Some terms

Machine translation - predicting which word is used is more frequent
Improvement in perplexity often correlates with improvement in speech recognition performance

Edit distance

Do row by row if letter is different, add substitution cost
min edit distance at any cell is the cost + 1 from the left cell
if letter is same take no cost from i-1, j-1 (diagonal)

Tag	Description	Example	Tag	Description	Example
CC	coordin. conjunction	*and, but, or*	SYM	symbol	+,%, &
CD	cardinal number	*one, two, three*	TO	"to"	*to*
DT	determiner	*a, the*	UH	interjection	*ah, oops*
EX	existential 'there'	*there*	VB	verb, base form	*eat*
FW	foreign word	*mea culpa*	VBD	verb, past tense	*ate*
IN	preposition/sub-conj	*of, in, by*	VBG	verb, gerund	*eating*
JJ	adjective	*yellow*	VBN	verb, past participle	*eaten*
JJR	adj., comparative	*bigger*	VBP	verb, non-3sg pres	*eat*
JJS	adj., superlative	*wildest*	VBZ	verb, 3sg pres	*eats*
LS	list item marker	*1, 2, One*	WDT	wh-determiner	*which, that*
MD	modal	*can, should*	WP	wh-pronoun	*what, who*
NN	noun, sing. or mass	*llama*	WP$	possessive wh-	*whose*
NNS	noun, plural	*llamas*	WRB	wh-adverb	*how, where*
NNP	proper noun, singular	*IBM*	$	dollar sign	$
NNPS	proper noun, plural	*Carolinas*	#	pound sign	#
PDT	predeterminer	*all, both*	"	left quote	' or "
POS	possessive ending	*'s*	"	right quote	' or "
PRP	personal pronoun	*I, you, he*	(	left parenthesis	[, (, {, <
PRP$	possessive pronoun	*your, one's*	)	right parenthesis	],), }, >
RB	adverb	*quickly, never*	,	comma	,
RBR	adverb, comparative	*faster*	.	sentence-final punc	. ! ?
RBS	adverb, superlative	*fastest*	:	mid-sentence punc	: ; ... – -
RP	particle	*up, off*			

	word	POS tag		word	POS tag
(a)	**Mary**	NNP	(f)	**Mr.**	NNP
(b)	**also**	RB	(g)	**resolved**	PP
(c)	**bought**	VBD	(h)	**must**	MD
(d)	**10**	CD	(i)	**not**	RB
(e)	**apples**	NNS	(j)	**Every**	DT

0 Personal pronoun

PRP - He

Determiners (DT)

the, a, some
E.g. Does that

particles (RP) can appear after object
prepositions (IN) can appear only after verb
E.g. of, off, on
I thought that/IN

adjectives (JJ)

E.g. other, grand, already married/JJ

Verbs

VBD - commented
VBP - do, have
VBZ - is
There/EX are 70 children there/RB

Modal

MD - could

Personal Pronouns

PRP - you
PRP$ - your

Binary classification

sign(x) = 1 for x $\geq$ 0, -1 for x < 0 Feature extraction so that there is a good mapping of x

Softmax

[2 -1 5]
$\frac{e^2}{e^2+e^{-1}+e^5}$ into another vector of 3 numbers where it sum to 1

Loss functions

$L_{cross-entropy}(\hat{y}, y) = -\sum_i ylog(\hat{y})$

or $-log(\hat{y}) for hard classification$
$\hat{y}$ is 1 then it can minimize the loss function
$\hat{y}$ requires softmax for transformation

Ranking loss

$L_{ranking}(x, x') = max(0, 1 - (f(x) - f(x')))$

Gradient descent

$w_i \leftarrow w_i - \alpha \frac{\partial_L}{\partial_{w_i}}$ for $\alpha > 0$
Successive iterative approximation, can't plug in a value like $w_{1,1}$
if L(w) is convex (single min point), then it will converge to global minimum

Expected number of bits

$\sum bits * P(bits)$ =1 $\times \frac{1}{2} + 2 \times \frac{1}{4} + 3 \times \frac{1}{8}$ ... = 2 Entropy is the number of bits needed to encode

Per word cross entropy

Entropy rate = $\frac{1}{n}H(w_1) = -\frac{1}{n}(\sum p(W) log(p(W))$

Smoothing

$P(w|c_i) = \frac{\#times\ w\ occur\ in\ texts\ of\ class\ c_i}{\sum \#timesw\ occurs\ in\ texts\ of\ class\ c_i}$
$P(w|c_i) = \frac{\#times\ w\ occur\ in\ texts\ of\ class\ c_i\ +1}{\sum \#times\ w\ occurs\ in\ texts\ of\ class c_i + V}$
$C^*(w_0 w) = \{C(w_0 w) + 1\} \times \frac{C(w_0)}{C(w_0)+V}$

Witten Bell

If C*(w$_x w_i$) > 0, $C(w_x) \times \frac{C(w_x w_i)}{C(w_x)+T(w_x)}$
If C*(w$_x w_i$) = 0, $\frac{c(w_x)T(w_x)}{Z(w_x)(c(w_x)+T(w_x))}$

Stochastic POS tagging

$P(T, W) = P(< s >, t_1, w_1, t_2, w_2 ... < s >)$
$= P(< s >) \cdot P(t_1| < s >) \cdot P(w_1| < s >, t_1)$
$P(T|W) = \frac{P(T,W)}{P(W)} = P(T, W)$

Markov assumption

w_k only depends on the previous n - 1 words
$P(w_k|w_1, .., w_{k-1})$ $\approx$P(w$_k|w_{k-1}$)

Vertibi

v(tag, word) = P(w$_i|t_i$) $\times P(t_i|t_{i-1}) \times P(t_{i-1})$

$Trigram P(t_i|t_{i-1}t_{i-2}) = P(t_i|t_{i-1}t_{i-2}) + P(t_i|t_{i-1}) + P(t_i)$

Forward computation

1. Look at the number of input nodes
2. Compute the s node which is the value before there is actually h_1(non-linear activation function)

$s_i = w_x i_i + w_{x+1} i_{i+1} ... + w_k i_k + ... b_i$
For the hidden layers i_i will be h_i

$h_i = \frac{1}{1+e^{-s_i}}$
Final value h_i will be o_i

$$L = \frac{1}{2}[(o_1 - t_1)^2 + (o_2 - t_2)^2]$$

Backward computation

Base case

1. Take the s values previously computed
2. Calculate $\frac{\partial_L}{\partial_{w_m}} = \frac{\partial_L}{\partial_{s_1}} \frac{\partial_{s_1}}{\partial_{w_m}}$

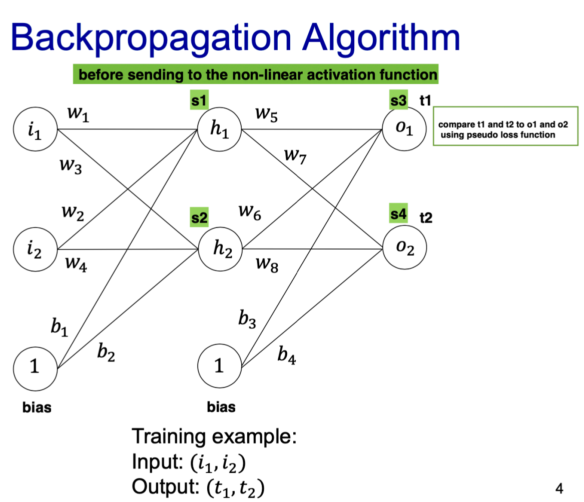
E.g. $\frac{\partial_L}{\partial_{w_6}}$

$$\frac{\partial_L}{\partial_{s_3}} = (o_1 - t_1) \times o_1(1 - o_1)$$
$$\frac{\partial_{s_3}}{\partial_{w_6}} = h_1$$

Recursive case

E.g. $\frac{\partial_L}{\partial_{w_2}}$

$$\frac{\partial_L}{\partial_{w_2}} = \frac{\partial_L}{\partial_{s_1}} \times i_2$$
$$\frac{\partial_L}{\partial_{s_1}} = [\frac{\partial_L}{\partial_{s_3}} \times w_5 + \frac{\partial_L}{\partial_{s_4}} \times w_7] \times h_1(1 - h_1)$$



Answer to Question 4:

$v(V, \text{water}) = v_1(1) = P(\text{water} \mid V) \times P(V \mid <s>) \times P(<s>) = \frac{1}{20} \times \frac{3}{4} \times 1 = \frac{3}{80}$

$v(N, \text{water}) = v_1(2) = P(\text{water} \mid N) \times P(N \mid <s>) \times P(<s>) = \frac{1}{50} \times \frac{1}{4} \times 1 = \frac{1}{200}$

Since

$$\begin{aligned} v(V, \text{plants}) &= v_2(1) \\ &= P(\text{plants} \mid V) \times \max\{P(V \mid V) \times v1(1), P(V \mid N) \times v1(2))\} \\ &= \frac{1}{50} \times \max\{\frac{2}{5} \times \frac{3}{80}, \frac{1}{6} \times \frac{1}{200}\} \\ &= \frac{3}{10000} \end{aligned}$$

Therefore, edge from V_1 and V_2 is chosen.

Since

$$\begin{aligned} v(N, \text{plants}) &= v_2(2) \\ &= P(\text{plants} \mid N) \times \max\{P(N \mid V) \times v1(1), P(N \mid N) \times v1(2))\} \\ &= \frac{1}{10} \times \max\{\frac{1}{5} \times \frac{3}{80}, \frac{2}{3} \times \frac{1}{200}\} \\ &= \frac{3}{4000} \end{aligned}$$

Therefore, edge from V_1 and N_2 is chosen.

Since

$$\begin{aligned} \max\{P(</s> \mid V) \times v_2(1), P(</s> \mid N) \times v_2(2)\} &= \max\{\frac{2}{5} \times \frac{3}{10000} \times 1, \frac{1}{6} \times \frac{3}{4000} \times 1\} \\ &= \frac{1}{8000} \end{aligned}$$

Therefore, edge between N_2 and </s> is chosen.

Hence, path: <s> → V_1 → N_2 → </s> is the chosen path.

Therefore, the optimal sequence of part-of-speech tags is <s>**, V, N,** </s>