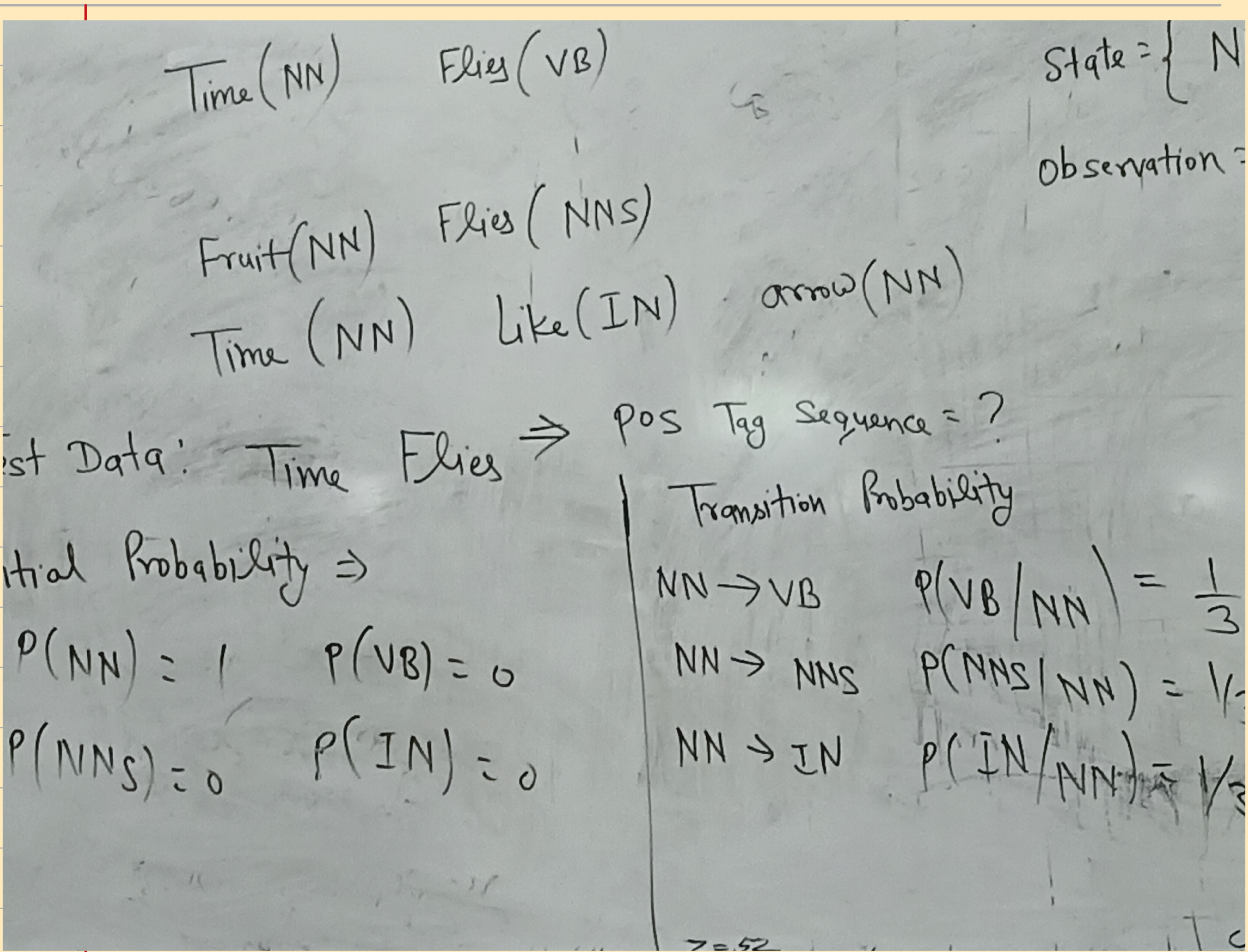
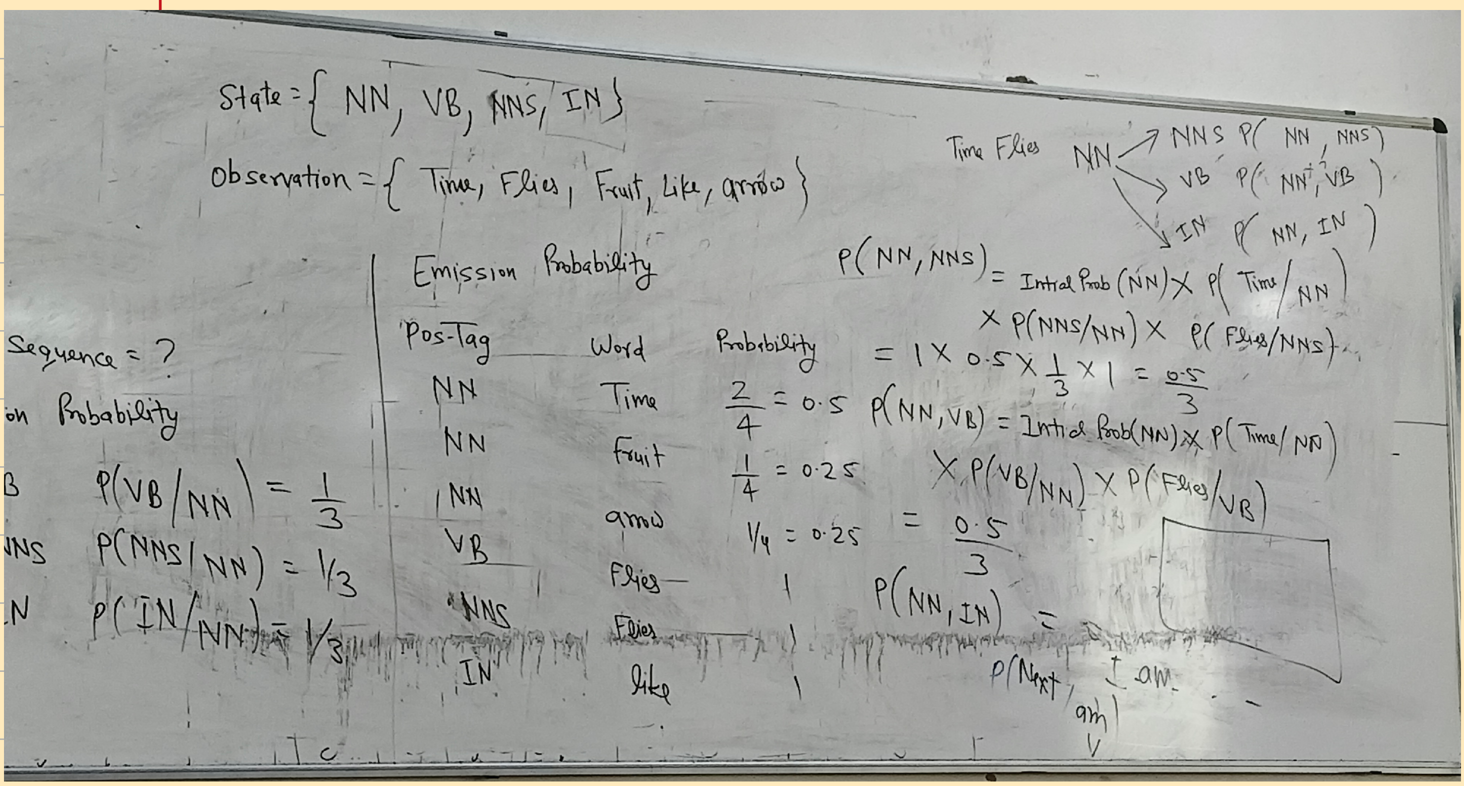
30/01/26


Time(NN) Flies(VB)
State = { N
observation =
Fruit(NN) Flies(NNS)
Time (NN) Like(IN) arrow(NN)
st Data: Time Flies ⇒ Pos Tag Sequence = ?
Transition Probability
itial Probability ⇒
P(NN) = 1 P(VB) = 0
P(NNS) = 0 P(IN) = 0
NN → VB P(VB/NN) = 1/3
NN → NNS P(NNS/NN) = 1/
NN → IN P(IN/NN) = 1/


State = { NN, VB, NNS, IN }
Observation = { Time, Flies, Fruit, Like, arrow }
Time Flies
NN → NNS P(NN , NNS)
NN → VB P(NN , VB)
NN → IN P(NN, IN)
Emission Probability
Pos-Tag Word Probability
NN Time 2/4 = 0.5
NN Fruit 1/4 = 0.25
NN arrow 1/4 = 0.25
VB Flies 1
NNS Flies 1
IN like 1
P(NN, NNS) = Initial Prob (NN) X P(Time/NN) X P(NNS/NN) X P(Flies/NNS)
= 1 X 0.5 X 1/3 X 1 = 0.5/3
P(NN, VB) = Initial Prob(NN) X P(Time/NN) X P(VB/NN) X P(Flies/VB)
= 0.5/3
P(NN, IN) =
P(Next / I am)
Sequence = ?
on Probability
B P(VB/NN) = 1/3
NNS P(NNS/NN) = 1/3
N P(IN/NN) = 1/3

State = { NN, VB, NNS, IN }

Observation = { Time, Flies, Fruit, Like, arrow }

Time Flies

NN → NNS $P(NN, NNS)$

NN → VB $P(NN, VB)$

NN → IN $P(NN, IN)$

Emission Probability

Pos-Tag	Word	Probability
NN	Time	$\frac{2}{4} = 0.5$
NN	Fruit	$\frac{1}{4} = 0.25$
NN	arrow	$1/4 = 0.25$
VB	Flies	1
NNS	Flies	1
IN	like	1

$P(VB/NN) = \frac{1}{3}$

$P(NNS/NN) = 1/3$

$P(IN/NN) = 1/3$

$P(NN, NNS) = \text{Initial Prob}(NN) \times P(Time/NN) \times P(NNS/NN) \times P(Flies/NNS)$

$= 1 \times 0.5 \times \frac{1}{3} \times 1 = \frac{0.5}{3}$

$P(NN, VB) = \text{Initial Prob}(NN) \times P(Time/NN) \times P(VB/NN) \times P(Flies/VB)$

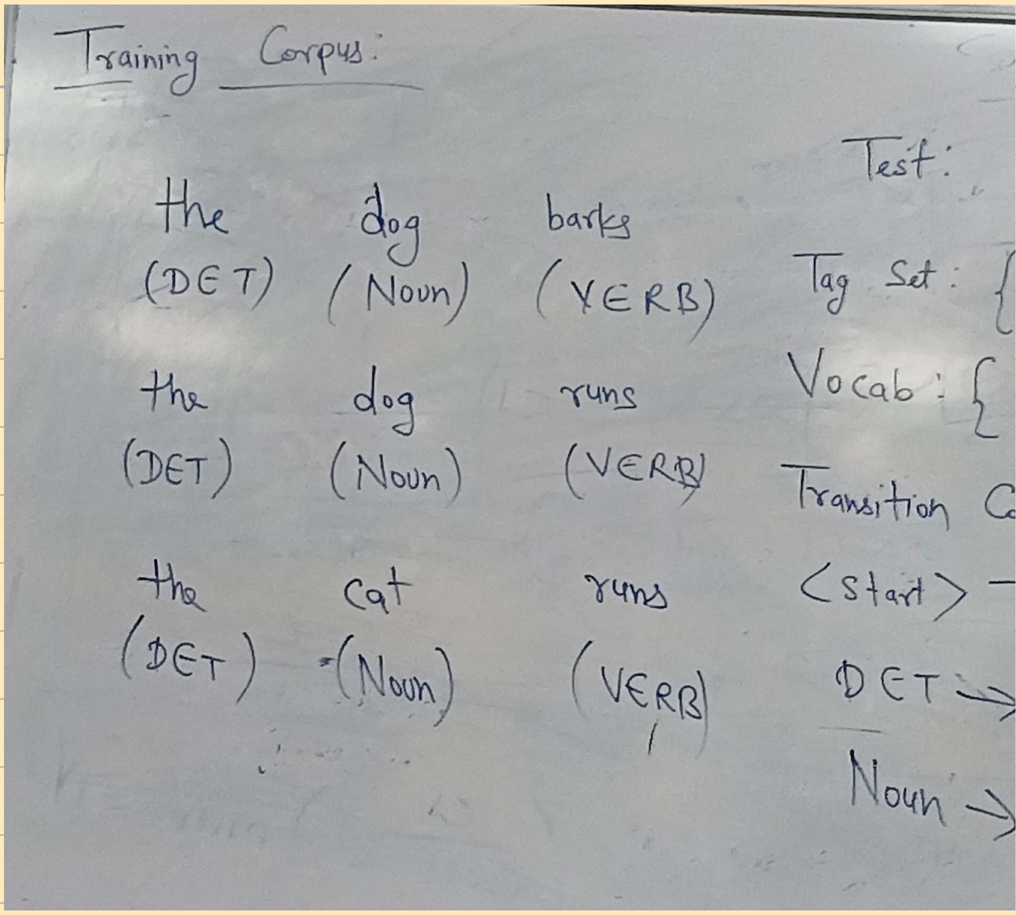
$= \frac{0.5}{3}$

$P(NN, IN) =$

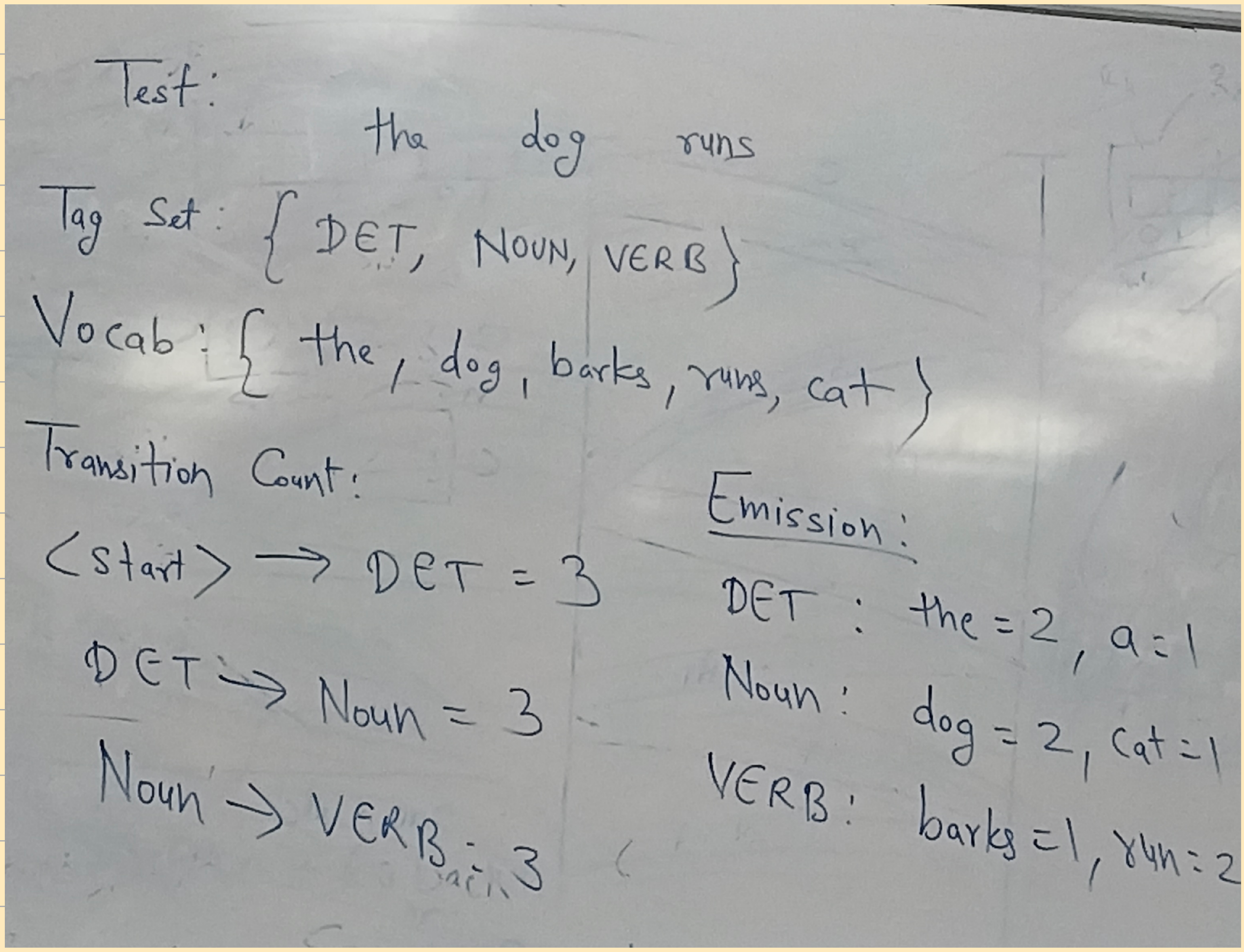
P(Next / am)

I am . . .

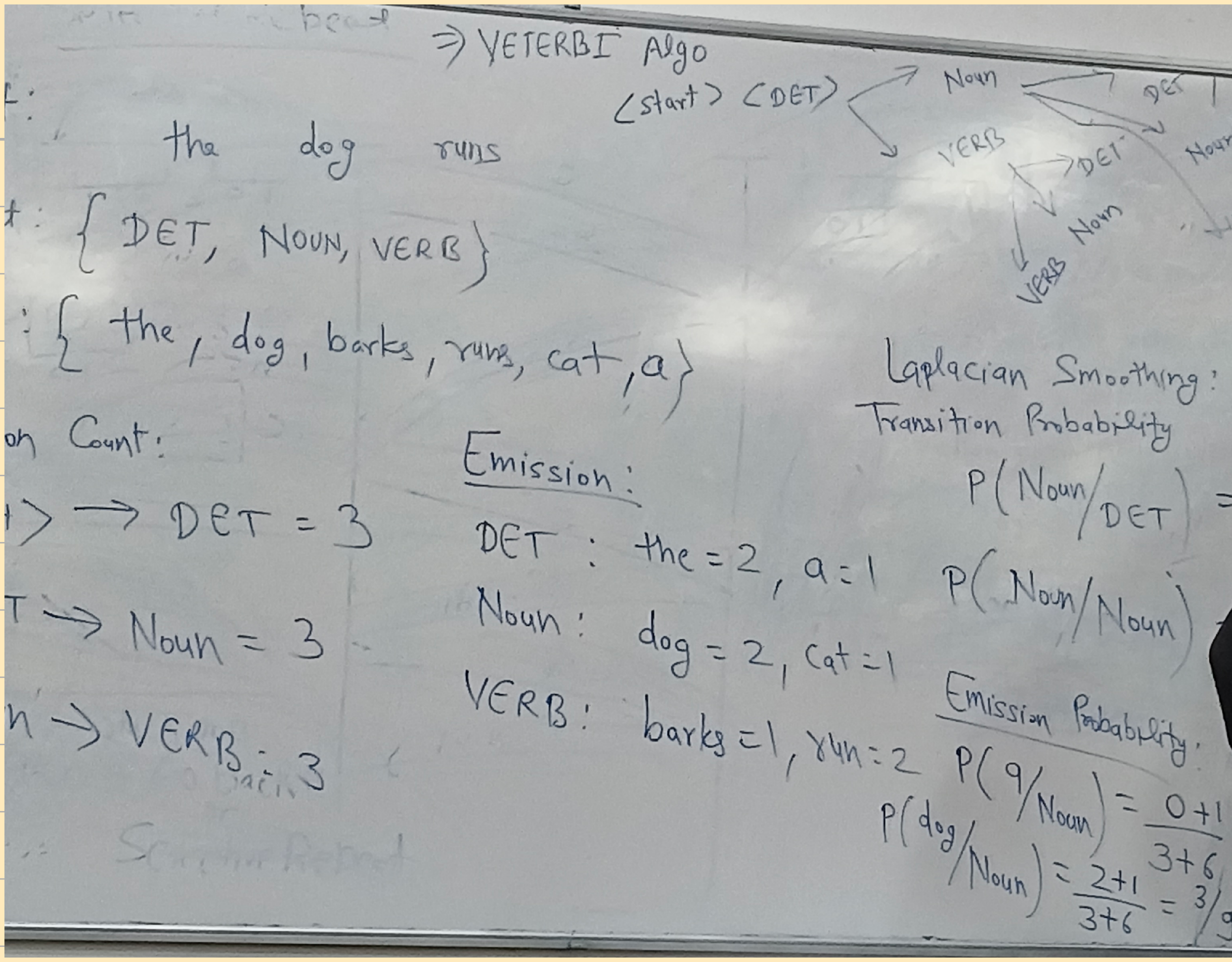
03/02/26

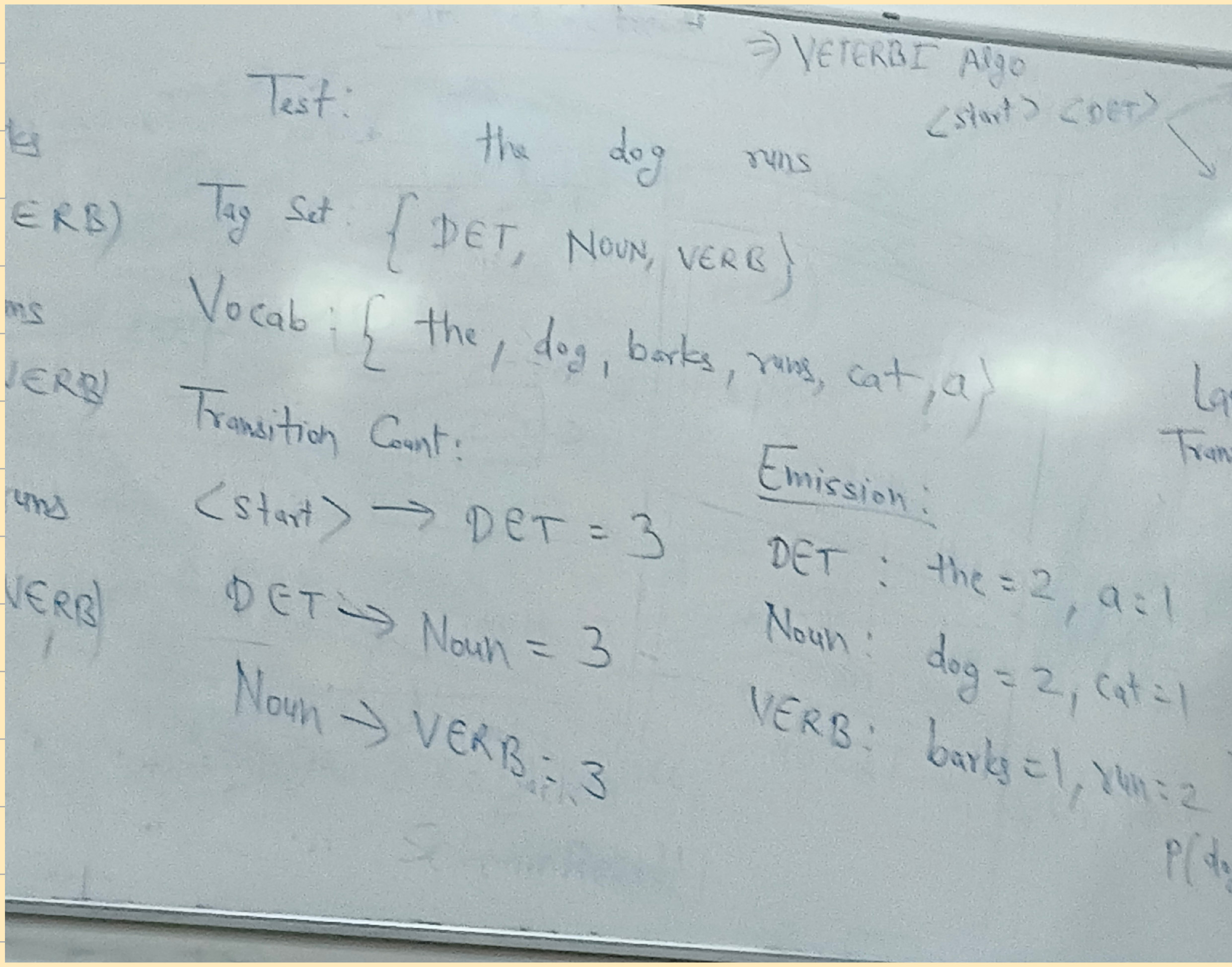


Training Corpus:
the dog barks
(DET) (Noun) (VERB)
the dog runs
(DET) (Noun) (VERB)
the cat runs
(DET) (Noun) (VERB)
Test:
Tag Set: {
Vocab: {
Transition C
<start> →
DET →
Noun →

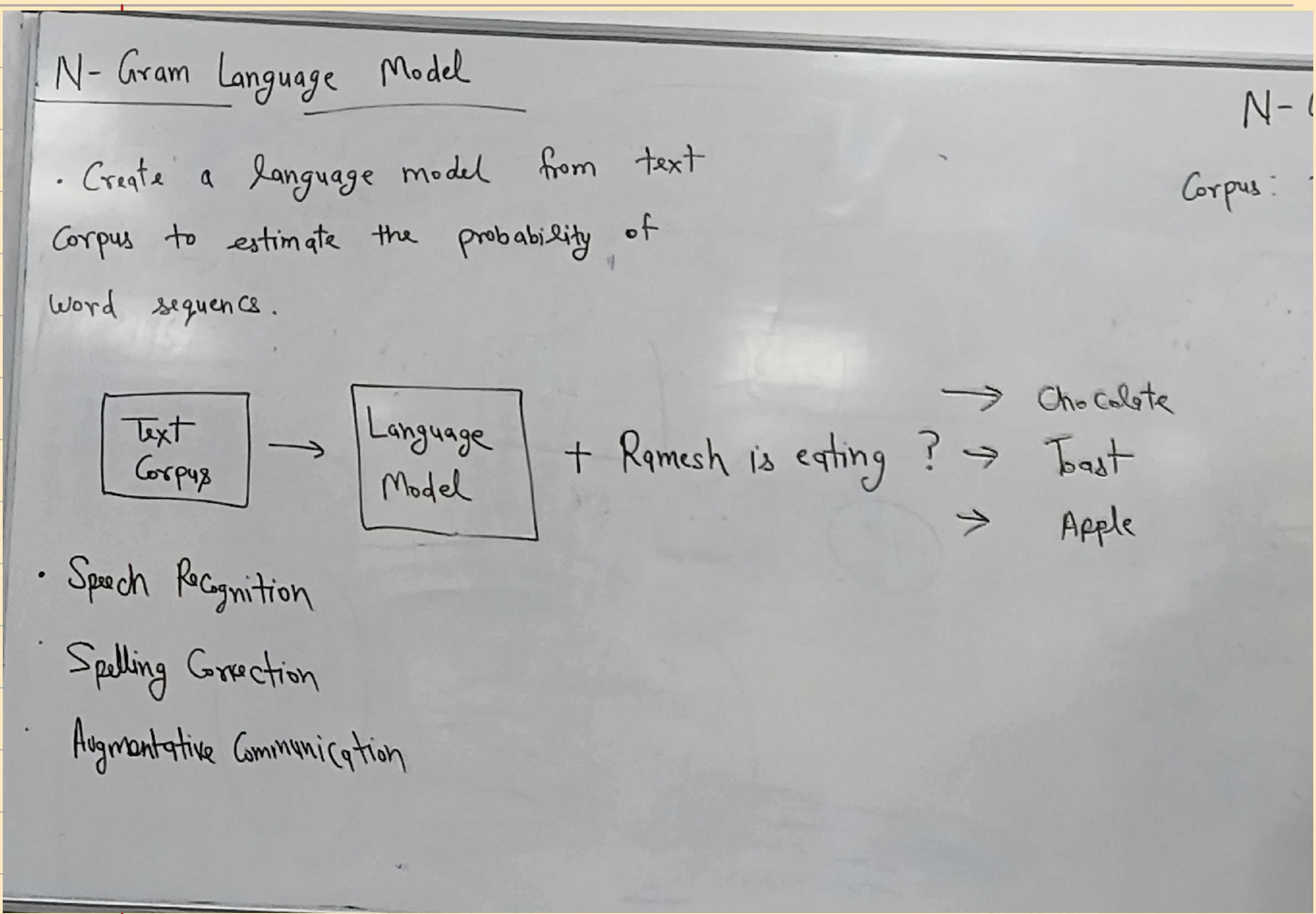


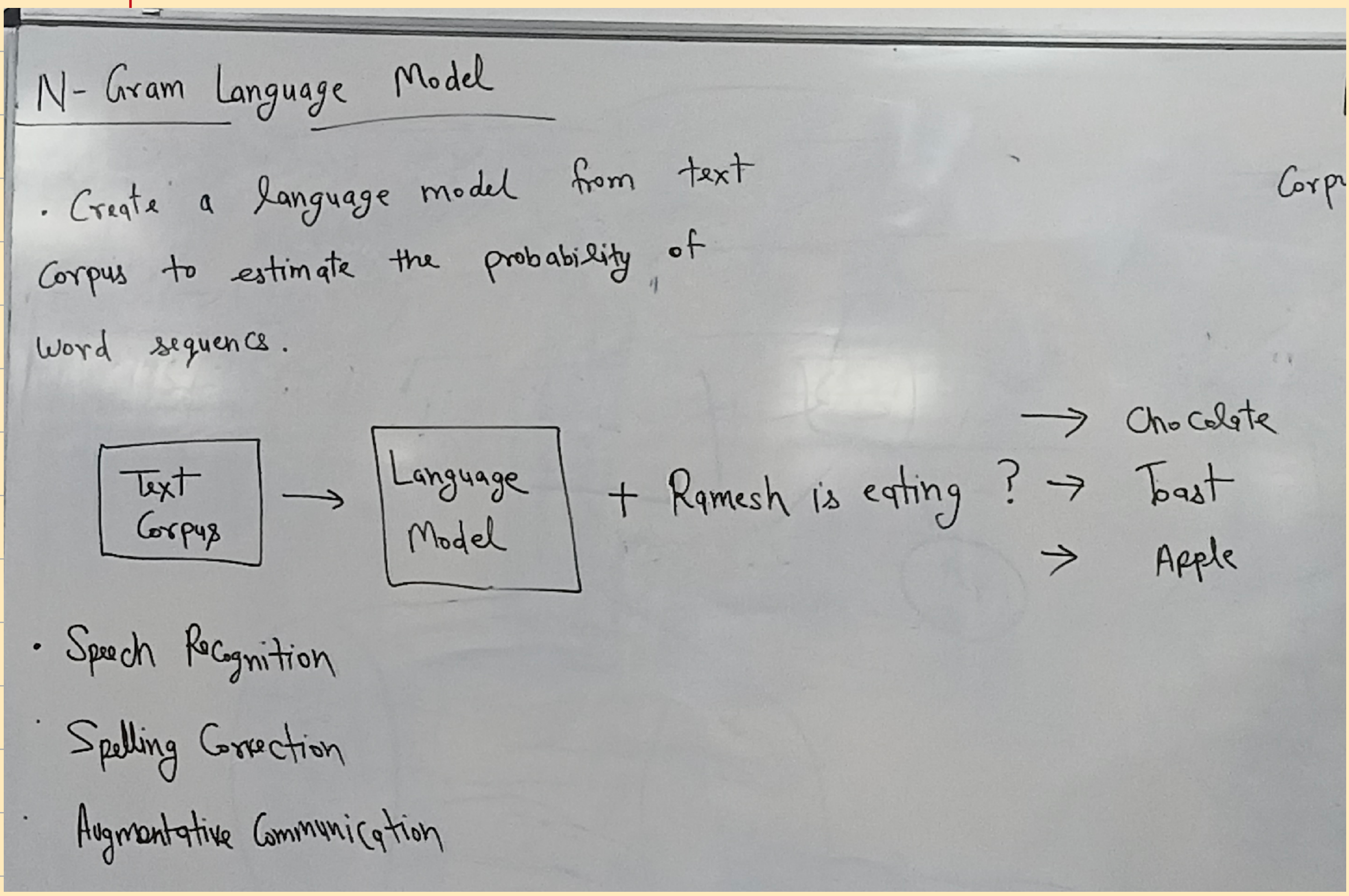
Test: the dog runs
Tag Set: { DET, NOUN, VERB }
Vocab: { the, dog, barks, runs, cat }
Transition Count:
<start> → DET = 3
DET → Noun = 3
Noun → VERB = 3
Emission:
DET : the = 2, a = 1
Noun : dog = 2, cat = 1
VERB : barks = 1, run = 2


⇒ VETERBI Algo
<start> <DET>
Noun
DET
Noun
VERB
DET
Noun
VERB
the dog runs
{DET, NOUN, VERB}
{the, dog, barks, runs, cat, a}
Count:
> → DET = 3
T → Noun = 3
n → VERB : 3
Emission:
DET : the = 2, a : 1
Noun : dog = 2, Cat = 1
VERB : barks = 1, run = 2
Laplacian Smoothing:
Transition Probability
P(Noun/DET) =
P(Noun/Noun)
Emission Probability.
P(a/Noun) = (0+1)/(3+6)
P(dog/Noun) = (2+1)/(3+6) = 3/9


⇒ VETERBI Algo
<start> <DET>
Test:
the dog runs
Tag Set {DET, NOUN, VERB}
Vocab {the, dog, barks, runs, cat, a}
Transition Count:
<start> → DET = 3
DET → Noun = 3
Noun → VERB : 3
Emission:
DET : the = 2, a : 1
Noun : dog = 2, Cat = 1
VERB : barks = 1, run = 2

05/02/26


N-Gram Language Model
· Create a language model from text
Corpus to estimate the probability of
word sequence.
Text Corpus
Language Model
+ Ramesh is eating ?
Chocolate
Toast
Apple
· Speech Recognition
· Spelling Correction
· Augmentative Communication
N-
Corpus :


N-Gram Language Model
· Create a language model from text
Corpus to estimate the probability of
word sequence.
Text Corpus
Language Model
+ Ramesh is eating ?
Chocolate
Toast
Apple
· Speech Recognition
· Spelling Correction
· Augmentative Communication

N-Gram : is a sequence of n words

Corpus : I am happy because I am learning NLP.

Unigram: Sequence of one word.

Ex: { I, am, happy, because, learning, NLP }

Bigram : Sequence of two words.

Ex: { I am, am happy, happy because, ... }

Trigrams: Sequence of three words

Ex: { I am happy, am happy because ... }

N-gram ⇒ { Sequence of n words, ... }

Sequence Notation :

NLP.

$$w_1^2 = \{w_1 w_2\}$$

$$w_1^3 = \{w_1 w_2 w_3\}$$

LP }

$$w_1^N = \{w_1 w_2 w_3 \ldots w_N\}$$

Unigram Probability

$$P(I) = \frac{2}{8} \qquad P(\text{happy}) = \frac{1}{8}$$

...}

Bigram Probability:

$$\{I, am\} \Rightarrow P(am / I) = \frac{\text{Count}(I\ am)}{\text{Count}(I)} = 2/2 = 1$$

$$P(x, y) = \frac{\text{Count}(x, y)}{\text{Count}(x)}$$