CS626: Speech, NLP and Web

POS Tagging

Pushpak Bhattacharyya
Computer Science and Engineering Department
IIT Bombay

Week 2 of 5th August, 2024

1-slide recap

- Turing Test and Chinese Room Experiment
- Definition of NLP: art science and tech of NLU and NLG
- LINGUISTICS+PROBABILITY= NLP
- LLMs: what do they predict- language object and properties
- Comparisons with humans: size, energy requirement, carbon footprint
- Well known LLMs, MLMs, SLMs
- NL Stack

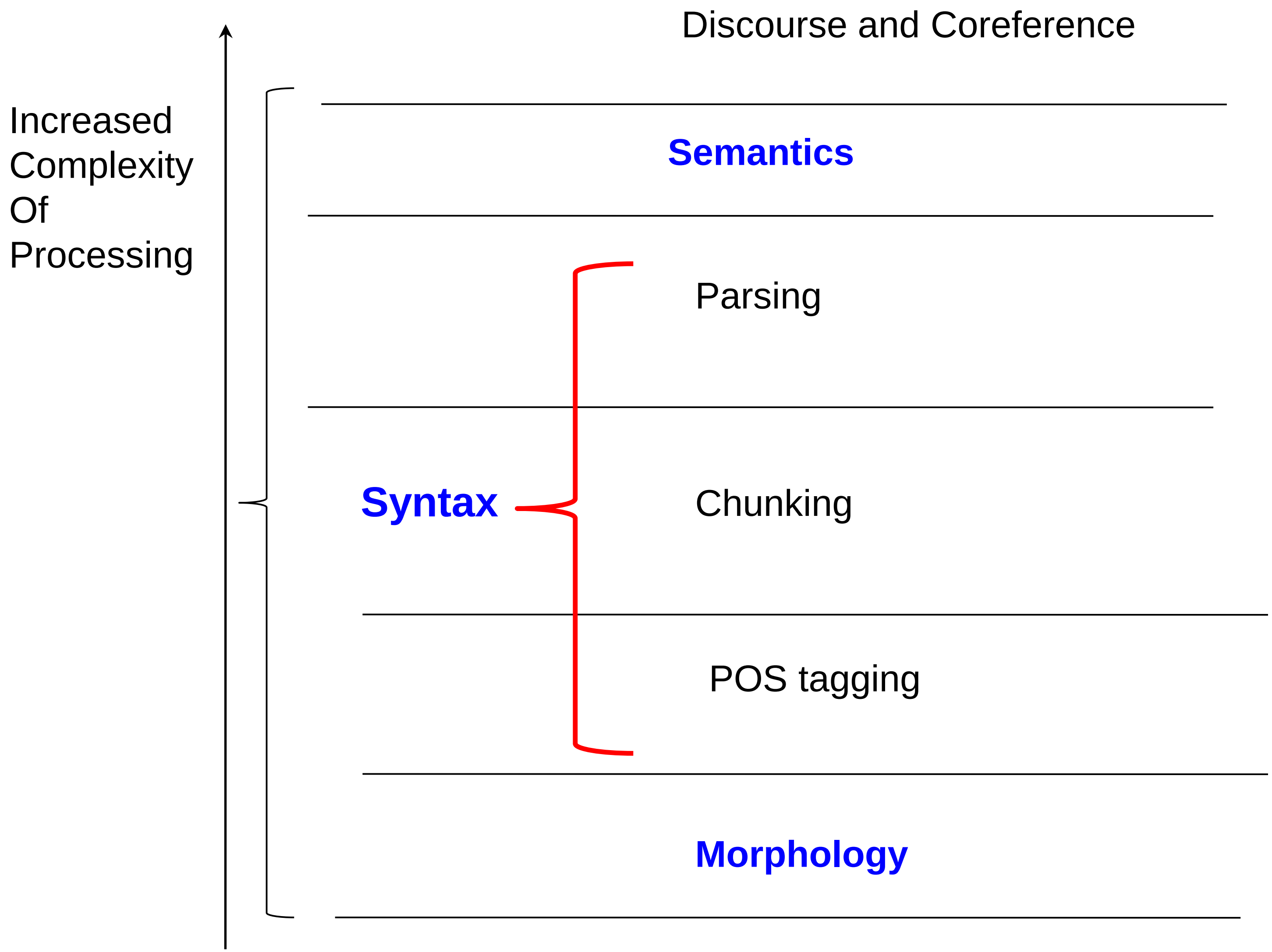
Assignments Plan

- End of August: (a) HMM based POS Tagging; (b) CRF Based POS tagging
  - E.g., Input: *People Laugh* → Output: *NN VB*
- End of September: (a) SVM based Named Entity Identifier (name or noname); (b) Logistic Regression based Named Entity Identifier
  - E.g., Input: *John laughs* → Output: *1 0*
- End of October: Transliteration from Indian Language to English; statistical
  - E.g., Input: *मुंबई* → Output: *mumbai*; *மும்பை* → *mumbai*
- (https://www.cse.iitb.ac.in/~pb/papers/icon20-xlit.pdf)
- MUST: compare performance with ChatGPT (Quantitative)

Project Plan

- Can and should be ambitious
- Core should be ONE paper and then some value addition of your own
- Suggested topics:
  - Multimodal emotion recognition
  - Mental health monitoring
  - Multimodal metaphor, simile, hyperbole…
  - Language mixed questions answering
  - OCR→MT→TTS pipeline
  - Physics driven NLP: application of chaos theory, implementation to demonstrate emergence of language
  - Hybrid of statistical and neural systems in low resource scenarios with applications
  - And anything you find interesting

NLP Layers

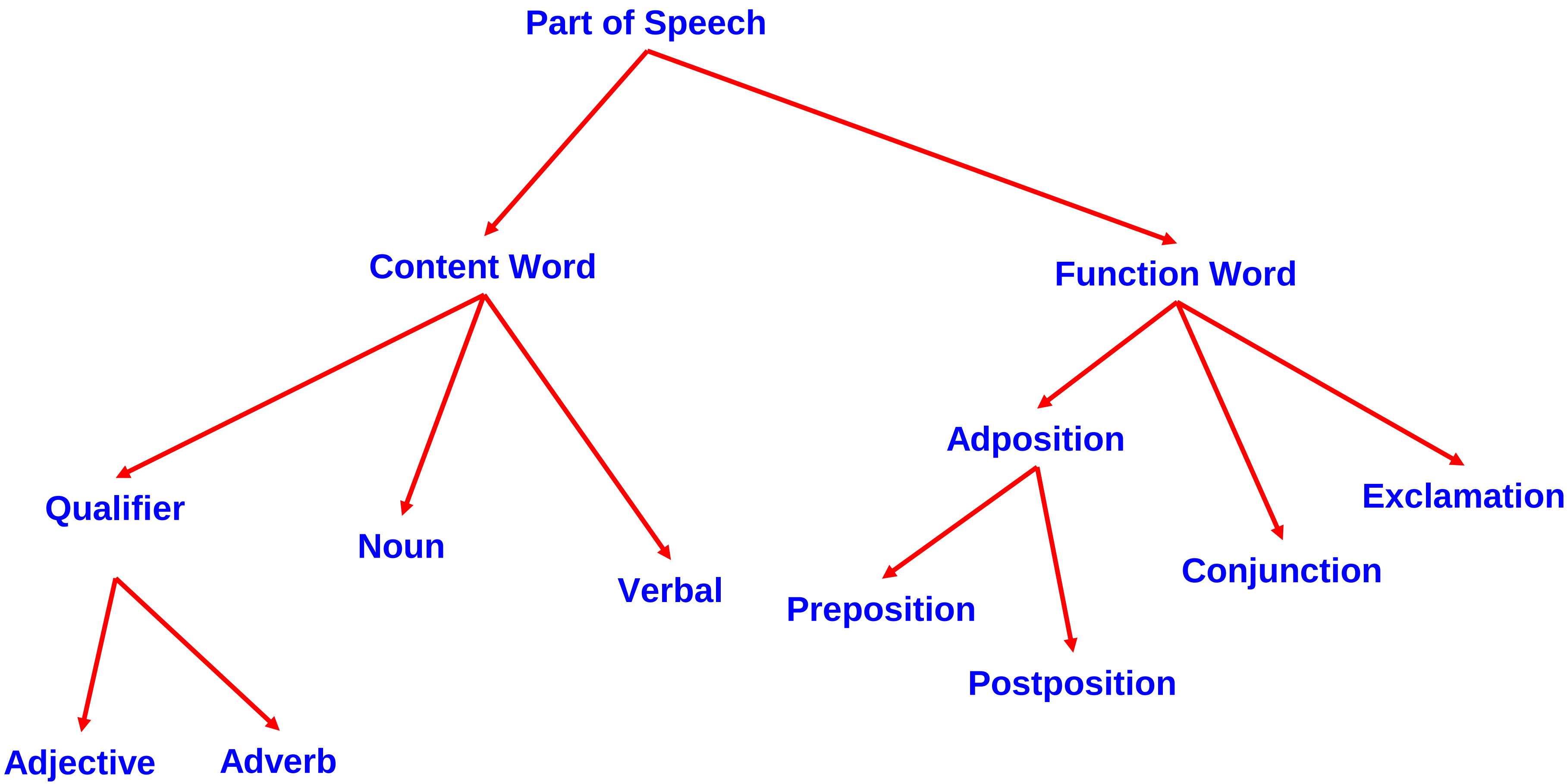

Discourse and Coreference
Increased
Complexity
Of
Processing
Semantics
Parsing
Syntax
Chunking
POS tagging
Morphology

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



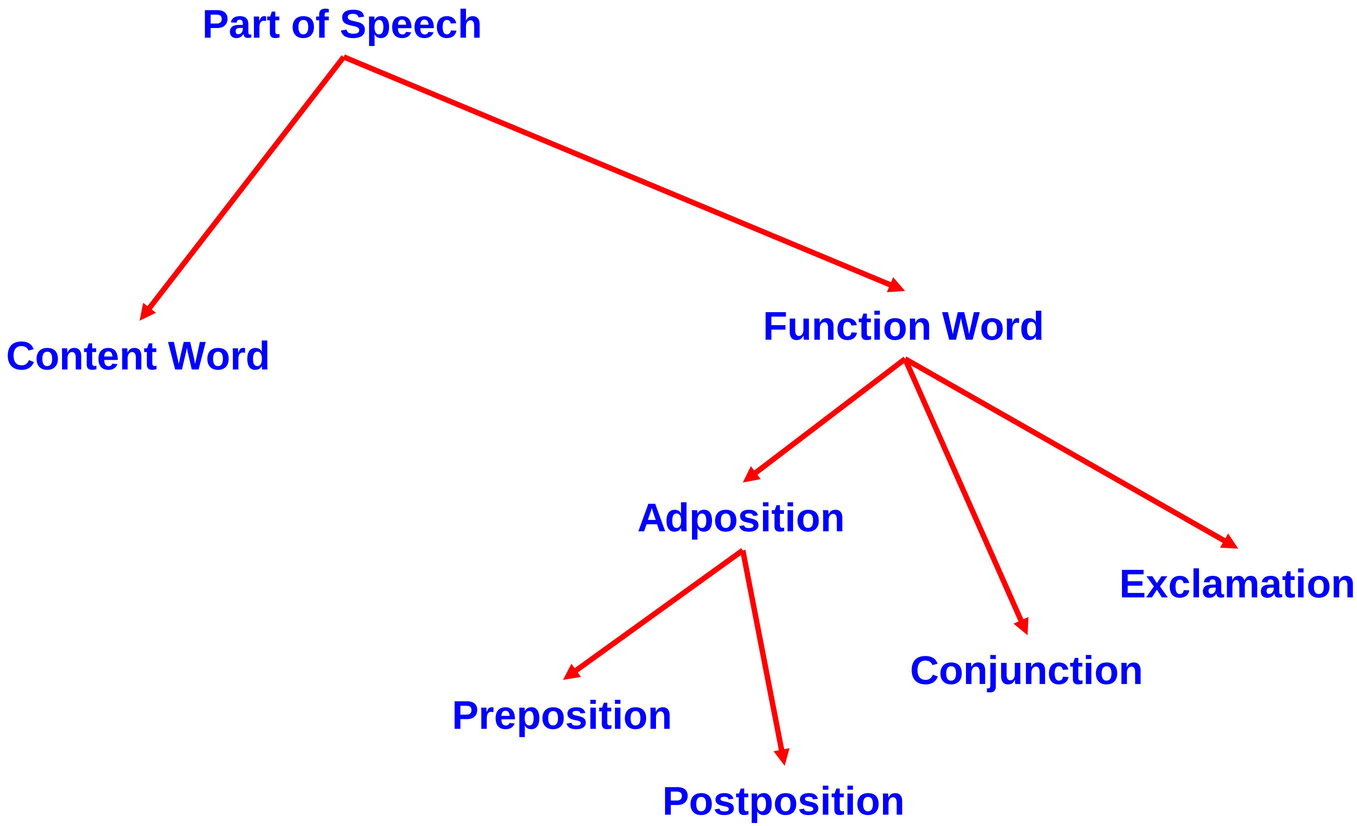
Importance of POS tags

- POS information is crucial for search engines
- Queries contain a set of nouns typically
- Verbs and qualifiers are dropped
- Function words are dropped as stop words
- E.g. *Qutub Minar Delhi* (all these words are nouns)

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

POS ambiguity instances

best ADJ ADV NP V
better ADJ ADV V DET

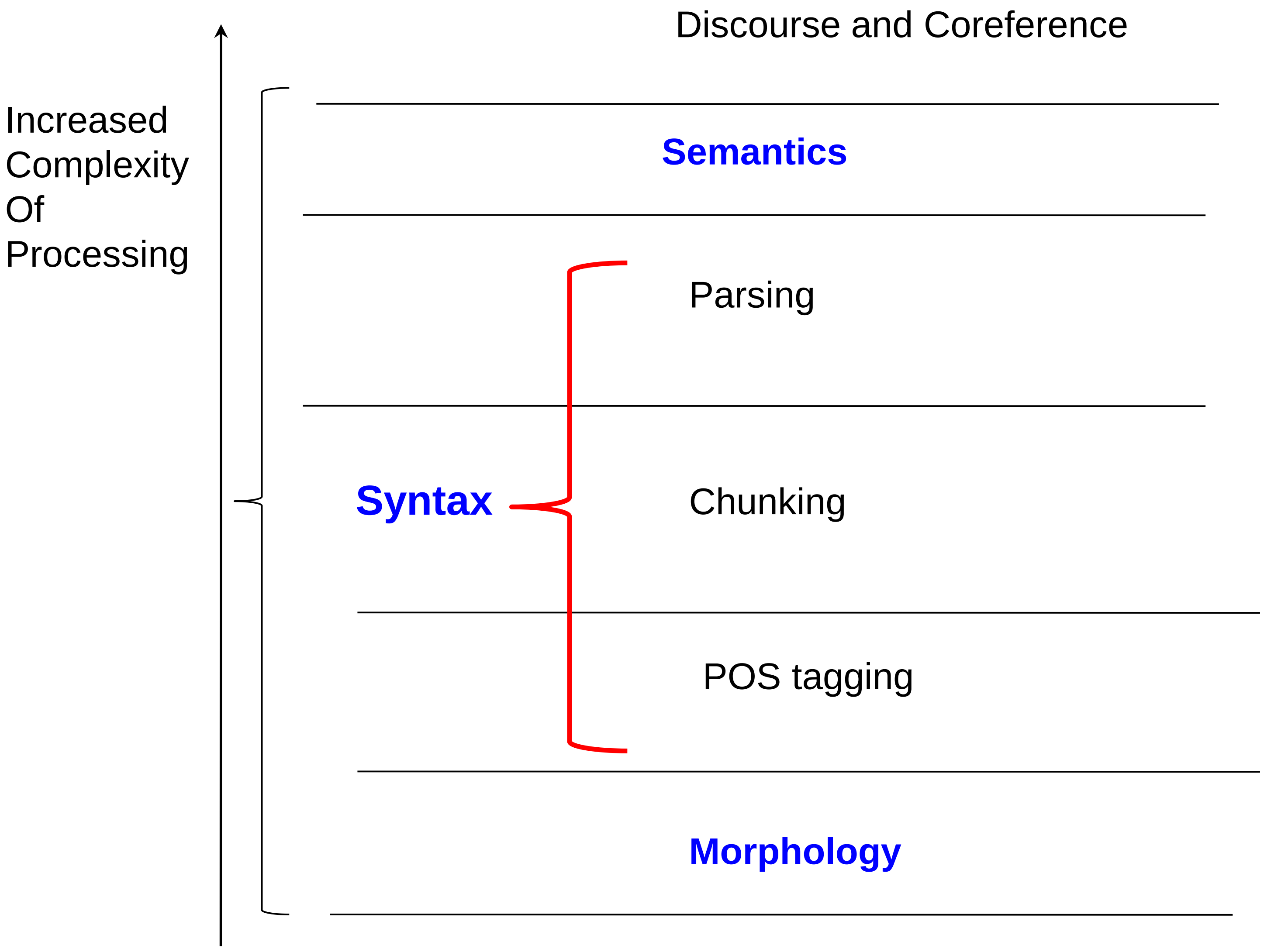
close ADV ADJ V N (*running close to the competitor, close escape, close the door, towards the close of the play*)

cut V N VN VD
even ADV DET ADJ V
grant NP N V -
hit V VD VN N
lay ADJ V NP VD
left VD ADJ N VN
like CNJ V ADJ P -
near P ADV ADJ DET
open ADJ V N ADV
past N ADJ DET P
present ADJ ADV V N
read V VN VD NP
right ADJ N DET ADV
second NUM ADV DET N
set VN V VD N -
that CNJ V WH DET

An interesting sentence and its analysis on the NLP Stack- called NLU

"To bank, I bank on the bank on the river bank"

NLP Layers


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

POS Tags- "To bank, I bank on the bank on the river bank"

- To (IN - Preposition)
- bank (VB - Verb, base form)
- , (PUNCT - Punctuation)
- I (PRP - Pronoun)
- bank (VBP - Verb, non-3rd person singular present)
- on (IN - Preposition)
- the (DT - Determiner)
- bank (NN - Noun, singular)
- on (IN - Preposition)
- the (DT - Determiner)
- river (NN - Noun, singular)
- bank (NN - Noun, singular)

Constituency Parse- "To bank, I bank on the bank on the river bank"

Release Justi ✕ | (539) WhatsApp ✕ | Google Calendar - ✕ | ChatGPT ✕ | AMS IIT Bombay ✕ | +

/c/ac0d13de-847d-4f17-9db0-64e38d077703

ChatGPT

Here is the constituency parse tree for the sentence "To bank, I bank on the bank on the river bank":

scss Copy code

```
(S
  (PP (TO To) (NP (NN bank)))
  (, ,)
  (S
    (NP (PRP I))
    (VP (VBP bank)
      (PP (IN on)
        (NP
          (NP (DT the) (NN bank))
          (PP (IN on)
            (NP (DT the) (NN river) (NN bank))))))))
```

Dependency Parse- "To bank, I bank on the bank on the river bank"

natsApp | CS 626 (2024) Stati | Google Calendar - | ChatGPT | AMS IIT Bombay

/ac0d13de-847d-4f17-9db0-64e38d077703

ChatGPT

Here is the dependency parse for the sentence "To bank, I bank on the bank on the river bank":

scss | Copy code

```scss
root(ROOT, bank-4)
mark(bank-4, To-1)
punct(bank-4, ,-2)
nsubj(bank-4, I-3)
prep_on(bank-4, bank-6)
det(bank-6, the-5)
prep_on(bank-6, bank-10)
det(bank-10, the-7)
compound(bank-10, river-8)
compound(bank-10, bank-9)
```

Here's a breakdown:

Get ci

Dependency parsing- "To_1 bank_2, I_3 bank on the bank on the river bank“ (cntd)

bank-4 is the root of the sentence.
To-1 is a marker (mark) for bank-4.
,-2 is punctuation (punct) related to bank-4.
I-3 is the nominal subject (nsubj) of bank-4.
bank-6 is the object of the preposition (prep_on) on that relates to bank-4.
the-5 is the determiner (det) of bank-6; **bank-10** is the object of the preposition (prep_on) on that relates to bank-6.
the-7 is the determiner (det) of bank-10; **river-8** is the first compound modifier (compound) of bank-10.
bank-9 is the second compound modifier (compound) of bank-10.

Semantic Roles- "To bank, I bank on the bank on the river bank"

To bank:

Purpose (ARGM-PNC): "To bank"

I bank:

Agent (ARG0): "I" (the doer of the action)

Verb (V): "bank" (the main action)

*on the bank on the river b*ank:

Location (ARGM-LOC): "on the bank on the river bank" (where the action is happening)

Bank (ARG1): "the bank" (the first location object)

River bank (ARG1): "the river bank" (the secondary location object, specifying the first)

Pragmatics- "To bank, I bank on the bank on the river bank“ (1/2)

Speech Act:

The speaker is likely making an assertion about their reliance on a specific place or institution, with a playful twist.

Inference:

The listener is expected to recognize the wordplay and understand that the same word "bank" has multiple meanings in this context. The speaker's choice to use all these meanings in one sentence suggests an intention to amuse or engage in linguistic creativity.

Pragmatics- "To bank, I bank on the bank on the river bank“ (2/2)

Implicature:

The sentence implies a clever use of language. The speaker might be highlighting their wit or drawing attention to the complexities of the English language.

Contextual Appropriateness:

This sentence would be appropriate in a context where wordplay or clever use of language is appreciated, such as in a casual conversation, a language game, or a literary context.

In summary, pragmatically, the sentence leverages the multiple meanings of "bank" to create a humorous or clever effect, relying on the listener to recognize and appreciate the wordplay.

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

ML Based POS Tagging

Noisy Channel Model

W *Noisy Channel* **T**

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

Sequence *W* is transformed into sequence *T*

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

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

Bayes Theorem

- $P(B|A)=[P(B).P(A|B)]/P(A)$
- $P(B|A)$: Posterior Probability
- $P(B)$: Prior
- $P(A|B)$: Likelihood

- Should we work with the LHS or the RHS?

Cancer Detection vs. Visa Granting

- *$P(B|A)=[P(B).P(A|B)]/P(A)$*
- *Cancer, if P(Cancer) > P(~cancer), i.e. P(cancer)>0.5*

- *Grant_Visa, if P(Grant_Visa) > P(~Grant_Visa), i.e. P(Grant_Visa)>0.5*

- Key consideration- which data is more reliable for obtaining probabilities

Probabilities involved: cancer

- *Posterior probability P(Cancer|patient_parameters),*
  - *Patient parameters: HBC count, weight, family history etc.*
- *Prior probability P(cancer)*
- *Likelihood P(patient_parameters| Cancer)*

Probabilities involved: visa

- *Posterior probability P(Grant_Visa|candidate_features)*
  - *Candidate features: income, education, travel history, etc.*
- *Prior probability P(Grant_Visa)*
- Likelihood *P(candidate_features|Grant_Visa)*

POS of “The sun is shining”

$$P(DT\ \ NN\ VBZ\ VBG\,|\,"The\ sun\ is\ shining")$$

$$=\frac{\#<DT\ \ NN\ VBZ\ VBG,"The\ sun\ is\ shining">}{\#The\ sun\ is\ shining}$$

Posterior probability

Mathematics of POS tagging

Argmax computation (1/2)

Best tag sequence

= T*

= argmax P(T|W)

= argmax P(T)P(W|T) (by Baye's Theorem)

$$\mathbf{P(T) = P(t_0{=}\hat{}\ t_1 t_2 \ldots t_{n+1}{=}.)}$$
$$\mathbf{= P(t_0)P(t_1|t_0)P(t_2|t_1 t_0)P(t_3|t_2 t_1 t_0) \ldots P(t_n|t_{n-1} t_{n-2} \ldots t_0)P(t_{n+1}|t_n t_{n-1} \ldots t_0)}$$
$$\mathbf{= P(t_0)P(t_1|t_0)P(t_2|t_1) \ldots P(t_n|t_{n-1})P(t_{n+1}|t_n)}$$
$$= \prod_{i=0}^{N+1} \mathbf{P(t_i|t_{i-1})}$$

Bigram Assumption

Argmax computation (2/2)

$$P(W|T) = P(w_0|t_0\text{-}t_{n+1})P(w_1|w_0t_0\text{-}t_{n+1})P(w_2|w_1w_0t_0\text{-}t_{n+1}) \ldots P(w_n|w_0\text{-}w_{n-1}t_0\text{-}t_{n+1})P(w_{n+1}|w_0\text{-}w_nt_0\text{-}t_{n+1})$$

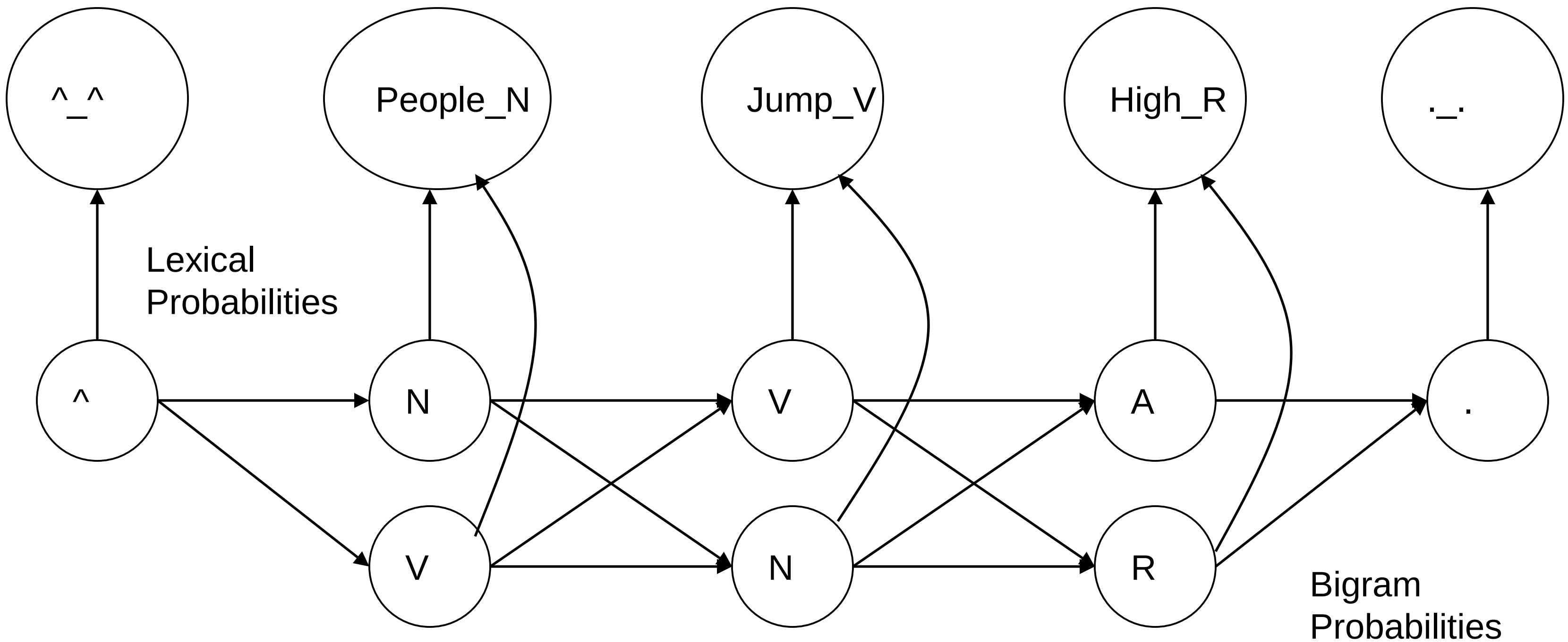
Assumption: A word is determined completely by its tag. This is inspired by speech recognition

$$= P(w_o|t_o)P(w_1|t_1) \ldots P(w_{n+1}|t_{n+1})$$

$$= \prod_{i=0}^{n+1} P(w_i|t_i)$$

$$= \prod_{i=1}^{n+1} P(w_i|t_i)$$ (Lexical Probability Assumption)

Generative Model

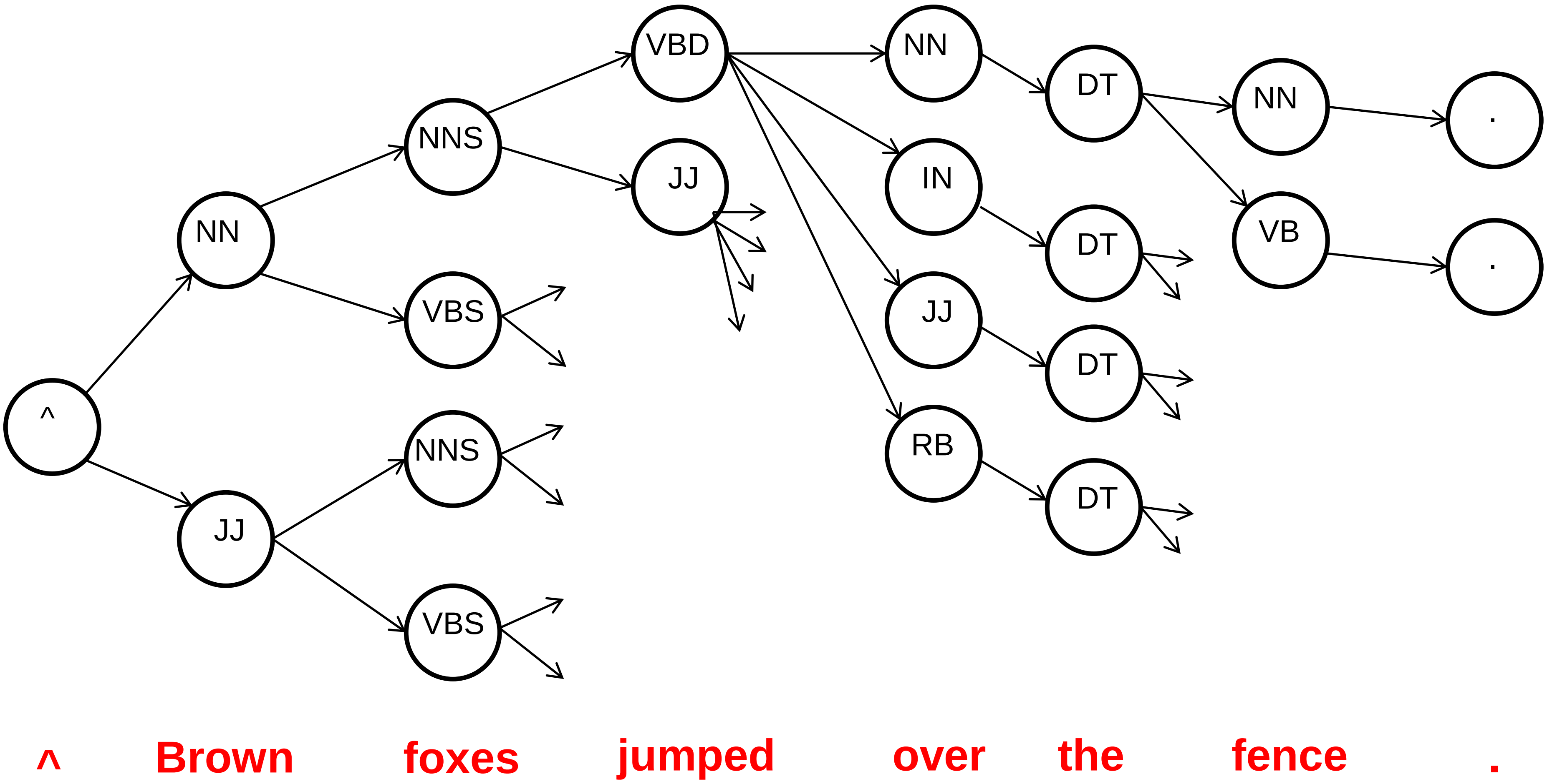


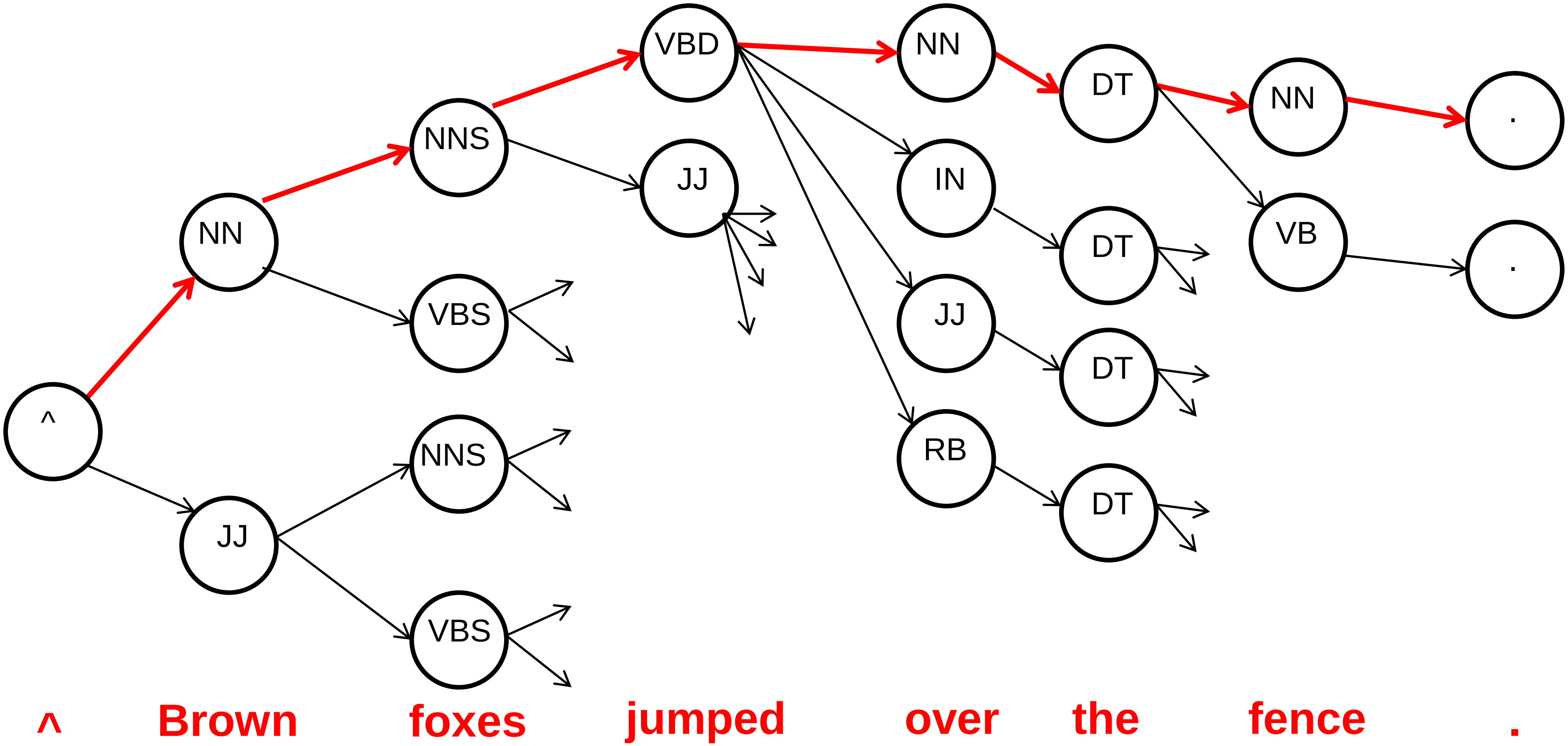
This model is called Generative model.
Here words are observed from tags as states.
This is similar to HMM.

Computation of POS tags

DECODING

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





Probability of a path (e.g. Top most path) = *P*(T) * *P*(W|T)

P(^) . P(NN|^) . P(NNS|NN) . P(VBD|NNS) . P(NN|VBD) . P(DT|NN) . P(NN|DT) . P(.|NN) . P(.)

*

P(^|^) . P(brown|NN) . P(foxes|NNS) . P(jumped|VBD) . P(over|NN) . P(the|DT) . P(fence|NN) . P(.|.)

Questions?

- Where do tags come from?
  - Tag set
- How to get probability values i.e. P(.)?
  - Annotated corpora

After modeling of the problem, emphasis should be on the corpus

Computing P(.) values

Let us suppose annotated corpus has the following sentence

I	**have**	**a**	**brown**	**bag**	**.**
PRN	**VB**	**DT**	**JJ**	**NN**	**.**

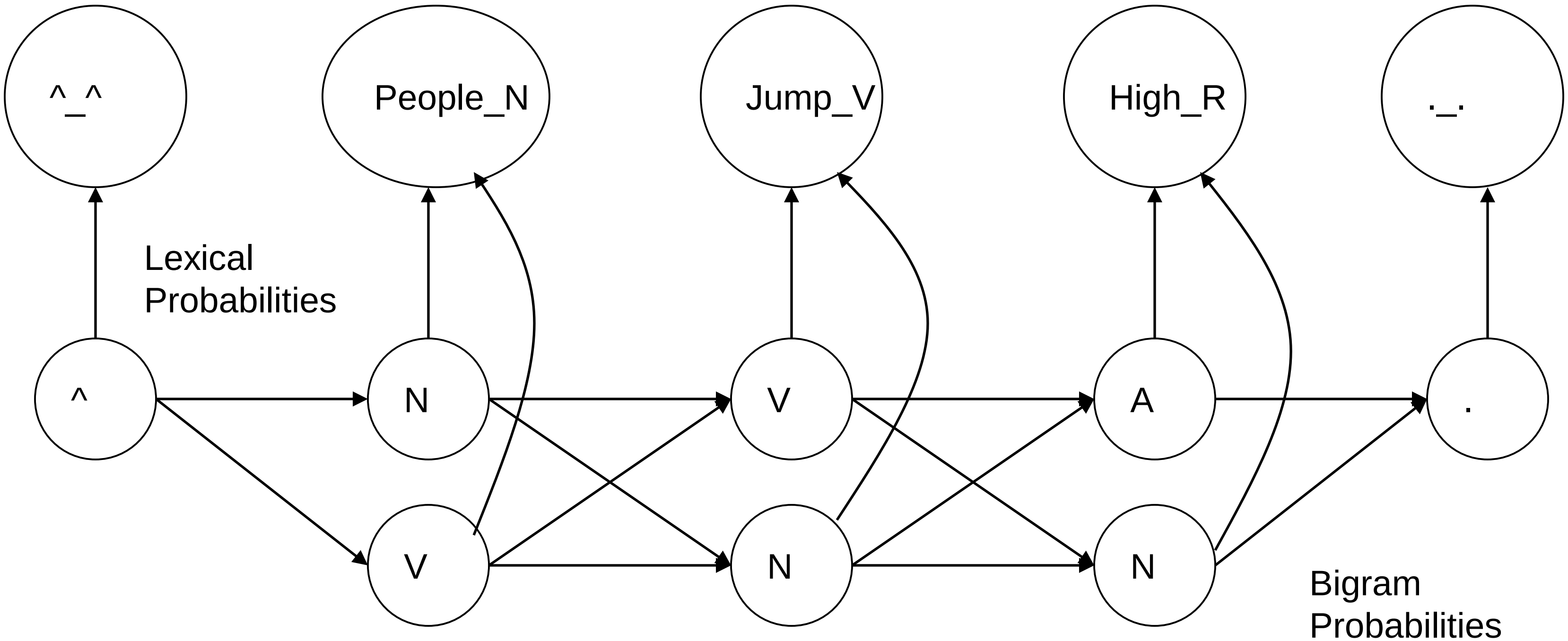
$$P(NN \mid JJ) = \frac{Number\_of\_times\_JJ\_followed\_by\_NN}{Number\_of\_times\_JJ\_appeared}$$

$$P(Brown \mid JJ) = \frac{Number\_of\_times\_Brown\_tagged\_as\_JJ}{Number\_of\_times\_JJ\_appeared}$$

Why Ratios?

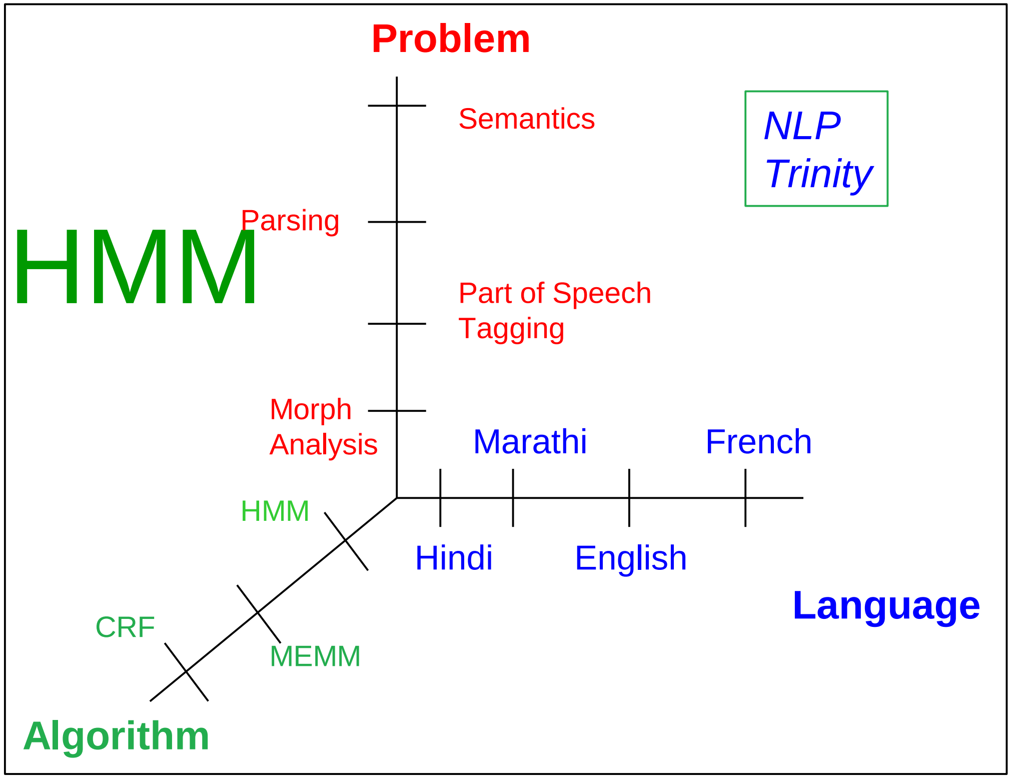
- This way of computing parameter probabilities: is this correct?
- What does "correct" mean?
- Is this principled?
- We are using Maximum Likelihood Estimate (**MLE**)
- Assumption: underlying distribution is multinomial

HMM: Generative Model



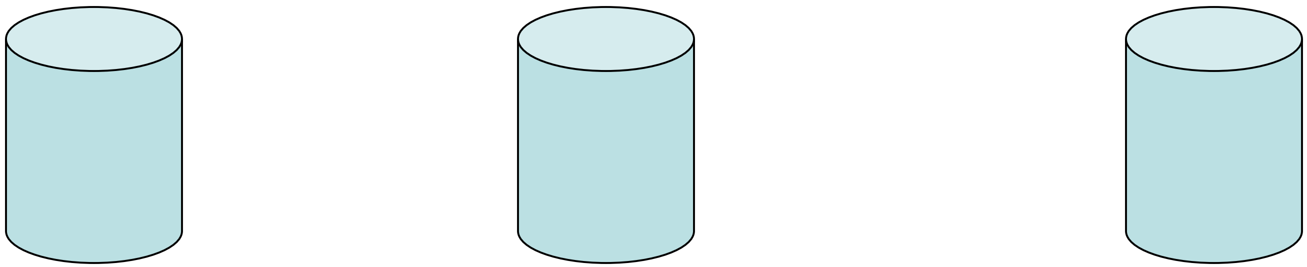
This model is called Generative model.
Here words are observed from tags as states.
This is similar to HMM.

HMM


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

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

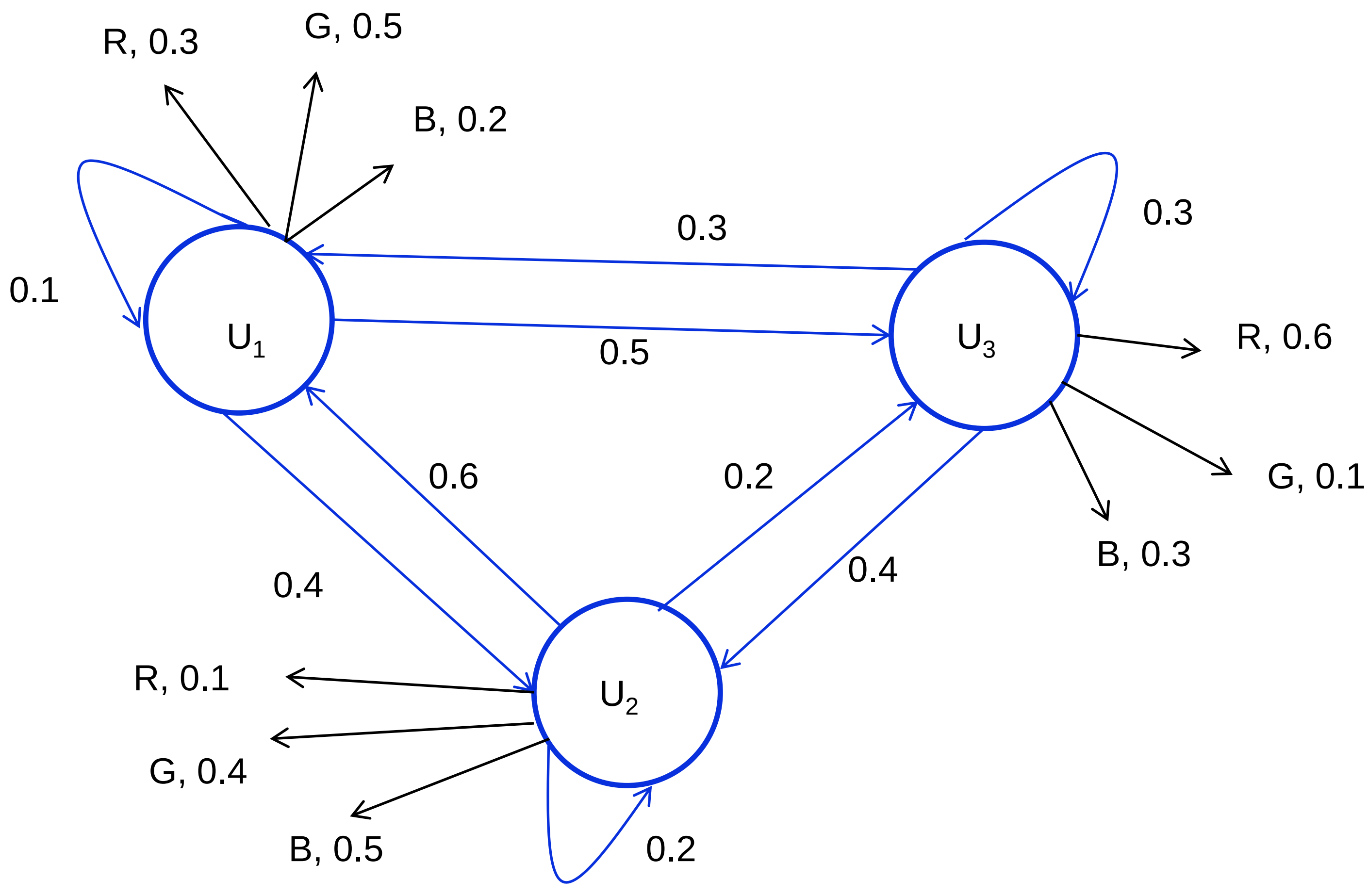
Observation : RRGGBRGR

State Sequence : ??

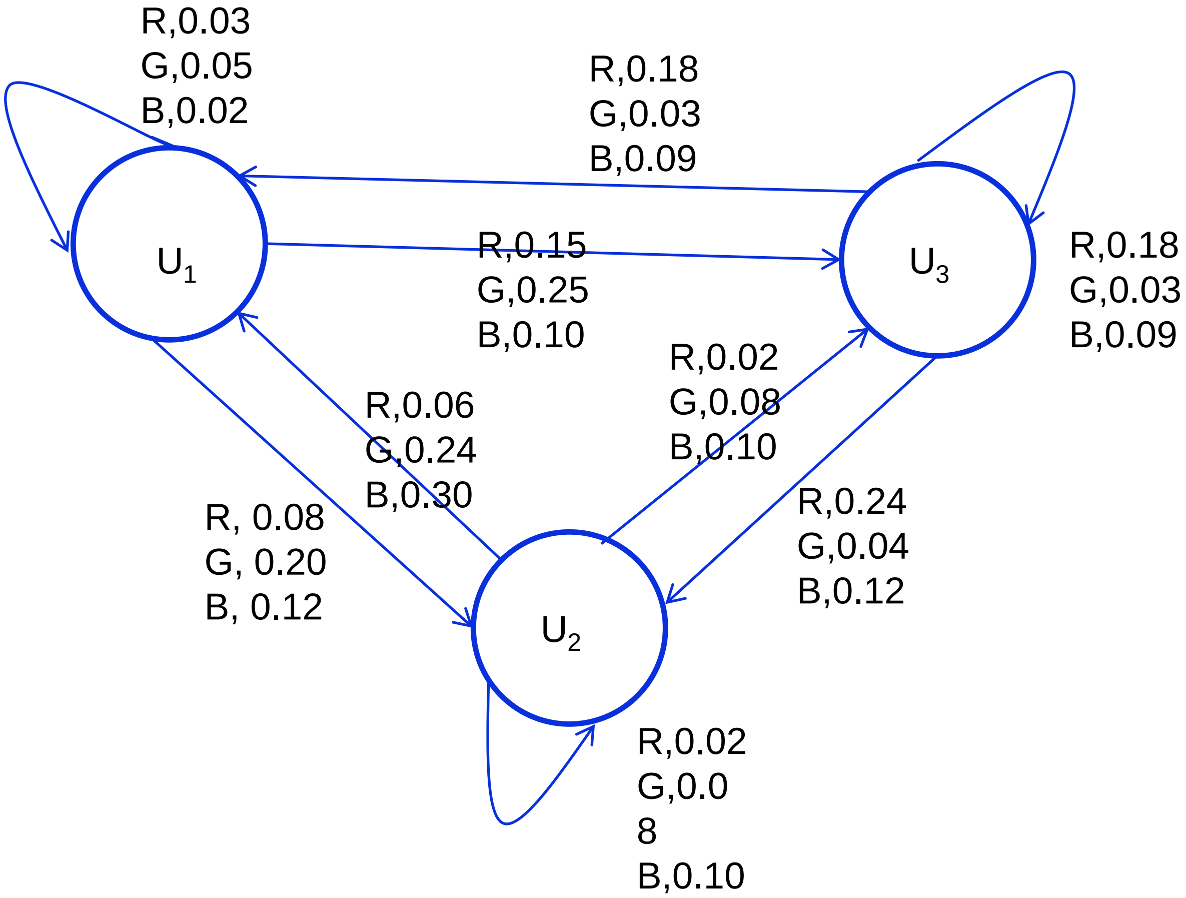
Not so Easily Computable.

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