Q.1] Backoff vs Interpolation

(A) Backoff

- Backoff N-gram modelling is a non-linear method
- We build on N gram model based on (N-1) gram model
- The difference is that in backoff if we have non-zero trigram counts we solely rely on trigram counts & don't interpolate the bigram and unigram counts at all.
- Backoff model in trigram format:

$$P(w_i \mid w_{i-2} w_{i-1}) = \begin{cases} \tilde{P}(w_i \mid w_{i-2} w_{i-1}) & \text{if } c(w_{i-2} w_{i-1} w_i) > 0 \\ \alpha(w_{n-2}^{n-1}) \tilde{P}(w_i \mid w_{i-1}) & \text{if } c(w_{i-2} w_{i-1} w_i) = 0 \ \& \ c(w_{i-1} w_i) > 0 \\ \alpha(w_{n-1}) \cdot \tilde{P}(w_i) & \text{otherwise} \end{cases}$$

- Doesn't yield valid probability distribution
- Works well for large datasets.

(B) Interpolation

- Combines different N-grams by linearly interpolating all 3 models whenever we are computing any trigram.
- Here, we don't train 3 λ's as trigram grammar. Instead we make each λ a function of the context.
- λ terms are used to decide how much to smooth
- $\sum_i \lambda_i = 1$
- Mathematically, $\tilde{P}(w_0 \mid w_{-2} w_{-1}) = \lambda_3 P(w_0 \mid w_{-2} w_{-1}) + \lambda_2 \cdot P(w_0 \mid w_{-1}) + \lambda_1 \cdot P(w_0)$
- Can interpolate 'customised' model with general model

Q.2] Viterbi algorithm

- It is a variation of the forward algorithm which considers all words simultaneously in order to compute the most likely path.

Algorithm:

Input: observations of length T, state-graph of length N

Output: best path

For each state s from 1 to N do

$q[1,s] \leftarrow P(s \mid S_0) \cdot P(O_1 \mid s)$

backpointers $[1,s] \leftarrow 0$

For each time step t from 2 to T do

For each state s from 1 to N do

$q[t,s] \leftarrow \max_{s'=1}^{N} q[t-1, s'] \cdot p(s \mid s') \cdot P(O_t \mid s)$

backpointers $[t,s] \leftarrow \operatorname{argmax}_{s'=1}^{N} q[t-1, s'] \cdot P(s \mid s')$

$s \leftarrow \operatorname{argmax}_{s'=1}^{N} q[T, s']$

return the backtrace path from the backpointers $[T, s]$

<u>Example</u>: Consider a 2 word language: 'fish' & 'sleep'.

Suppose in our training corpus,

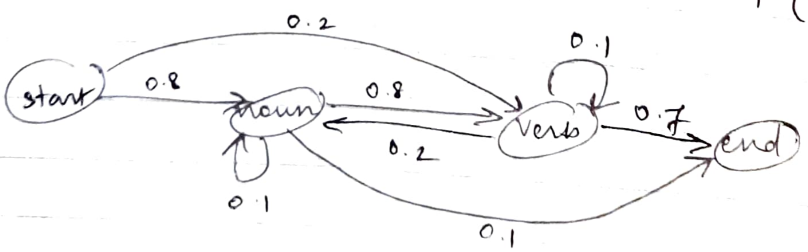
'fish' appears 8 times as a noun & 5 times as a verb

'sleep' " 2 " " " " & 5 " " " "

⇒ Emission probabilities

- Noun
  - $P(\text{fish} \mid \text{noun}) = 0.8$
  - $P(\text{sleep} \mid \text{noun}) = 0.2$

- Verb)
  - $P(\text{fish} \mid \text{verb}) = 0.5$
  - $P(\text{~~fish~~ sleep} \mid \text{verb}) = 0.5$



Token 1 : fish

	0	1	2	3
start	1	0		
verb	0	0.2×0.5		
noun	0	0.8×0.8		
end	0	0		

Token 2 : sleep

	0	1	2	3
start	1	0	0	
verb	0	0.1	0.64×0.8×0.5 ← max ✓ 0.1×0.1×0.5	
noun	0	0.64	0.64×0.1×0.2 ← max ✓ 0.1×0.2×0.2	
end	0	0	-	

Token 3 : end

	0	1	2	3
start	1	0	0	0
verb	0	0.1	0.256	-
noun	0	0.64	0.0128	-
end	0	0		0.256×0.7 ← max ✓ 0.0128×0.1

∴ now we can backtrack the most likely path

Q.3] Corpus

<s> I am from DJ </s>

<s> I am a teacher </s>

<s> All students are good and intelligent </s>

<s> Students from DJ score high marks </s>

Test data

<s> Students are from DJ </s>

- <u>Unigram</u>

<s>	students	are	from	DJ	</s>
4	2	1	2	2	4

Bigram occurence count

	<s>	students	are	from	DJ	</s>
<s>	0	1	0	0	0	0
students	0	0	1	1	0	0
are	0	0	0	0	0	0
from	0	0	0	0	2	0
DJ	0	0	0	0	0	1
</s>	0	0	0	0	0	0

	<s>	students	are	from	DJ	</s>
<s>	0	1/4	0	0	0	0
students	0	0	1/2	1/2	0	0
are	0	0	0	0	0	0
from	0	0	0	0	2/2=1	0
DJ	0	0	0	0	0	1/2
</s>	0	0	0	0	0	0

Using MLE to estimate probability of test data

P = P(students | <s>) · P(are | students) · P(from | are) · P(DJ | from) · P(</s> | DJ)

$= \frac{1}{4} \times \frac{1}{2} \times 0 \times 1 \times \frac{1}{2}$ hence we need to apply laplace smoothing

To apply laplace smoothing

V = count of unique vocabulary

= count({<s>, </s>, I, am, from, DJ, a, teacher, all, students, are, good, and, intelligent, score, high, marks})

= 17

$\therefore P = \left(\frac{1+1}{4+17}\right) \cdot \left(\frac{1+1}{2+17}\right) \cdot \left(\frac{0+1}{1+17}\right) \cdot \left(\frac{2+1}{2+17}\right) \cdot \left(\frac{1+1}{2+17}\right) = 9.257 \times 10^{-6}$

Q4] Corpus

<s> I am sam </s>

<s> Sam I am </s>

<s> I do not like green eggs & ham </s>

(a) Bigram probability (i) $P(am \mid sam)$ (ii) $P(do \mid I)$ (iii) $P(am \mid I)$

- $P(w_n \mid w_{n-1}) = \dfrac{C(w_{n-1}\, w_n)}{C(w_{n-1})}$

(i) $P(am \mid sam) = \dfrac{P(sam\ am)}{P(sam)} = \dfrac{0}{2} = 0$

(ii) $P(do \mid I) = \dfrac{P(I\ do)}{P(I)} = \dfrac{1}{3}$

(iii) $P(am \mid I) = \dfrac{P(I\ am)}{P(I)} = \dfrac{2}{3}$

(b) Trigram probability 'I am Sam'

- $P(w_n \mid w_{n-2}\, w_{n-1}) = \dfrac{C(w_{n-2}\, w_{n-1}\, w_n)}{C(w_{n-2}\, w_{n-1})}$

$P(Sam \mid I\ am) = \dfrac{C(I\ am\ Sam)}{C(I\ am)} = \dfrac{1}{2}$

(c) MLE for 'I am Sam' using bigram

- $(\langle s\rangle, I), (I, am), (am, Sam), (Sam, \langle/s\rangle)$

$MLE = P(I \mid \langle s\rangle) \cdot P(am \mid I) \cdot P(Sam \mid am) \cdot P(\langle/s\rangle \mid Sam)$

$= \dfrac{2}{3} \times \dfrac{2}{3} \times \dfrac{1}{2} \times \dfrac{1}{2} = \dfrac{1}{9}$

Q.5] Corpus

<s> John read Moby Dick </s>

<s> Mary read a different book </s>

<s> She read a book by Cher </s>

ⓐ MLE for 'John read a book'.

((<s>, John), (John, read), (read, a), (a, book), (book, <

MLE = P(John | <s>) . P(read | John) . P(a | read) . P(book |

P(</s> | book)

$$= \frac{1}{3} \times \frac{1}{1} \times \frac{2}{3} \times \frac{1}{2} \times \frac{1}{2} = \frac{1}{18} = 0.056$$

ⓑ MLE for 'Cher read a book'

((<s>, cher), (cher, read), (read, a), (a, book), (book, </s>)

MLE = P(Cher | <s>) . P(read | Cher) . P(a | read) . P(book |

P(</s> | book)

$$= \frac{0}{3} \times \frac{0}{1} \times \frac{2}{3} \times \frac{1}{2} \times \frac{1}{2}$$

Using add-one smoothing (Laplace) $\left(\frac{\text{Count} + 1}{\text{Count} + V} \right)$

Total unique tokens = 11

$$= \frac{0+1}{3+11} \cdot \frac{0+1}{1+11} \cdot \frac{2+1}{3+11} \cdot \frac{1+1}{2+11} \cdot \frac{1+1}{2+11}$$

$$= \frac{1}{14} \cdot \frac{1}{12} \cdot \frac{3}{14} \cdot \frac{2}{13} \cdot \frac{2}{13} = 3.019 \times 10^{-5}$$