n-gram Model:

* To estimate the probability of a sentence, decompose sentence probability into a product of conditional probabilities using chain rule:

$$P(s) = P(w_1, w_2, w_3 \ldots w_n)$$
$$= P(w_1)\,P(w_2/w_1)\,P(w_3/w_1,w_2)\,P(w_4/w_1,w_2,w_3)\ldots$$
$$\ldots(P w_n/(w_1,w_2 \ldots w_{n-1})$$
$$= \boxed{\prod_{i=1}^{n} P(w_i/h_i)}$$

($h_i \rightarrow$ history of word w_i
↳ (or $w_1, w_2 \ldots w_{i-1}$)

* To get sentence probability, we need to calculate the word probability which is preceding it in a sequence

* The ~~time~~ history (h_i) to the previous one word only.
↳ bi-gram ($n=1$)

* previous 2 words → tri-gram ($n=2$)

*
$$\boxed{P(s) \approx \prod_{i=1}^{n} P(w_i/w_{i-1})} \rightarrow \text{Bi-gram}$$

$$\boxed{P(s) \approx \prod_{i=1}^{n} P(w_i/w_{i-2}, w_{i-1})} \quad \text{Tri-gram.}$$

* Ex: P(east / The Arabian Knights are fairy tales of the)
↳ P(east / the) — Bi-gram

↳ P(east / of the) — tri gram

* pseudoword(<s>) is introduced to mark the beginning of the sentence
→ Bi-gram <s1> trigram → <s1> & <s2>.

Advantage   N-Gram.   (1)

* Easy to understand, implement
* Can be easily convert to any gram

Disadv:

* Underflow due to multiplication of probabilities
    Sol: Use log. Add probabilities
* Zero probability problem.
    Sol: Use Laplace Smoothing

$$\prod_{i=1}^{n} \frac{\text{value}+1}{\text{value} + (\text{uniquewords})}$$

except <s>

Given Corpus:

<s> I am Henry </s>
<s> I like College </s>
<s> Do Henry like college </s>
<s> Henry I am </s>
<s> Do I like Henry </s>
<s> Do I like College </s>
<s> I do like Henry </s>.

P(<s> I like college </s>)

$= P(I \mid \langle s \rangle) \times P(\text{like} \mid I) \times P(\text{college} \mid \text{like}) \times P(\langle /s \rangle \mid \text{College})$

$= 3/7 \times 3/6 \times 3/5 \times 3/3 = \frac{9}{70} = 0.13.$

$= \log_e(3/7) + \log_e\left(\frac{3}{6}\right) + \log_e\left(\frac{3}{5}\right) + \log_e(3/3) = -2.0513$

Perplexity

It is the inverse probability of the test data which is normalized by the number of words.

$$PP(w) = P(w_1, w_2, w_3 \dots w_n)^{-1/N}$$

$$PP(w) = \left(\prod_{i=1}^{n} \frac{1}{P(w_i | w_{i-1})} \right)^{1/N}$$

Ex: <s> I like College </s>

[Perplexity for Bigram]

$= P(I | \langle s \rangle) \times P(like | I) \times P(college | like) \times P(\langle /s \rangle | college)$

$= 3/7 \times 3/6 \times 3/5 \times 3/3$

$= 9/70$

$= 0.13$

$PP(w) = (1/0.13)^{1/4} = \underline{1.67}$

Perplexity for Trigram

$P(w) = P(like | \langle s \rangle) \times P(college | I\ like) \times P(\langle /s \rangle | like\ college)$

$P(w) = 1/3 \times 2/3 \times 3/3$

$= 2/9$

$= 0.22$

$PP(w) = \left(\frac{1}{0.22}\right)^{1/3}$

$= \boxed{1.66}$ ← less probability so ~~more~~ best Predicting model.

[Which language model will have less perplexity is the best model for your data]

Parts of Speech Tagging (HMM)

POS Tagging

Noun: john, car, India..

Verb: run, swim ...

Modal verbs: must, will, would, can, may.

N N N
Mary Jane will

Jane saw will
N V N

Collect data

Create lookuptable

Tag our target sentence.

done.

Lookuptable

Ex1: N V N
Jane saw will

N V N
Mary saw Jane.

	N	V
Mary	1	–
saw	–	2
Jane	2	–
will	1	–

N N N
Mary Jane will

Ex2:

N M V N
Mary will see Jane.

N M V N
will will see Mary

N M V N
Jane will see will

N
Mary will see will.
N? M? V? ??

lookup table

.	N	V	M
Mary	2	0	0
See	0	3	0
Jane	2	0	0
will	2	0	3

* So should consider "context" into consideration (2) using Bigram.

	N-M	M-V	V-N
Mary-will	1	-	-
will-see	-	3	-
See-jane	-	-	1
will-will	1	-	-
~~will-see~~	~~0~~	~~0~~	~~1~~
See-mary	0	0	1
Jane-will	1	0	0
See-will	0	0	1

Mary will see will
? ? ? ?
Noun M. V N

Ex:3 Mary Jane will Spot
N N N N

Mary Jane Can See will

Spot will See Mary

will Jane spot Mary?

Mary will ~~spot~~ pat Spot.

Lookup table

	N-M	M-V	V-N
Mary-Jane			
Jane-can			
Can-see			
⋮			

~~Quest~~ I/P stm is $\boxed{\text{Jane will spot will.}}$

If the occurance is not there in data? → Soln. HMM

2 types of Probabilities

1. Transition Probability → How likely ~~to~~ tags probability after another tag
2. Emission Probability → How likely ~~to~~ tag will allocate for a word.

Mary/N Jane/N Can/M See/V will/N

Spot/N will/M see/V Mary/M

Will/M Jane/N Spot/V Mary/N

Mary/N will/M pat/V Spot/N

lookup table.

	N	M	V
Mary	4/9	0	0
Jane	2/9	0	0
Will	1/9	3/4	0
Spot	2/9	0	1/4
Can	0	1/4	0
See	0	0	2/4 1/2
Pat	0	0	1/4

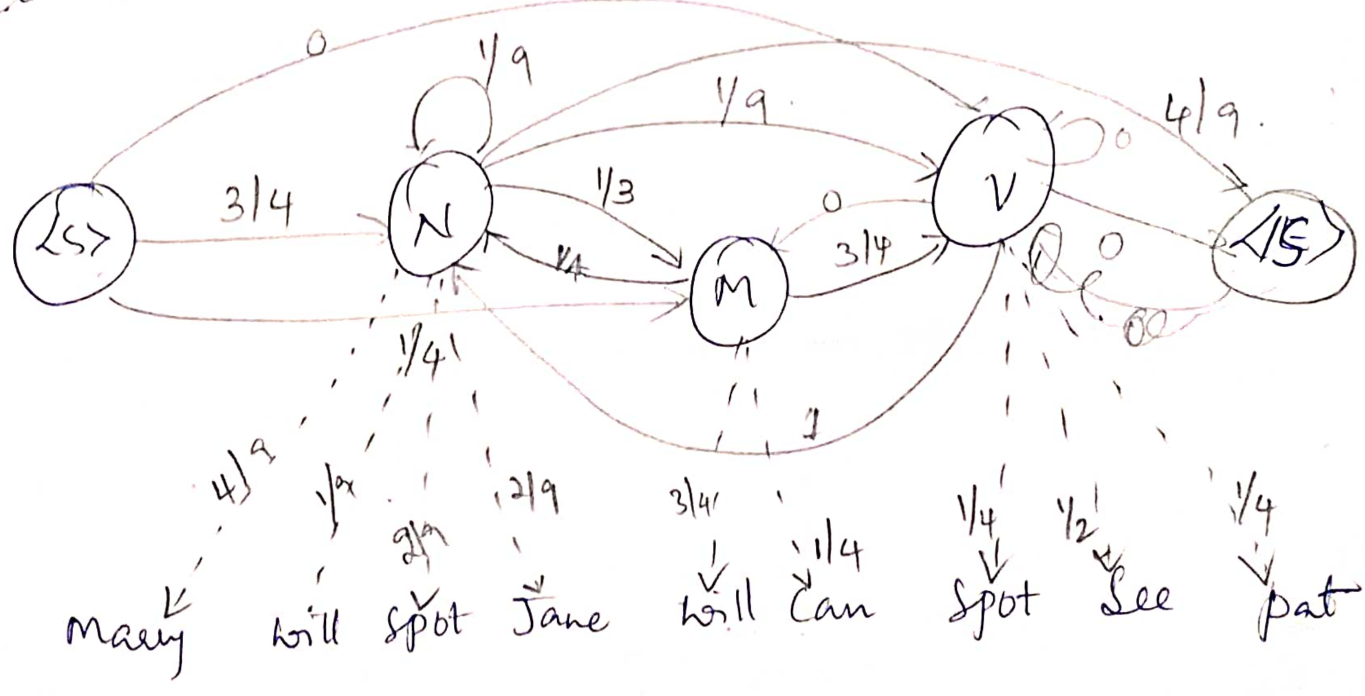
Emission Probability

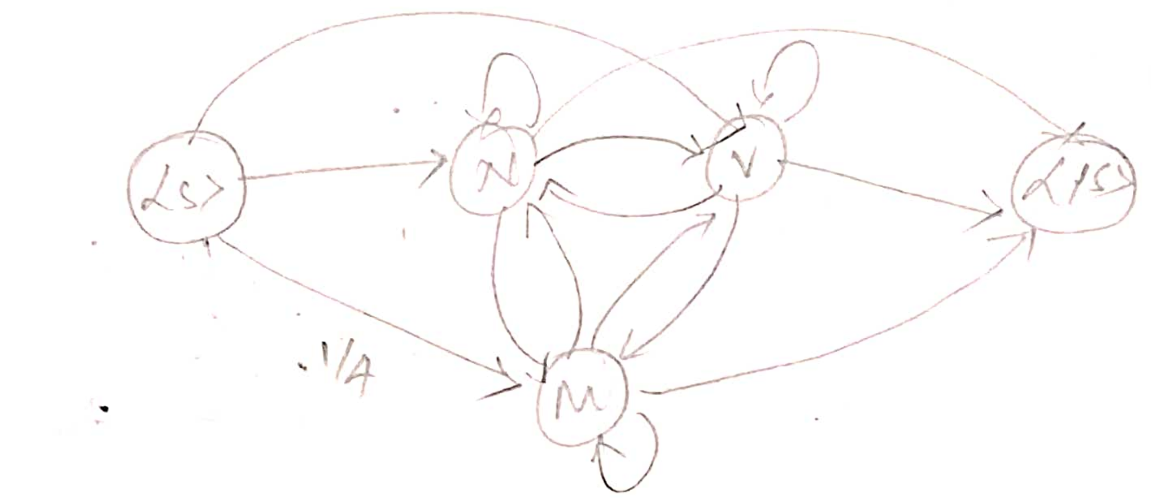
Transition Probability

add <S> & </S> to the stmts.

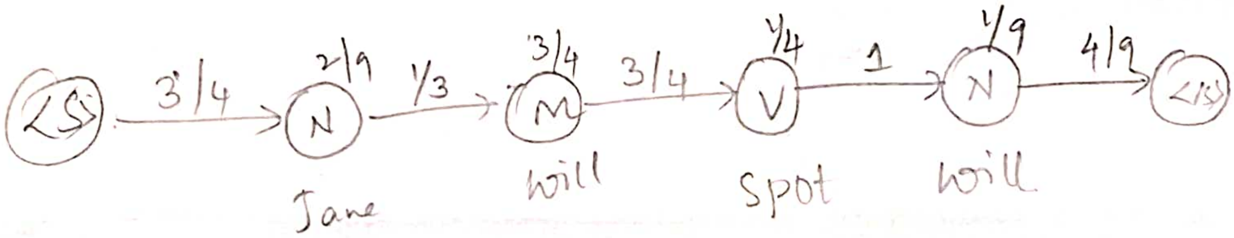
	N	M	V	</S>
<S>	3/4	1/4	–	–
N	1/9	1/3 3/9	1/9	4/9
M	1/4	0	3/4	–
V	4/4	–	–	–

Combine 2 Probabilities to create HMM.

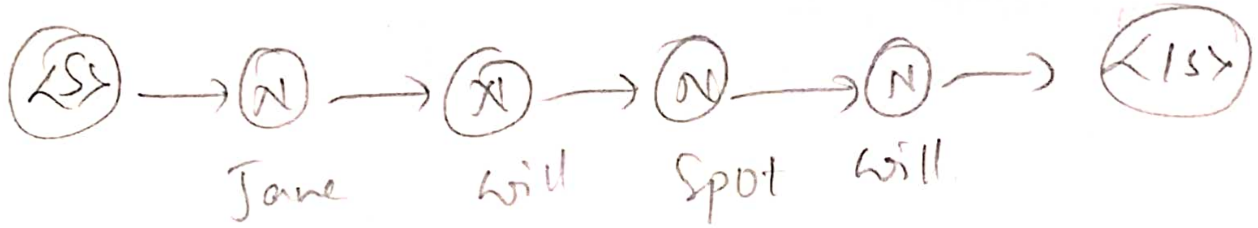




Jane will Spot Will



= $\boxed{0.003858}$ Multiply all the values.



= 0.0000002788

How many Combinations to Compare:

3^4 → 81 (Hidden states) ^ No. of words in the stmt

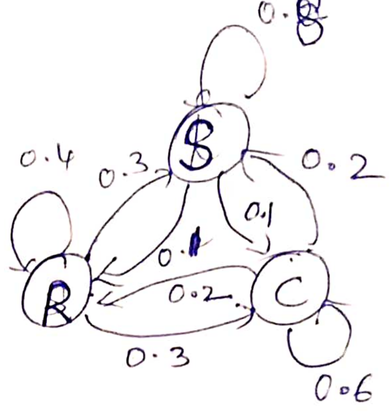
* Computationally expensive.
* As data grows the no. of observations & comparing Probabilities also grow.

Weather Observation

S1 : Rainy   S2: Cloudy   S3: Sunny.

The state transition probabilities are

$$
A = \begin{array}{c|ccc} & R & C & S \\ \hline R & 0.4 & 0.3 & 0.3 \\ C & 0.2 & 0.6 & 0.2 \\ S & 0.1 & 0.1 & 0.8 \end{array} \rightarrow \text{sum}=1
$$



Question:

① Given that the weather on day 1 is sunny, what is the probability that the weather for the next 7 day will be "Sun-Sun-rain-rain-Sun-Cloudy-sun?

$$O = \{S_3, S_3, S_3, S_1, S_1, S_3, S_2, S_3\}$$

$$P(O|\text{model}) = P(S_3, S_3, S_3, S_1, S_1, S_3, S_2, S_3 | \text{Model})$$

$$= P(S_3)P(S_3|S_3)P(S_3|S_3)\,P(S_1|S_3)\,P(S_1|S_1)$$

$$P(S_3|S_1)\,P(S_2|S_3)\,P(S_3|S_2)$$

$$= 1 * 0.8 * 0.8 * 0.1 * 0.4 * 0.3 * 0.1 * 0.2.$$

$$= 1.536 \times 10^{-4}.$$

② ~~Proba~~ Today is Sunny, what is the probability that tomorrow is sunny & next day is Rainy?

$$P(\text{Sunny}|\text{sunny}) * P(\text{Rainy}|\text{Sunny})$$

$$= 0.8 * 0.1$$

$$= 0.08$$