6}$	$1x10^{-3}$	…	…
O	0	0	$1x10^{-9}$	…	…
.	1	0	0	0	0

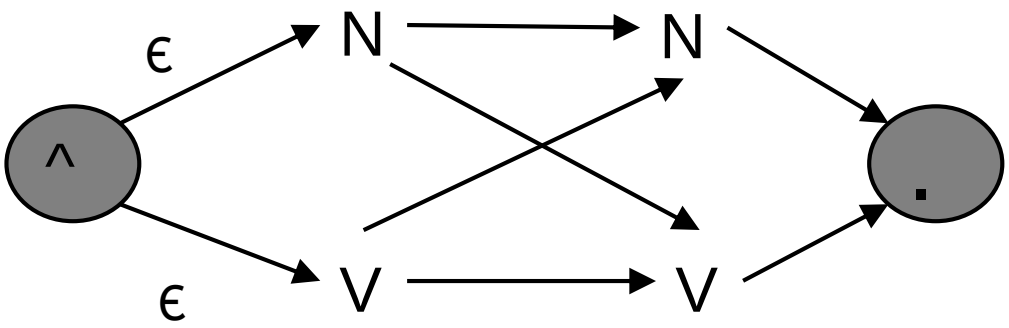
Size of this table = # pos tags in tagset X vocabulary size

vocabulary size = # unique words in corpus

Inference

New Sentence:

^ people laugh .



p(^ N N . | ^ people laugh .)

= (0.6 x 0.1) x (0.1 x 1 x 10^{-3}) x (0.2 x 1 x 10^{-5})

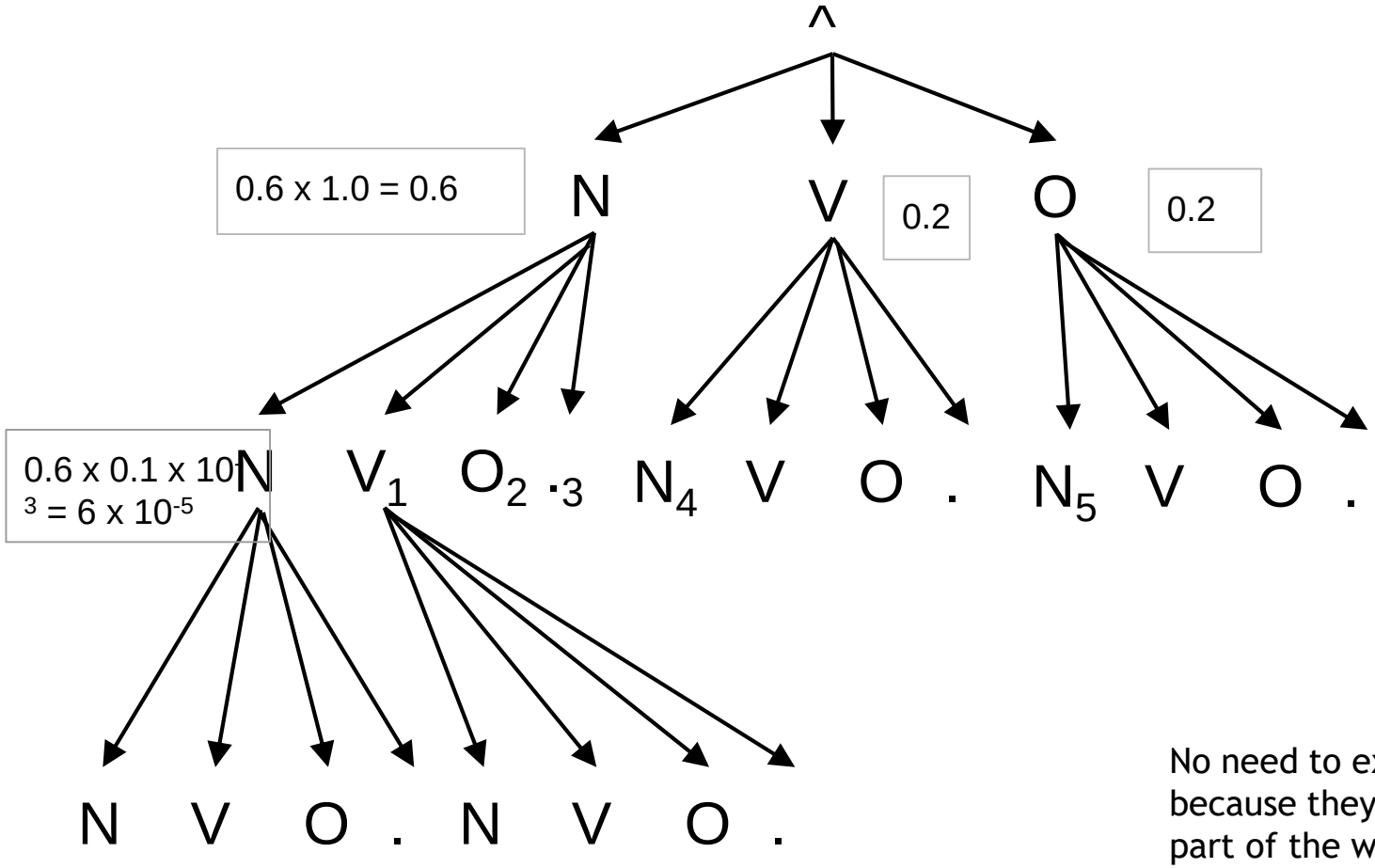
Computational Complexity

- If we have to get the probability of each sequence and then find maximum among them, we would run into exponential number of computations.
- If |s| = #states (tags + ^ + .)

  and |o| = length of sentence (words + ^ + .)

  Then, #sequences = $s^{|o|-2}$
- But, a large number of partial computations can be reused using Dynamic Programming.

Dynamic Programming



ϵ

people

laugh

No need to expand N_4 and N_5 because they will never be a part of the winning sequence.

[1] $0.6 \times 0.4 \times 10^{-3} = 2.4 \times 10^{-4}$

[2] $0.6 \times 0.3 \times 10^{-3} = 1.8 \times 10^{-4}$

[3] $0.6 \times 0.2 \times 10^{-3} = 1.2 \times 10^{-4}$

Computational Complexity

- Retain only those N / V / O nodes which ends in the highest sequence probability.
- Now, complexity reduces from $|s|^{|o|}$ to $|s|.|o|$
- Here, we followed the Markov assumption of order 1.

Back to urn example

Back to the Urn Example

- Here :
  - S = {U1, U2, U3}
  - V = { R,G,B}
- For observation:
  - O ={$o_1 \ldots o_n$}
- And State sequence
  - Q ={$q_1 \ldots q_n$}
- π is $\pi_i = P(q_1 = U_i)$

A =

	U_1	U_2	U_3
U_1	0.1	0.4	0.5
U_2	0.6	0.2	0.2
U_3	0.3	0.4	0.3

B=

	R	G	B
U_1	0.3	0.5	0.2
U_2	0.1	0.4	0.5
U_3	0.6	0.1	0.3

Observations and states

	O_1	O_2	O_3	O_4	O_5	O_6	O_7	O_8
OBS:	R	R	G	G	B	R	G	R
State:	S_1	S_2	S_3	S_4	S_5	S_6	S_7	S_8

$S_i = U_1/U_2/U_3$; A particular state

S: State sequence

O: Observation sequence

S* = “best” possible state (urn) sequence

Goal: Maximize P(S*|O) by choosing “best” S

Goal

- Maximize P(S|O) where S is the State Sequence and O is the Observation Sequence

$$S^* = \arg\max_S (P(S \mid O))$$

False Start

	O_1	O_2	O_3	O_4	O_5	O_6	O_7	O_8
OBS:	R	R	G	G	B	R	G	R
State:	S_1	S_2	S_3	S_4	S_5	S_6	S_7	S_8

$$P(S \mid O) = P(S_{1-8} \mid O_{1-8})$$

$$P(S \mid O) = P(S_1 \mid O).P(S_2 \mid S_1, O).P(S_3 \mid S_{1-2}, O)...P(S_8 \mid S_{1-7}, O)$$

By Markov Assumption (a state depends only on the previous state)

$$P(S \mid O) = P(S_1 \mid O).P(S_2 \mid S_1, O).P(S_3 \mid S_2, O)...P(S_8 \mid S_7, O)$$

Baye's Theorem

$$P(A \mid B) = P(A).P(B \mid A) / P(B)$$

P(A) -: Prior
P(B|A) -: Likelihood

$$\arg\max_S P(S \mid O) = \arg\max_S P(S).P(O \mid S)$$

State Transitions Probability

$$P(S) = P(S_{1-8})$$

$$P(S) = P(S_1).P(S_2 \mid S_1).P(S_3 \mid S_{1-2}).P(S_4 \mid S_{1-3})...P(S_8 \mid S_{1-7})$$

By Markov Assumption (k=1)

$$P(S) = P(S_1).P(S_2 \mid S_1).P(S_3 \mid S_2).P(S_4 \mid S_3)...P(S_8 \mid S_7)$$

Observation Sequence probability

$$P(O|S) = P(O_1|S_{1-8}).P(O_2|O_1, S_{1-8}).P(O_3|O_{1-2}, S_{1-8})...P(O_8|O_{1-7}, S_{1-8})$$

Assumption that ball drawn depends only on the Urn chosen

$$P(O|S) = P(O_1|S_1).P(O_2|S_2).P(O_3|S_3)...P(O_8|S_8)$$

$$P(S|O) = P(S).P(O|S)$$

$$P(S|O) = P(S_1).P(S_2|S_1).P(S_3|S_2).P(S_4|S_3)...P(S_8|S_7).$$

$$P(O_1|S_1).P(O_2|S_2).P(O_3|S_3)...P(O_8|S_8)$$

Grouping terms

	O_0	O_1	O_2	O_3	O_4	O_5	O_6	O_7	O_8	
Obs:	ε	R	R	G	G	B	R	G	R	
State:	S_0	S_1	S_2	S_3	S_4	S_5	S_6	S_7	S_8	S_9

$P(S).P(O|S)$

$= [P(O_0|S_0).P(S_1|S_0)].$
$[P(O_1|S_1). \quad P(S_2|S_1)].$
$[P(O_2|S_2). \quad P(S_3|S_2)].$
$[P(O_3|S_3).P(S_4|S_3)].$
$[P(O_4|S_4).P(S_5|S_4)].$
$[P(O_5|S_5).P(S_6|S_5)].$
$[P(O_6|S_6).P(S_7|S_6)].$
$[P(O_7|S_7).P(S_8|S_7)].$
$[P(O_8|S_8).P(S_9|S_8)].$

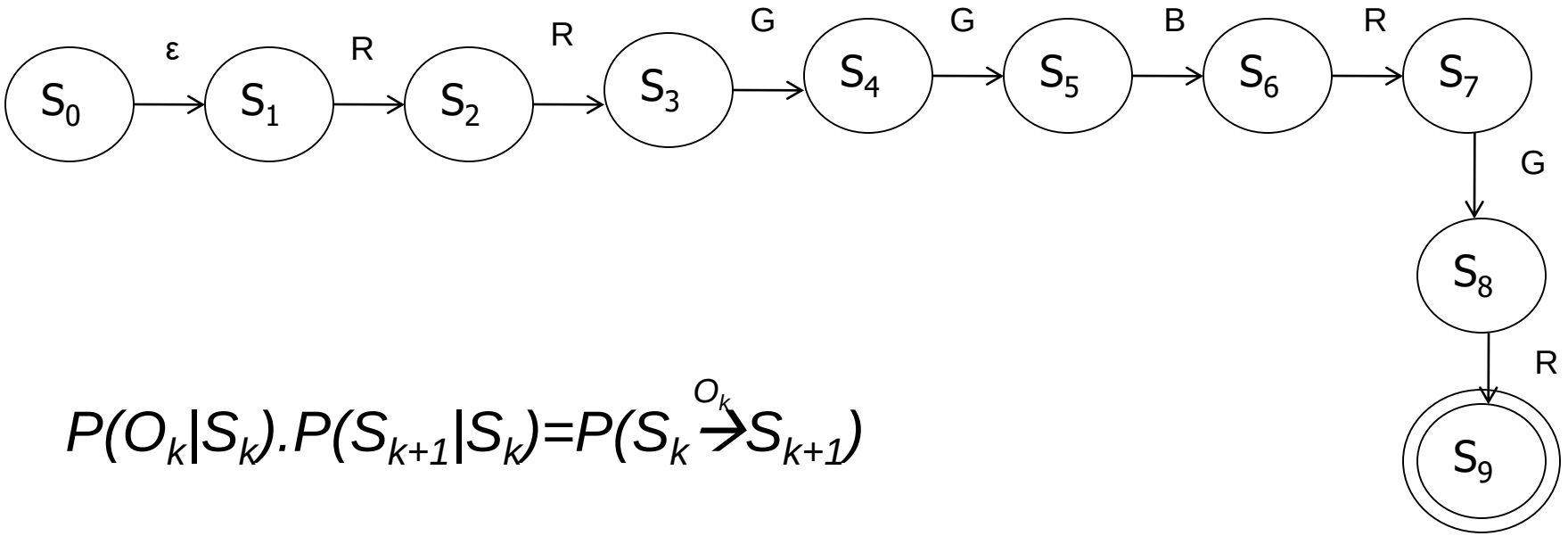
We introduce the states S_0 and S_9 as initial and final states respectively.

After S_8 the next state is S_9 with probability 1, i.e., $P(S_9|S_8)=1$

O_0 is ε-transition

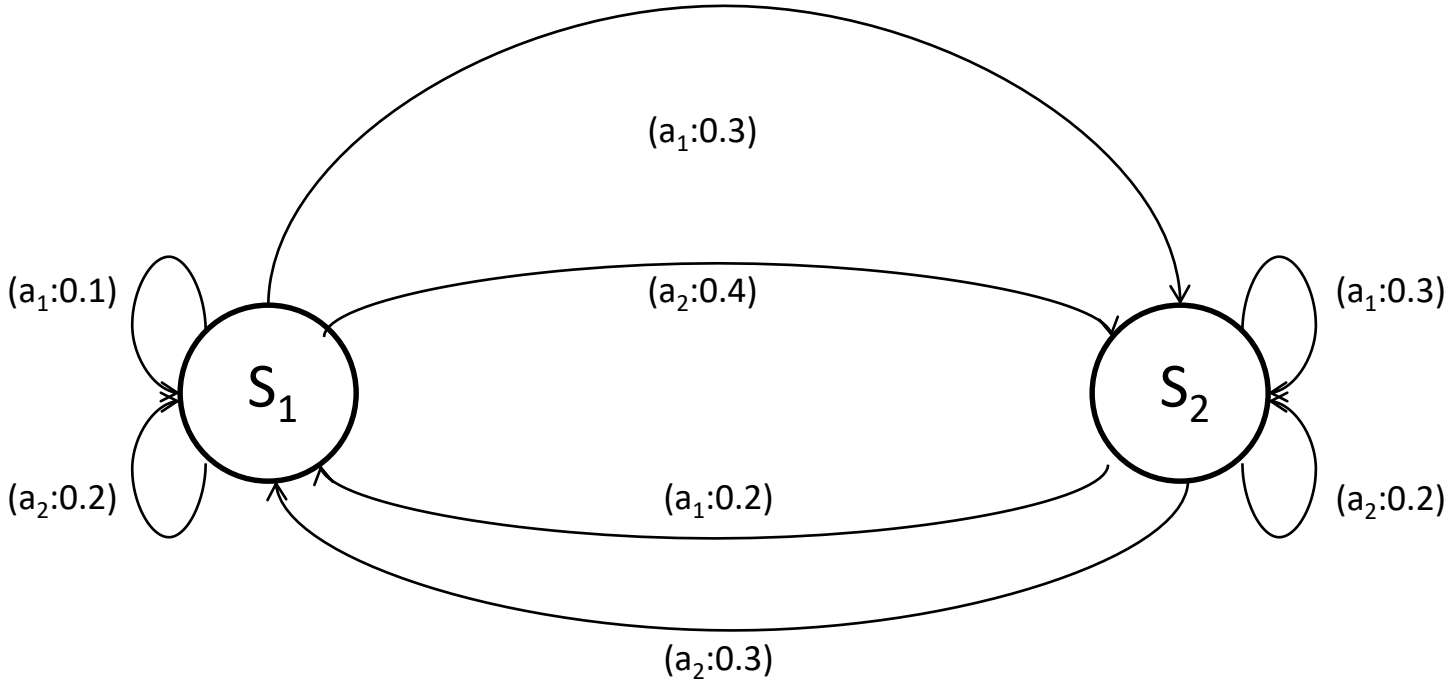
Introducing useful notation

	O_0	O_1	O_2	O_3	O_4	O_5	O_6	O_7	O_8	
Obs:	ε	R	R	G	G	B	R	G	R	
State:	S_0	S_1	S_2	S_3	S_4	S_5	S_6	S_7	S_8	S_9



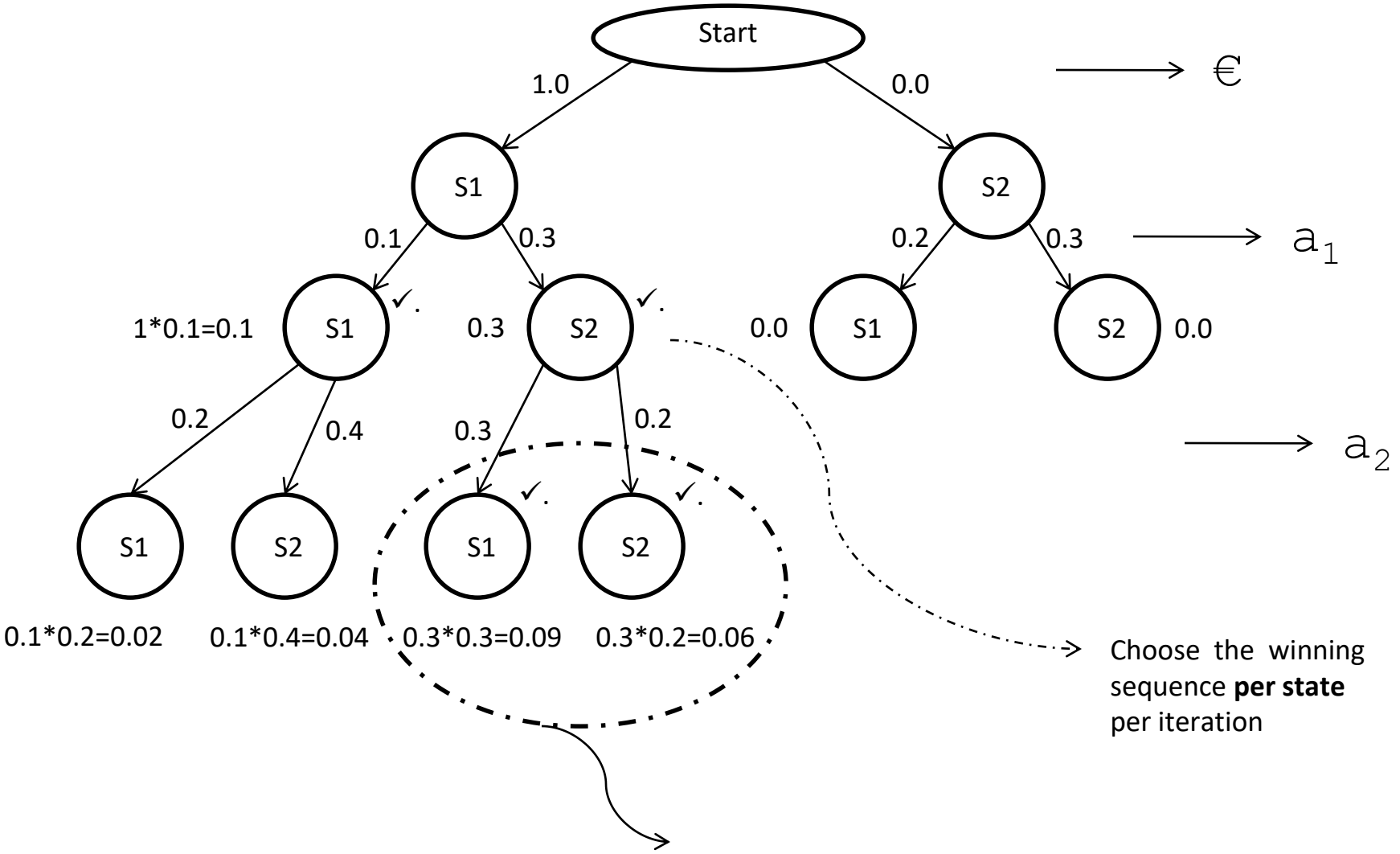
$$P(O_k|S_k).P(S_{k+1}|S_k)=P(S_k \xrightarrow{O_k} S_{k+1})$$

Probabilistic FSM

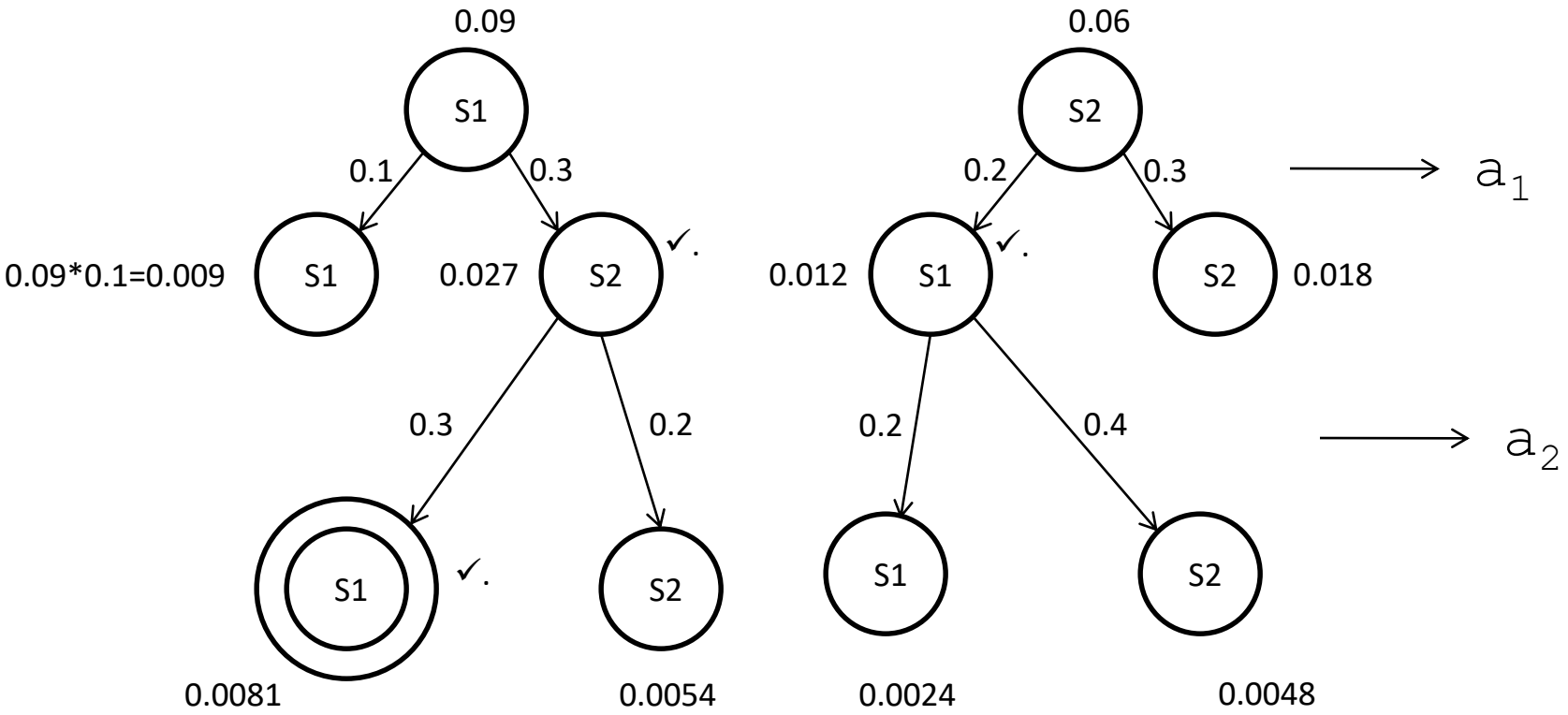


The question here is:
"what is the most likely state sequence given the output sequence seen"

Developing the tree



Tree structure contd…



The problem being addressed by this tree is $S^* = \arg\max_s P(S \mid a_1 - a_2 - a_1 - a_{2,\mu})$

a1-a2-a1-a2 is the output sequence and μ the model or the machine

POS tagging to be contd.