Laplacian (Add-One) Smoothing
Used to avoid zero probability for unseen words or transitions.

Emission:
$P(w|t) = (count(w,t) + 1) / (count(t) + |V|)$

Transition:
$P(tj|ti) = (count(ti,tj) + 1) / (count(ti) + |T|)$

4. Numerical Example

Training Corpus:
1. I/PRP like/VB dogs/NNS
2. You/PRP like/VB cats/NNS

Vocabulary Size $|V| = 5$
Tag Set Size $|T| = 3$

Initial Probabilities:
$P(PRP) = (2+1)/(2+3) = 3/5$
$P(VB) = 1/5$
$P(NNS) = 1/5$

Transition Probabilities:
$P(VB|PRP) = (2+1)/(2+3) = 3/5$
$P(NNS|VB) = (2+1)/(2+3) = 3/5$

Emission Probabilities:
$P(I|PRP) = (1+1)/(2+5) = 2/7$
$P(like|VB) = (2+1)/(2+5) = 3/7$
$P(dogs|NNS) = (1+1)/(2+5) = 2/7$

Sentence Probability:
$P(I\ like\ dogs) \approx 0.013$

Example 2: Another example with Laplacian smoothing

Problem Statement
Consider a small corpus with the following POS tags:
N = Noun, V = Verb

Vocabulary: dog, barks

Training data:
1. dog/N barks/V
2. dog/N barks/V
Step 1: Count Frequencies
Tag counts:
N = 2, V = 2

Transition counts:
N → V = 2
Start → N = 2

Emission counts:
P(dog|N) = 2
P(barks|V) = 2
Step 2: Apply Laplacian Smoothing
Number of tags = 2
Vocabulary size = 2

Transition Probability:
P(V|N) = (2 + 1) / (2 + 2) = 3/4
P(N|N) = (0 + 1) / (2 + 2) = 1/4

Emission Probability:
P(dog|N) = (2 + 1) / (2 + 2) = 3/4
P(barks|N) = (0 + 1) / (2 + 2) = 1/4
P(barks|V) = (2 + 1) / (2 + 2) = 3/4
P(dog|V) = (0 + 1) / (2 + 2) = 1/4
Step 3: POS Tagging (Viterbi Calculation)
Sentence: dog barks

Step 1 (dog):
P(N) × P(dog|N) = 1 × 3/4 = 0.75
P(V) × P(dog|V) = 1 × 1/4 = 0.25

Step 2 (barks):

From N → V: 0.75 × 3/4 × 3/4 = 0.4219
From N → N: 0.75 × 1/4 × 1/4 = 0.0469

Best path: N → V
Final Answer
The most likely POS tagging sequence is:
dog/N barks/V