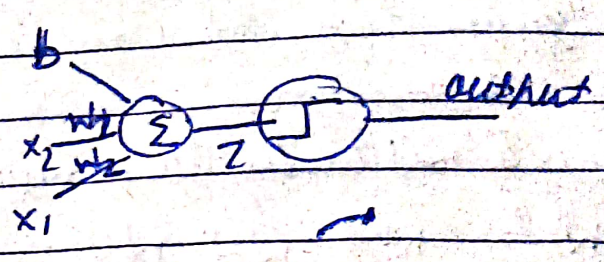
① XOR

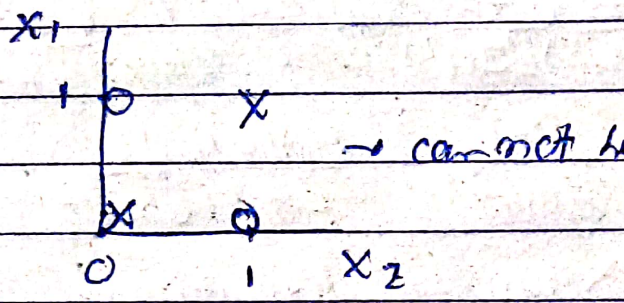
x_1	x_2	$\hat{y}$
0	1	1
1	0	1
1	1	0
0	0	0



$z = w_1x_1 + w_2x_2 + b$

A perceptron can only classify data if it is linearly separable.

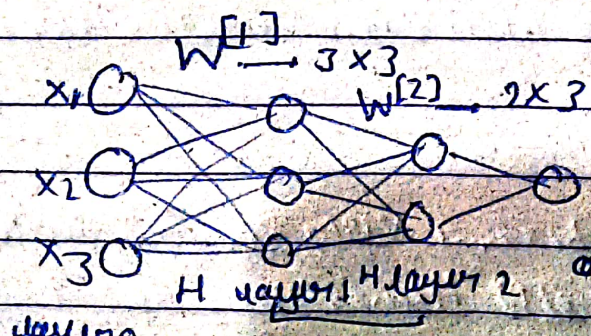
$w_1x_1 + w_2x_2 + b = 0$ acts as a decision boundary separating two classes



→ cannot be separated by a line

In multilayer perceptron to introduce non linearity we apply activation function to the pre activation ex. Sigmoid, tanh, ReLU.

② X (3×1)



without activation fn = linear

$z^{[1]} = W^{[1]}X + b^{[1]}$

$z^{[2]} = W^{[2]}(W^{[1]}X + b^{[1]}) + b^{[2]}$

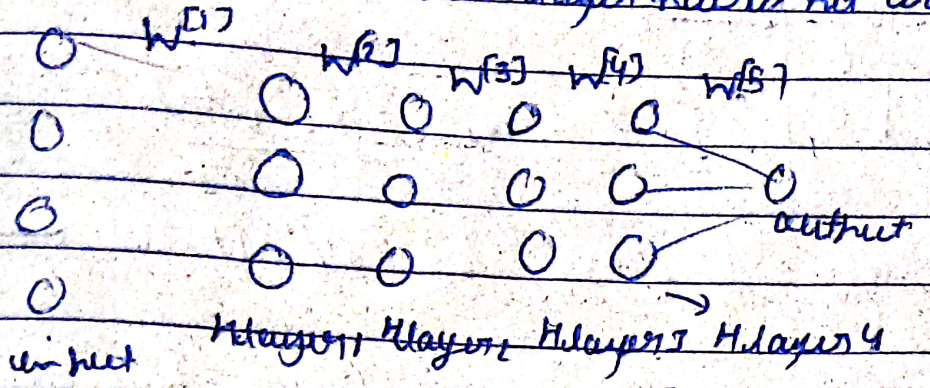
$z^{[2]} = W^{[2]}W^{[1]}X + W^{[2]}b^{[1]} + b^{[2]}$

$y = WX + b$

DATE: / /

- Linear layer stacked on a linear layer is linear; stacking linear layers does not increase model capacity.

1). Consider a 5 hidden layer neural net work



$$\frac{dL}{dW^{[1]}} = \frac{dL}{dy}\frac{da(z^{[4]})}{dz^4}\frac{dz^4}{da(z^3)}\frac{dz^3}{da(z^2)}\frac{da(z^2)}{dz^2}\frac{dz^2}{da(z^1)}\frac{da(z^1)}{dz^1}\frac{dz^1}{dW^{[1]}}$$

$$= \frac{dL}{dy}\prod_{k=2}^{4}\left(\frac{da(z^i)}{dz^i}\frac{dz^i}{da(z^i)}\right)\cdot\frac{da(z^1)}{dz^1}\frac{dz^1}{dW^{[1]}}$$

weight × activation derivatives

So, as the no of layers increases the terms that are being multiplication also increases.

Causes of Shrinking gradients

- Repeated multiplications across many layers
- derivatives of activation fns like (sigmoid, tanh < 1) & weights < 1

Due to this gradients become very small & early layers stop learning. For ReLU derivative is 1 when inputs are positive, unlike sigmoid hence solves the vanishing gradient problem.

DATE: / /

~~the positio~~
it help to capture

Q3 ~~Ans~~

~~Positional encoding is necessary because when we give inputs and output to encoder and decoder respectively~~

* Positional encoding is necessary because it helps to capture the position of words in the input and output sequence which is given to the encoder & decoder which is important because Self attention is permutation invariant for Self attention
"I am Mayank" = "Mayank am I"

* Absolute positional encoding
* uses position embedding matrix whose parameters are learned during training
* Fixed max length

Sinusoidal Positional encoding

* Uses
$$PE(POS, 2i) = \sin\left(\frac{POS}{10000^{2i/d}}\right)$$
$$PE(POS, 2i+1) = \cos\left(\frac{POS}{10000^{2i/d}}\right)$$
* to make positional embedding
* Encodes relative distance similarity
* No learnable parameters
* max length can vary

· ROPE (Rotary positional embeddings)

* It encodes positional information by Rotating query & key vectors.

$$Q_{rotated} = ROPE(Q)$$

PAGE NO.
DATE: / /

PROS

- Attention scores depends on relative distance
- Can be generalised to sequences of unseen lengths
- positional encoding does not change vector magnitude,

Q4 $\text{Attention}(Q, K, V) = \text{softmax}\left(\frac{QK^T}{\sqrt{d_K}}\right)V$

$Q = XW_Q,\ K = XW_K,\ V = XW_V$

Intuition

Q → what the token is looking for

K → what the token offers or how it defines itself

V → The actual information content that is passed on

Attention scores are scaled by $\sqrt{d_K}$ to ensure variance remains constant as Var $\propto d_K$, so for high dimension large variance leads to large values due to which softmax saturates and causes the problem of vanishing gradient.

Diagonal values are usually highest because diagonal entries correspond to $Q_i \cdot K_i^T$ i.e a word attending to itself, since they come from the same token so also has same position they have highest values.

Q5 ~~In multi head attention~~

With one single attention head the model focuses on one type of relationship at a time. But with multiple attention head each head learns a different relationship. eg Head 1 → Find subjects, Head 2 → Tracks verbs

d_{model} : Total embedding dimension of model

h : no of attention heads

d_{head} : Dimension per head

$$d_{head} = \frac{d_{model}}{h}$$

Q6 In Greedy decoding at each step we chooses the token with most probability whereas in Beam search at each step we keep top k sequences, to maximize global sequence probability. Exploring multiple future possibilities.

Ex

at time step: 1		at time step: 2	
Word	Probability	Word	Probability
"I"	0.6	"I am"	0.3
"You"	0.4	"You are"	0.9

Greedy

I am = 0.6 X 0.3 = 0.18

Beam search

I am = 0.6 X 0.3 = 0.18

You are = 0.4 X 0.9 = 0.36 ✓

Q1.

$d_{model} = 768$

$h = 12$

$d_{head} = \frac{d_{model}}{h} = \frac{768}{12} = 64$

• $W_Q, W_K, W_V \in R^{768 \times 768}$

Total parameters = $3 \times 768 \times 768$

= 1769472

Q2 [2.0, 1.0, 0.0]

$$= \left[\frac{e^2}{e^2+e^1+e^0}, \frac{e^1}{e^2+e^1+e^0}, \frac{e^0}{e^2+e^1+e^0} \right]$$

$$= [0.665, 0.245, 0.090]$$