D Y PATIL
DEEMED TO BE
UNIVERSITY
RAMRAO ADIK
INSTITUTE OF TECHNOLOGY
NAVI MUMBAI

Subject Name: Natural Language Processing

Unit No:3

Unit Name: Syntax Analysis

Faculty Name :Ms. Ekta Sarda

Index -

Lecture 13 –POS tagging ambiguities

Lecture 14 – Rule based POS tagging

Lecture 15 –Stochastic POS tagging

Unit No: 3 Syntax Analysis

Lecture No: 13
Part-Of-Speech(POS) Tag Ambiguities

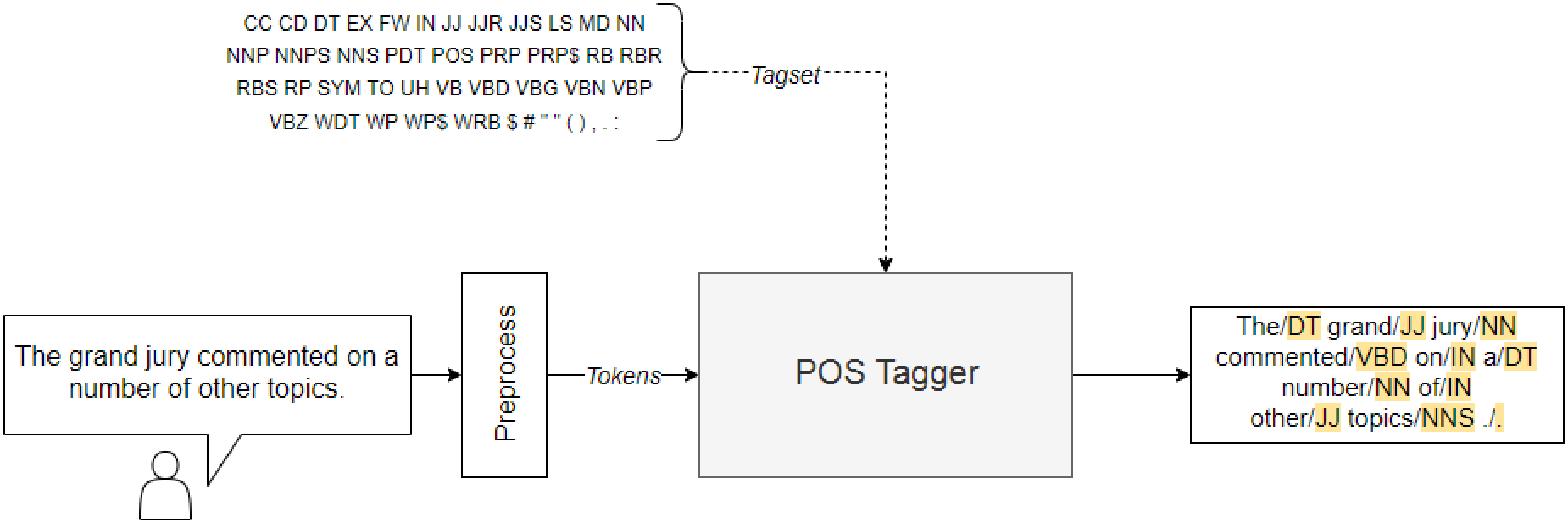
Recap- Syntax Analysis

- Syntactic analysis is defined as analysis that tells us the **logical meaning** of certain given sentences or parts of those sentences.
- We also need to consider **rules of grammar** in order to define the logical meaning as well as correctness of the sentences.
- Example: **School go a boy**, this sentence does not logically convey its meaning, and its grammatical structure is not correct. So, syntactic analysis tells us whether the given sentence conveys its logical meaning and whether its grammatical structure is correct.
- Syntactic analysis is a well-developed area of NLP **that deals with the syntax of NL**

Recap- Part Of Speech tagging (POS)

- The part of speech tagging is a **process of assigning corresponding part of speech** like noun, verb, adverb, adjective, verb to each word in a sentence.

- It is a process of converting a sentence to forms – list of words, list of tuples (where each tuple is having a form (word, tag)).



CC CD DT EX FW IN JJ JJR JJS LS MD NN NNP NNPS NNS PDT POS PRP PRP$ RB RBR RBS RP SYM TO UH VB VBD VBG VBN VBP VBZ WDT WP WP$ WRB $ # " " ( ) , . :

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

Role of POS Tagging in Syntax Analysis

- In natural language, to understand the meaning of any sentence we need to **understand the proper structure of the sentence and the relationship between the words** available in the given sentence.
- To overcome this issue, we need to learn POS Tagging and Chunking in NLP.

POS Tagging

Online Demo:

- https://textanalysisonline.com/nltk-pos-tagging

Why is POS Tagging Hard?

- The main problem with POS tagging is **ambiguity**.
- In English, many common words have multiple meanings and therefore multiple POS

- Example

1. Book/VB that/DT flight/NN
2. Does/VBZ that/DT flight/NN serve/VB dinner/NN

- Book can be NN or VB
- Can I read a book on this flight?

POS tag Ambiguity

- POS Tag Ambiguity is a challenge where a **single word has multiple POS tags**.
- The type of ambiguity where a single word can have multiple word categories is also called a **lexical ambiguity**.

- The Problem:
  - ○ Words often have more than one word class: this
  - ○ This is a nice day = PRP
  - ○ This day is nice = DT
  - ○ You can go this far = RB

POS tag Ambiguity

I **bank** on the **bank** on the river **bank** for my transactions .[English]

noun verb verb

Mujhe **khaanna** **khaanna** hai .[Hindi]

Noun verb

पुजा देवीची **पुजा** कर. [Marathi]

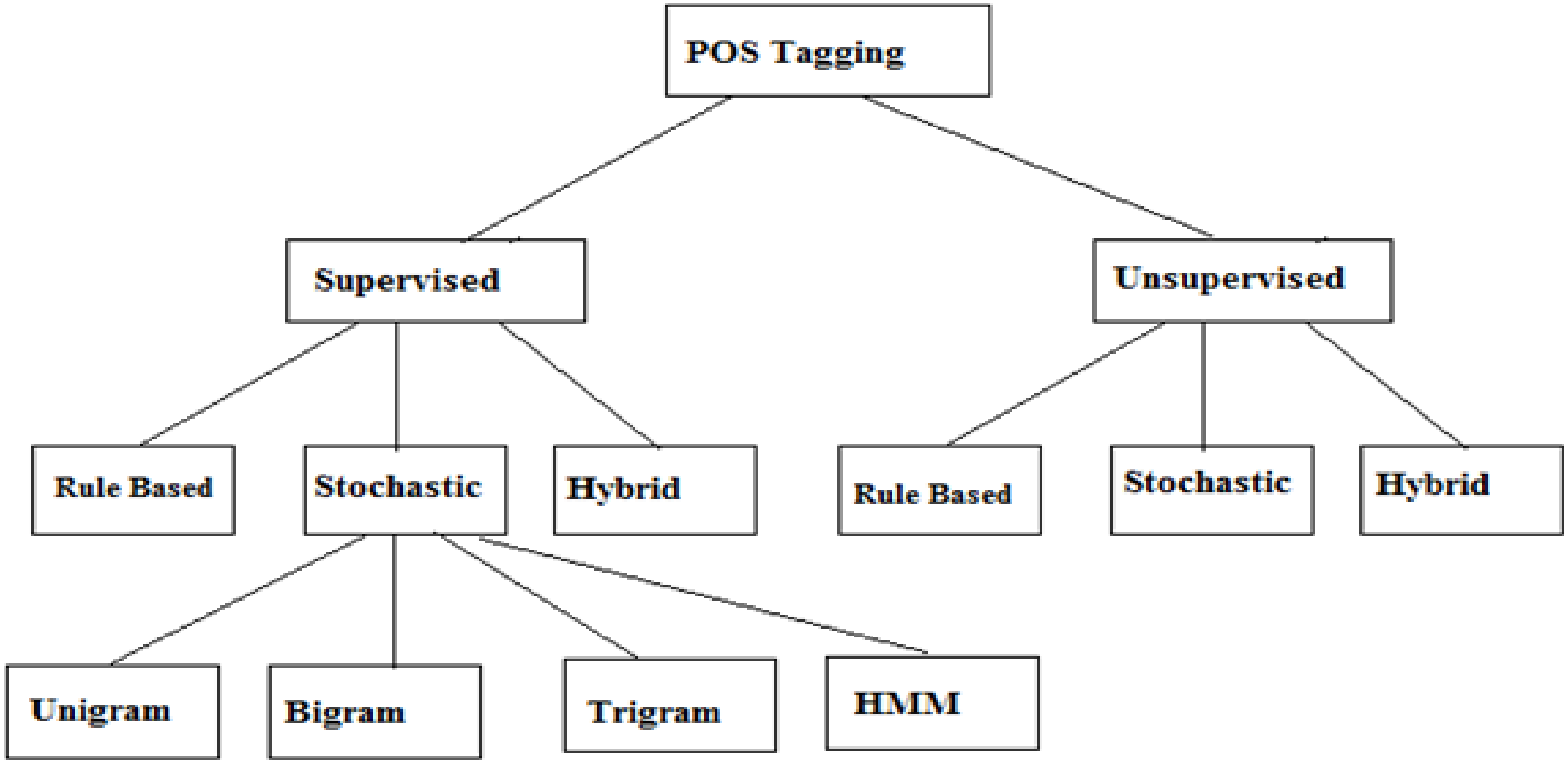
Noun verb

The aim here is identify the appropriate tag for an ambiguous word

Unit No: 3 Syntax Analysis

Lecture No: 14
Rule based POS tagging

Classification of POS tagging approaches



Classification of POS tagging approaches

Supervised POS Tagging:

- Supervised taggers are based on pre-tagged corpora which is used for training to learn information about the tagset, word-tag frequencies, rule sets.

Unsupervised POS Tagging:

- Instead, they use those methods through which automatically tags are assigned to words.

- Advanced computational methods like the Baum-Welch algorithm to automatically induce tag sets, transformation rules.

Classification of POS tagging approaches

Initialize and maintain tagging criteria

- – Supervised: uses pre-tagged corpora
- – Unsupervised: Automatically induce classes by probability and learning algorithms
- – Partially supervised: combines the above approaches

Algorithms

- – Rule based: Use pre-defined grammatical rules
- – Stochastic: use HMM and other probabilistic algorithms
- – Neural: Use neural nets to learn the probabilities

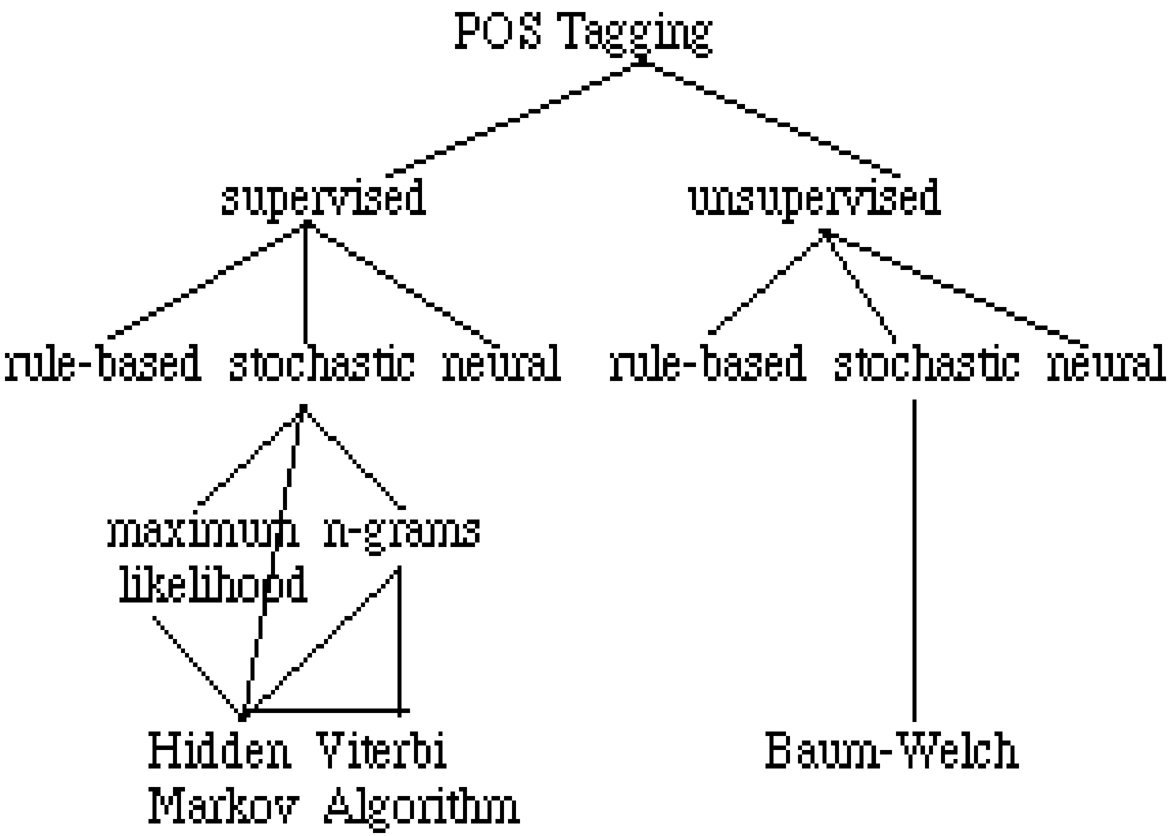
POS tagging

Rule-Based Tagger: ENGTWOL (ENGlish TWO Levelanalysis)

Stochastic Tagger: HMM-based

Transformation-Based Tagger (Brill)

POS tagging


POS Tagging
supervised
unsupervised
rule-based
stochastic
neural
rule-based
stochastic
neural
maximum likelihood
n-grams
Hidden Markov
Viterbi Algorithm
Baum-Welch

Rule based POS tagging

- One of the oldest techniques of tagging
- Rule-based taggers use dictionary or lexicon for getting possible tags for tagging each word.
- If the word has more than one possible tag, then rule-based taggers use hand-written rules to identify the correct tag.
- Disambiguation can also be performed in rule-based tagging by analyzing the linguistic features of a word along with its preceding as well as following words.
- For example, suppose if the preceding word of a word is article then word must be a noun.

Rule Based Approach

It use a set of hand written rules and depend on lexicon or dictionary.

Tagger divided into two stages.

a) Search: words are searched from the lexicon
b) Disambiguation: done by analyzing linguistic features of word .

The rules can be categorized in to two parts:
a) Lexical: Word level
b) Context Sensitive Rules: Sentence level

The transformation based approaches use a pre-defined set of handcrafted rules as well as automatically induced rules that are generated during training

Rule based POS tagging

- Rule-based POS taggers possess the following properties −
- These taggers are knowledge-driven taggers.
- The rules in Rule-based POS tagging are built manually.
- The information is coded in the form of rules.
- We have some limited number of rules approximately around 1000.
- Smoothing and language modeling is defined explicitly in rule-based taggers.

Rule based POS tagging

Basic Idea:

- Assign all possible tags to words
- Remove tags according to set of rules of type:
- 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.

Rule based POS tagging

- First Stage: In the first stage, it uses a dictionary to assign each word a list of potential parts-of-speech.

```
FOR each word
  Get all possible parts of speech using a morphological analysis algorithm
```

- Example

				NN		
				RB		
	VBN		JJ			VB
PRP	VBD	TO	VB		DT	NN
She	promised	to	back	the	bill	

Rule based POS tagging

In the second stage, it uses large lists of hand-written disambiguation rules to sort down the list to a single part-of-speech for each word.

Apply rules to remove possibilities

Example Rule:

IF VBD is an option and VBN|VBD follows “<start>PRP”
THEN Eliminate VBN

				NN			
				RB			
	VBN			JJ			VB
PRP	VBD		TO	VB		DT	NN
She	promised	to	back		the	bill	

ENGTWOL Rule-Based Tagger

A Two-stage architecture

Use lexicon FST (dictionary) to tag each word with all possible POS

Apply hand-written rules to eliminate tags.

The rules eliminate tags that are inconsistent with the context, and should reduce the list of POS tags to a single POS per word.

Some rules :

If an ambiguous word follows a determiner, tag it as a noun

Given input “that”

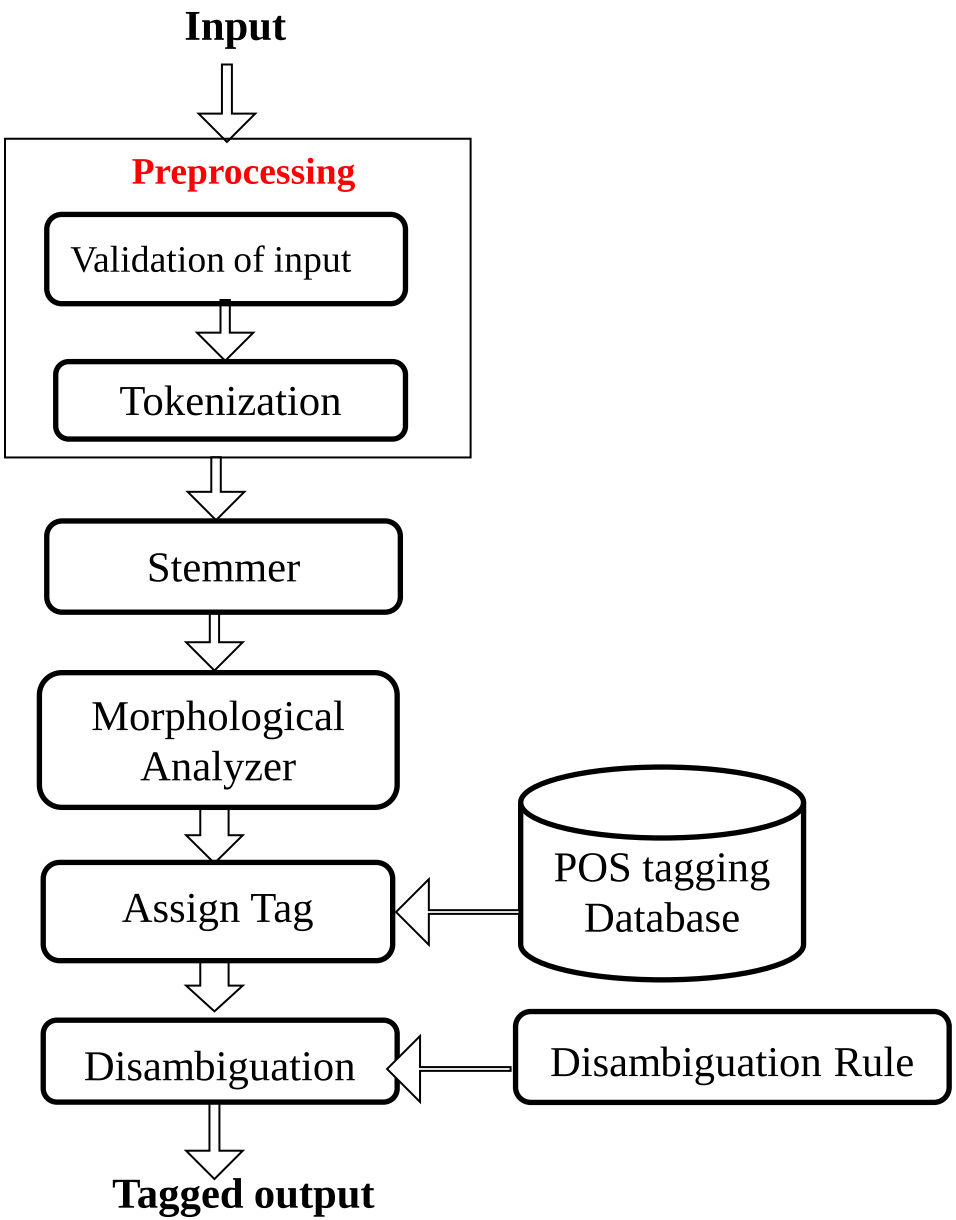
If the next word is adj, adverb, or quantifier, and following that is a sentence boundary, and the previous word is not a verb like “consider” which allows adjs as object complements,

Then eliminate non-ADV tags,

Else eliminate ADV tag

I consider that odd. (that is NOT ADV)

It isn’t that strange. (that is an ADV)



Rule based POS tagger System

24

Unit No: 3 Syntax Analysis

Lecture No: 15
Stochastic POS tagging

Stochastic Tagging

- The use of probabilities in tags is quite old. Stochastic taggers use **probabilistic and statistical information** to assign tags to words.
- The model that includes **frequency or probability (statistics**) can be called stochastic.
- These taggers might use ‘**tag sequence probabilities**’, ‘**word frequency measurements**’ or a combination of both.
- The tag encountered most frequently in the training set is the one assigned to an ambiguous instance of that word (word frequency measurements).

Stochastic Tagging

- The best tag for a given word is determined by the probability that it occurs with the n previous tags (tag sequence probabilities).

- Resolves the ambiguity by computing the probability of a given word (or the tag).

- The problem with this approach is that it can come up with sequences of tags for sentences that are not acceptable according to the grammar rules of a language.

Stochastic Tagging

Stochastic tagger applies the following approaches for POS tagging

1.Word Frequency Approach

- In this approach, the stochastic taggers disambiguate the words based on the probability that a word occurs with a particular tag.
- We can also say that the tag encountered most frequently with the word in the training set is the one assigned to an ambiguous instance of that word.
- The main issue with this approach is that it may yield inadmissible sequence of tags.

2**. Tag Sequence Probabilities**

- Here the tagger calculates the probability of a given sequence of tags occurring.
- It is also called n-gram approach. It is called so because the best tag for a given word is determined by the probability at which it occurs with the n previous tags.

Stochastic Tagging

- Unigram approach assigns each word to its most common tag and consider one word at a time. **P (ti/wi) = freq (wi/ti)/freq (wi)**
- Here Probability of tag given word is computed by frequency count of word given tag divided by frequency count of that particular word
- Bigram approach is based on preceding tag i.e. it take two tags: the preceding tag and current tag into account. **P (ti/wi) = P (wi/ti). P (ti/ti-1)**
- Here P (wi/ti) is the probability of current word given current tag and P (ti/ti-1) is the probability of a current tag given the previous tag
- Tigram is based on previous two tags. **P (ti/wi) = P (wi/ti). P (ti/ti-2, ti-1),** Where ti denotes tag sequence and wi denote word sequence.
- P (wi/ti) is the probability of current word given current tag.
- Here, P(ti|ti-2, ti-1)is the probability of a current tag given the previous two tags.

Properties of Stochastic POST Tagging

Stochastic POS taggers possess the following properties −

- This POS tagging is based on the probability of tag occurring.
- It requires training corpus.
- There would be no probability for the words that do not exist in the corpus.
- It uses different testing corpus (other than training corpus).
- It is the simplest POS tagging because it chooses most frequent tags associated with a word in training corpus.

Stochastic POST Tagging :HMM Tagger

Intuition: Pick the most likely tag based on context

Maximize the formula using a HMM : P(word|tag) × P(tag|previous n tags)

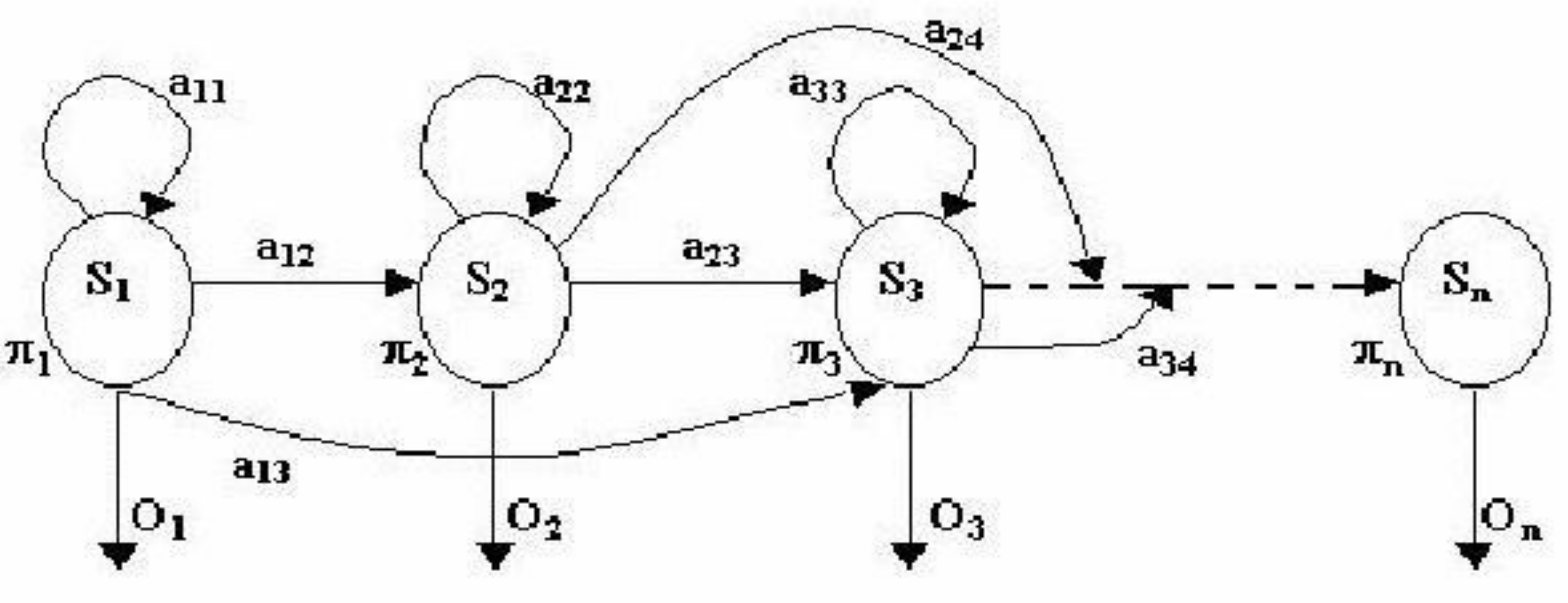
Observe: W = w1, w2, …, wn

Hidden: T = t1,t2,…,tn

Goal: Find POS tags that generate a sequence of words, i.e., look for most probable sequence of tags T underlying the observed words W

- We cannot determine the exact sequence of tags that generated and calculate using t = argmax P(w, t) and it is based on the Markovian assumption that the current tag depends only on the previous n tags.
- Use transition probability(i.e. forward tag and backward tags).
- P (ti/wi) = P (ti/ti-1). P (ti+1/ti). P (wi/ti)
- P (ti/ti-1) is the probability of current tag given previous tag

HMM



HMM Tagger

- P (ti+1/ti) is the probability of future tag given current tag.
- P (wi/ti) Probability of word given current tag
- Secretariat/NNP is/VBZ expected/VBN to/TO race/VB tomorrow/NN
- People/NNS continue/VBP to/TO inquire/VB the DT reason/NN for/IN the/DT race/NN for/IN outer/JJ space/NN
- to/TO race/???
- the/DT race/???
- ti = argmaxj P(tj|ti-1)P(wi|tj)
- max[P(VB|TO)P(race|VB) , P(NN|TO)P(race|NN)]
- Brown: P(NN|TO) = .021×P(race|NN) = .00041 = .000007
- P(VB|TO) = .34×P(race|VB) = .00003 = .00001

Part-of-speech (POS) tagging with Hidden Markov Model (HMM)

- Video Link on HMM Tagger Example:

https://www.youtube.com/watch?v=IGt8HPRARS0&t=1122s

Thank You