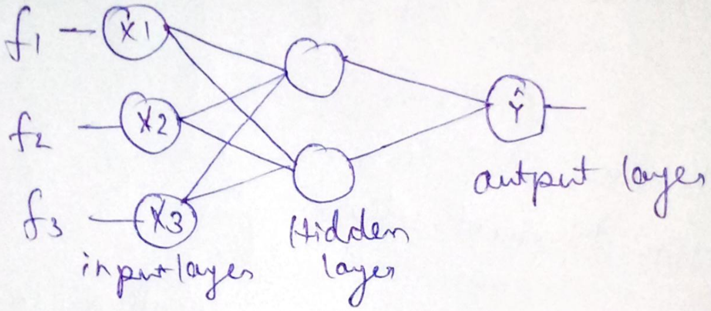
14 Mar 2025

# LAYER NORMALIZATION

what is normalisatiom?

normalisation in deep learning. refers to the process of transforming data or model autputs to have specific statistical properties typically a mean of zero and variance of one

What do we normalize



Benefits of Normalization in Deep learning:

- Improved Training stability

Normalisation helps to stablize and accelerate the training process by reducing the likelihood of the extreme values that can cause gradients to explode or vanish.

- Faster Convergence.

By normalizing inputs or activation models can improve more quickly because the gradients have more consistent magnitude. This allows ~~the~~ for more stable updates during backpropagation.

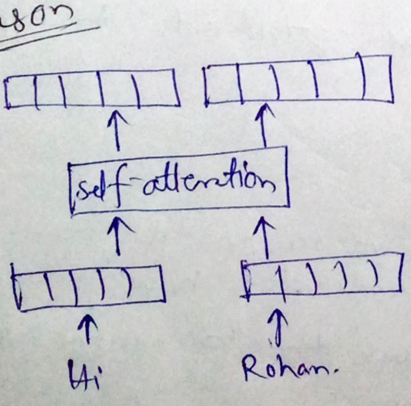
- Mitigating internal covariate shift:
internal covariate shift refers to the change in the distribution of layer inputs during training. normalisation techniques, like batch normalisation help to reduce this shift, making the training process more robust.

- Regularisation Effect:
Some normalisation techniques, like batch normalisation introduce a slight regularizing effect by adding noise to the mini-batches during training. This can help to reduce overfitting.

Normalisation on Activation
— Batch Normalisation
— Layer Normalisation.

why we don't use Batch normalisation in transformer?
→ Batch normalisation not work well on sequential data

Reason



| Review | sentiment |
|---|---|
| Hi Rohan | 1 |
| How are you today | 0 |
| I am Good | 0 |
| You? | 1 |

lets take
each word's embedding dim = 3
Batch size = 2
(provide two sentence.
at a time)

|  |  |  |
|---|---|---|
| 0.2 | 0.45 | 0.71 |
| 0.21 | 0.3 | 0.8 |
| 0 | 0 | 0 |
| 0 | 0 | 0 |

|  |  |  |
|---|---|---|
| 0.1 | 0.5 | 0.34 |
| 0.1 | 0.0 | 0.25 |
| 0.33 | 0.56 | 0.9 |
| 0.11 | 0.4 | 0.54 |

- input

|  |  |  |
|---|---|---|
| 0.8 | 0.33 | 0.b |
| 0.9 | 0.1 | 0.41 |
| 0 | 0 | 0 |
| 0 | 0 | 0 |

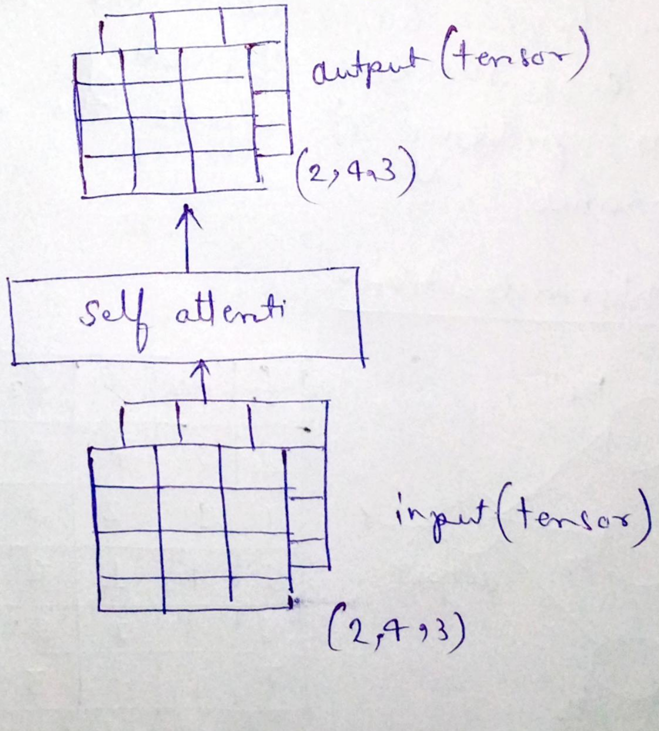
|  |  |  |
|---|---|---|
| 0.1 | 0.2 | 0.33 |
| 0.45 | 0.56 | 0.89 |
| 0.76 | 0.81 | 0.93 |
| 0.43 | 0.47 | 0.97 |

output

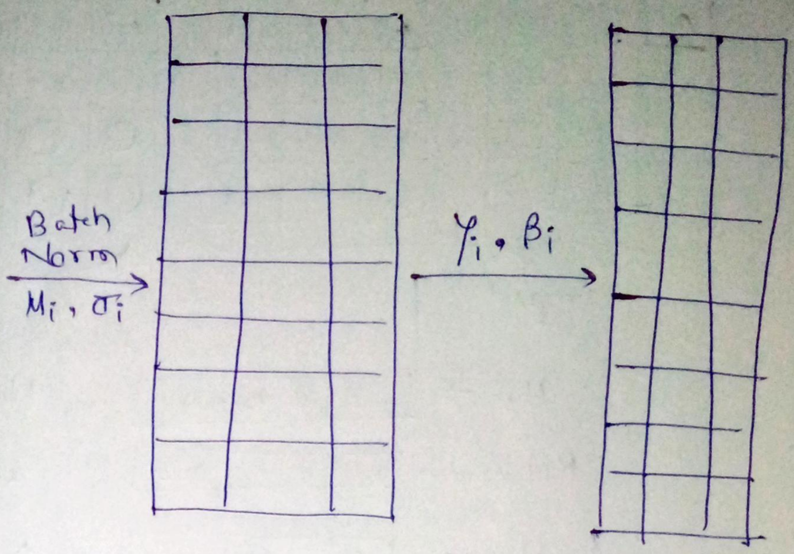
|  |  |  |  |
|---|---|---|---|
| Hi → | 0.2 | 0.45 | 0.71 |
| Rohan → | 0.21 | 0.3 | 0.8 |
| padding → | 0 | 0 | 0 |

|  |  |  |  |
|---|---|---|---|
| How → | 0.1 | 0.5 | 0.34 |
| are → | 0.1 | 0.0 | 0.25 |
| you → | 0.33 | 0.56 | 0.9 |
| today → | 0.11 | 0.4 | 0.54 |

- Here all values are hypothetical.

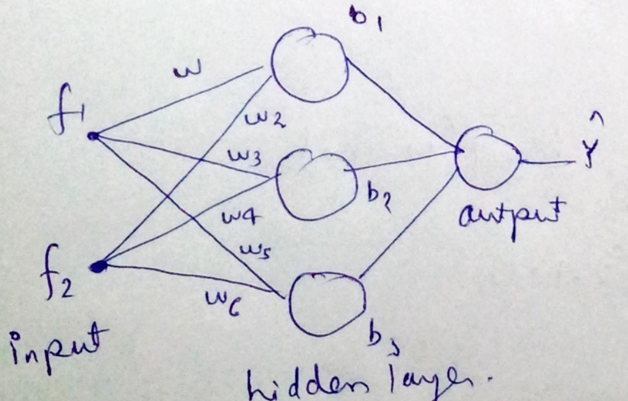


| | | | |
|---|---|---|---|
| Hi. | 0.8 | 0.33 | 0.6 |
| Rohan | 0.9 | 0.1 | 0.41 |
| <padding> | 0 | 0 | 0 |
| <padding> | 0 | 0 | 0 |
| How | 0.1 | 0.2 | 0.33 |
| are | 0.45 | 0.56 | 0.89 |
| you | 0.76 | 0.81 | 0.93 |
| Today. | 0.43 | 0.47 | 0.97 |



- Her we only take a two sentence as a example what if we take a large pdf in which smallest sentence has only 4 word and large sentence have 60 word. due to that we have to add large number of padding.
- due to padding it affects the output result performance.

## Layer Normalisation:



| $f_1$ | $f_2$ | $Z_1$ | $Z_2$ | $Z_3$ |
|---|---|---|---|---|
| 2 | 3 | 7 | 5 | 4 |
| 1 | 1 | 2 | 3 | 4 |
| 5 | 4 | 1 | 2 | 3 |
| 6 | 1 | 7 | 5 | 6 |
| 7 | 1 | 3 | 3 | 4 |

all values are hypothetical

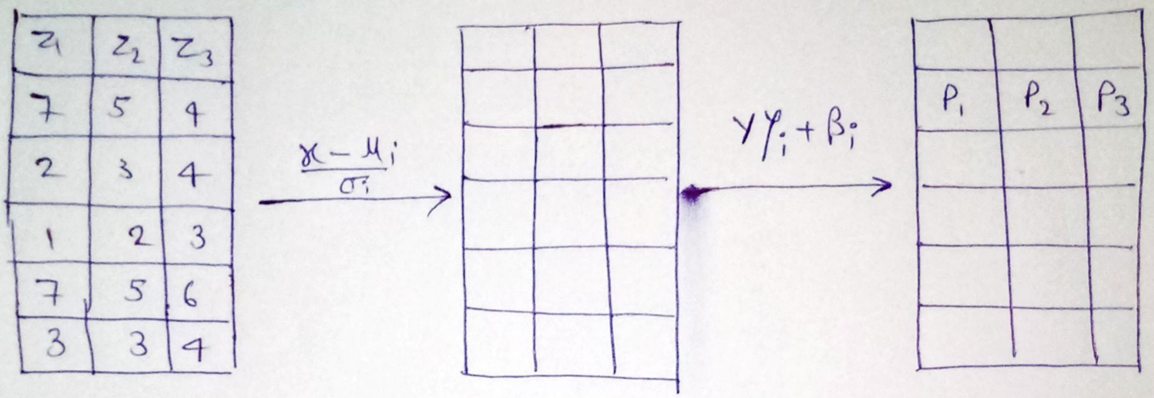
$$Z_1 = 2W_1 + 3W_2 + b_1 = 7$$

$$Z_2 = 2W_3 + 3W_4 + b_2 = 5$$

$$Z_3 = 2W_3 + 3W_6 + b_3 = 4$$

and so on

- and each internal node have its own. $\gamma_i$ and $\beta_i$ parameters
- in batch normalisation it normalise the value across column.
- but in layer normalisation it normalize the value across raws.



| $z_1$ | $z_2$ | $z_3$ |
|---|---|---|
| 7 | 5 | 4 |
| 2 | 3 | 4 |
| 1 | 2 | 3 |
| 7 | 5 | 6 |
| 3 | 3 | 4 |

Example $\frac{7-\mu_1}{\sigma_1} = 0.3$

$$0.3\gamma_1 + \beta_1 = P_1$$

and so on.

Here normalisation across raw-wise

$$\left(\frac{5-\mu_1}{\sigma_1}\right)\gamma_2 + \beta_2 = P_2 \qquad \left(\frac{4-\mu_1}{\sigma_1}\right)\gamma_3 + \beta_3 = P_3$$