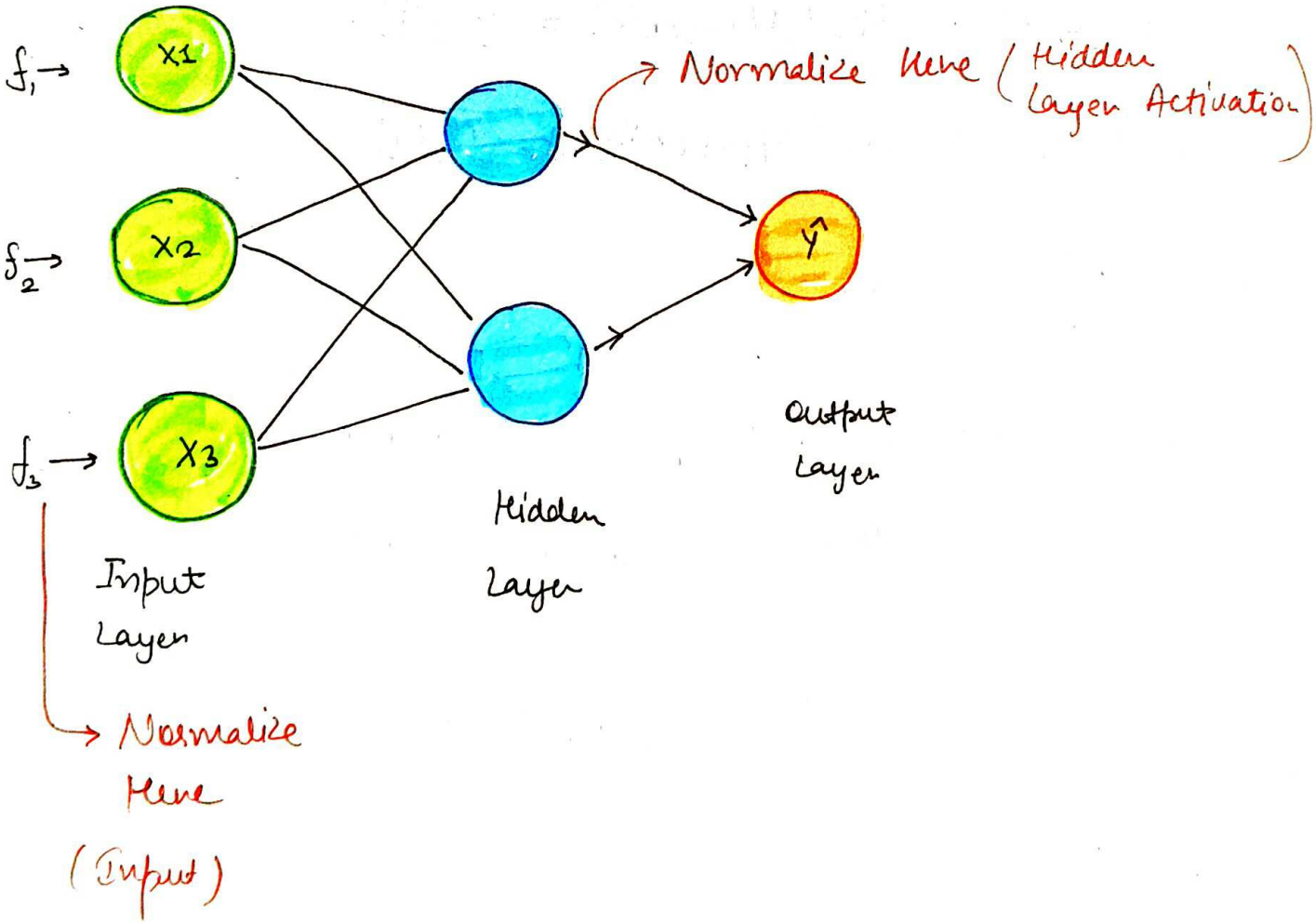
Layer Normalization

What is Normalization?

Normalization in deep learning refers to the process of transforming data as model outputs to have specific statistical properties, typically a mean of zero and a ~~variable~~ variance of one.

What do we normalize?



Benefits of Normalization in Deep Learning

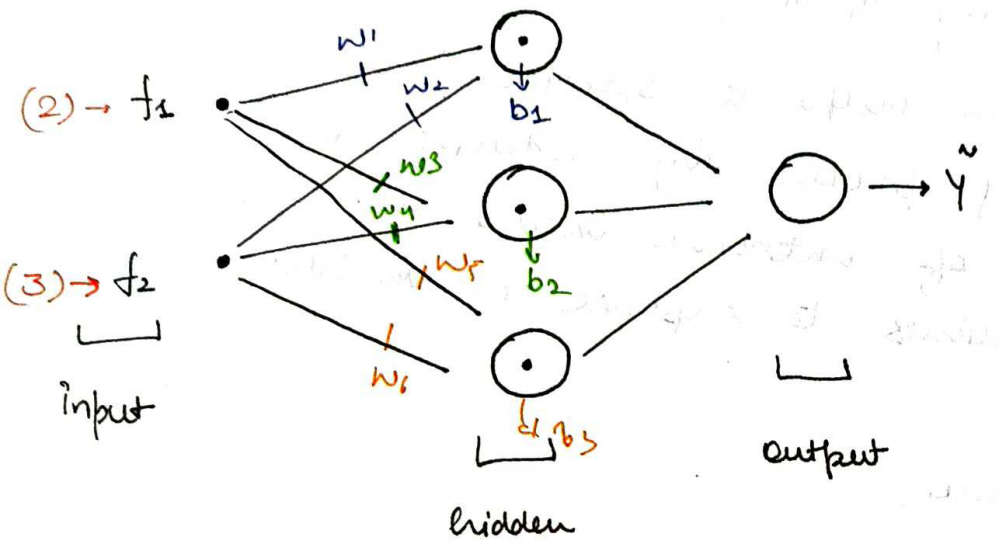
- **Improved Training Stability**:
  → Normalization helps to stablize and accelerate the training process by reducing the likelihood of extreme values that can cause gradients to explode or vanish.

- **Faster Convergence**:
  → By Normalizing inputs or activations, model can converge more quickly because the gradients have more consistent magnitudes. This allows 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. Normalization techniques, like batch normalization, help to reduce this shift, making the training process more robust.

- **Regularization Effect**:
  → Some normalization techniques, like batch normalization, introduces a slight regularization effect by adding noise to the mini-batches during training. This can help to reduce overfitting.

Batch Norm (Revision)



f1	f2
2	3
1	1
5	4
6	1
7	1

batch = 5

First row

$z_1 = 2w_1 + 3w_2 + b_1 = 7$

↳ We have to Normalize z_1

$z_2 = 2w_3 + 3w_4 + b_2 = 5$

↳ Normalize

$z_3 = 2w_5 + 3w_6 + b_3 = 4$

↳ Normalize

Second row

$z_1 = 1w_1 + 1w_2 + b_1 = 2$

↳ Normalize

$z_2 = 1w_3 + 1w_4 + b_2 = 3$

↳ Normalize

$z_3 = 1w_3 + 1w_4 + b_3 = 4$

↳ Normalize

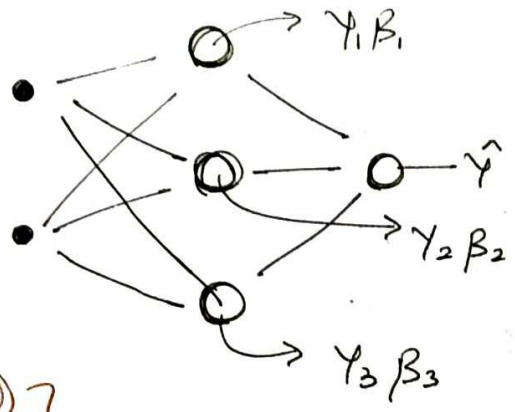
f1	f2	z_1	z_2	z_3
2	3	7	5	4
1	1	2	3	4
5	4	1	2	3
6	1	7	5	6
7	1	3	3	4

Normalize

$z_1 \rightarrow \mu_1, \sigma_1$

$z_2 \rightarrow \mu_2, \sigma_2$

$z_3 \rightarrow \mu_3, \sigma_3$



Normalize z_1

$$\frac{7-\mu_1}{\sigma_1} = 0.36$$

$$0.36\, Y_1 + \beta_1 = 0.36$$

(1) ← default (0) ← default

z_2

$$\frac{5-\mu_2}{\sigma_2} = -0.21 Y_1 + \beta_2 = -0.21$$

$$\frac{2-\mu_1}{\sigma_1} = 0.71 Y_1 + \beta_1 = 0.71$$

z_3

$$\frac{4-\mu_3}{\sigma_3} = 0.12 Y_3 + \beta_3 = 5$$

Why don't we use Batch Norm in Transformers?

↳ Batch Normalization is not good work with self Attention.

↳ Batch Normalization is not good with Sequential Data.

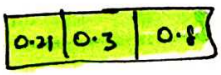
Bach-Normalization with Self Attention

	Review	Sentiment
r_1	Hi Nitish	1
r_2	How are you Today	0
r_3	I am good	0
r_4	You?	1

Embedding dimension = 3

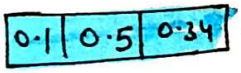
Batch Size = 2 → ek sath 2 rows

Hi

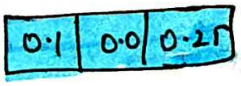


Nitish

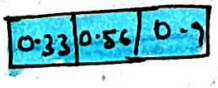
→ Embedding



How



are



You

today

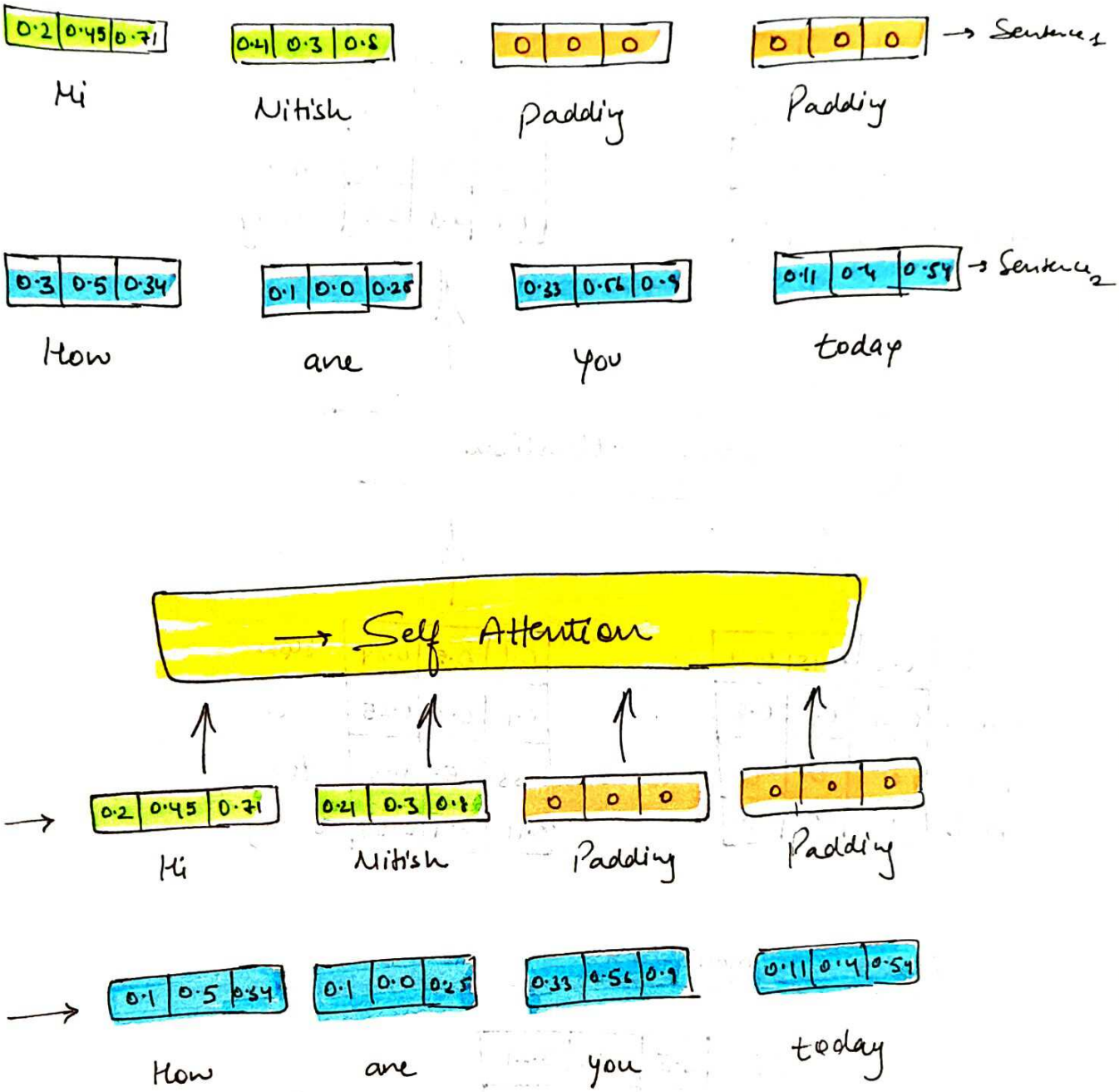
Embedding

* Both embedding are different number
  first row → 2 embedding
  second row → 4 embedding

* Using padding for same number.

→ Hi Nitish <pad> <pad>

→ How are You today


0.2 0.45 0.71
Hi
0.41 0.3 0.8
Nitish
0 0 0
Paddiy
0 0 0
Paddiy
→ Sentence 1
0.3 0.5 0.34
How
0.1 0.0 0.25
are
0.33 0.56 0.9
you
0.11 0.4 0.54
today
→ Sentence 2
→ Self Attention
0.2 0.45 0.71
Hi
0.21 0.3 0.8
Nitish
0 0 0
Padding
0 0 0
Padding
0.1 0.5 0.34
How
0.1 0.0 0.25
are
0.33 0.56 0.9
you
0.11 0.4 0.54
today
Self Attention
Hi 0.2 0.45 0.71
Nitish 0.21 0.3 0.8
<Pad> 0 0 0
<Pad> 0 0 0
0.1 0.5 0.34 How
0.1 0.0 0.25 are
0.33 0.56 0.9 you
0.11 0.4 0.54 Today

Hi	6.5	2.41	3.21
Nitish	2.21	0.4	3.6
<pad>	0	0	0
<pad>	0	0	0

7.5	9.2	1.5	How
2.2	1.1	6.7	are
2.9	6	9	you
9.9	2.3	6.5	today

Self Attention

Hi	0.2	0.45	0.71
Nitish	0.21	0.3	0.8
<pad>	0	0	0
<pad>	0	0	0

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

Batch Normalization

Hi	6.5	2.41	3.21
Nitish	2.21	0.4	3.6
<pad>	0	0	0
<pad>	0	0	0
How	7.5	9.2	1.5
are	2.2	1.1	6.7
you	2.9	6	9
today	9.9	2.3	6.5

μ_1 σ_1 μ_2 σ_2 μ_3 σ_3

$\gamma_1\beta_1$ $\gamma_2\beta_2$ $\gamma_3\beta_3$

Let say

32 → Batch

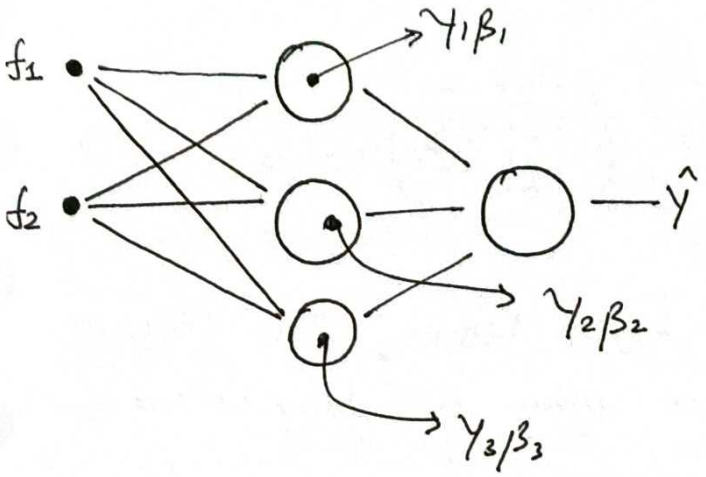
Most of sentence = 32 words

Larges sentence = 100 words

To Match No. of words we use padding.

After Padding, mean and std are not true. Because Maximum zeros are present, We can not use Batch Norm.

Layer Norm



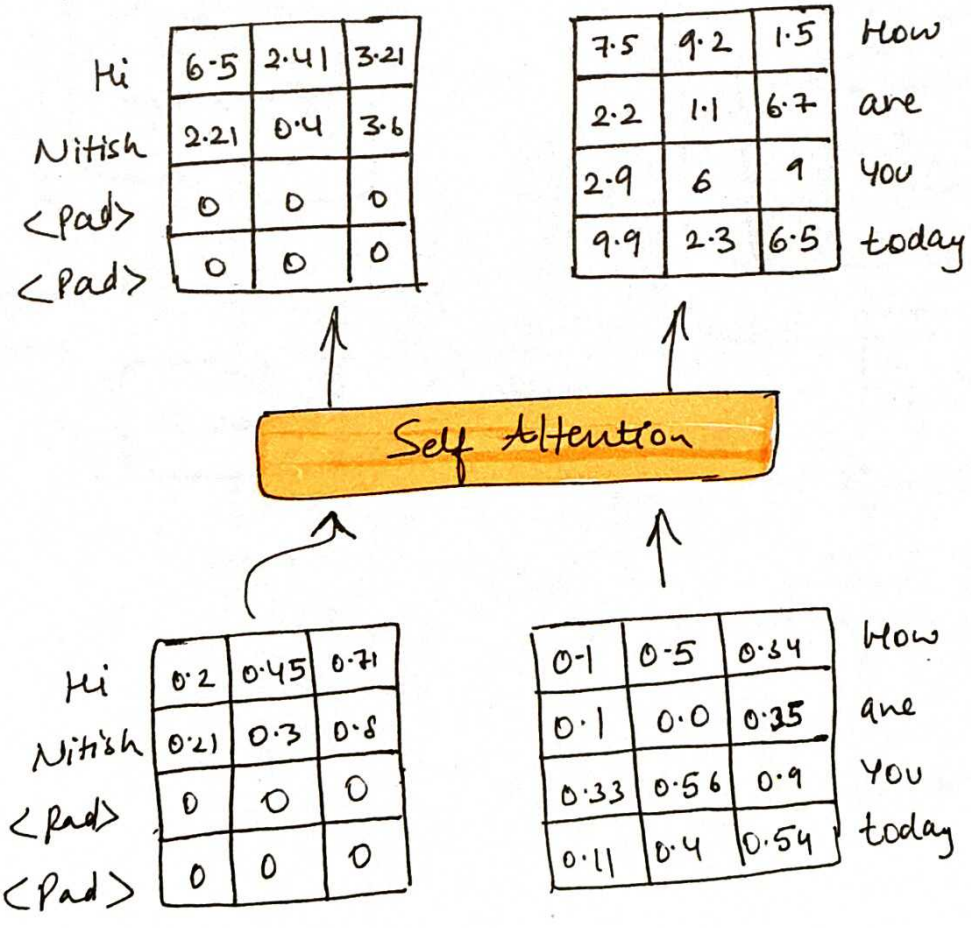
Across features

f_1	f_2	z_1	z_2	z_3	
2	3	7	5	4	μ_1, σ_1
1	1	2	3	4	μ_2, σ_2
5	4	1	2	3	μ_3, σ_3
6	1	7	5	6	σ_4, μ_4
7	1	3	3	4	μ_5, σ_5

across batch — μ_0 σ_2

$$\frac{7-\mu_1}{\sigma_1} = 0.3\gamma_1 + \beta_1$$

$$\left(\frac{5-\mu_1}{\sigma_1}\right)\gamma_2 + \beta_2$$



Hi	6.5	2.41	3.21
Nitish	2.21	0.4	3.6
<Pad>	0	0	0
<Pad>	0	0	0
How	7.5	9.2	1.5
are	2.2	1.1	6.7
You	2.9	6	9
today	9.9	2.3	6.5

μ_1, σ_1

$$\left(\frac{6.5 - \mu_1}{\sigma_1}\right) \gamma_1 + \beta_1$$

$$\left(\frac{2.41 - \mu_2}{\sigma_1}\right) \gamma_2 + \beta_2$$

* In Layer Norm, Zero doesn't affect other embedding value.

$$\left(\frac{0 - \mu_3}{\sigma_3}\right) \gamma_1 + \beta_1 = 0$$

* Norm Batch, will affect the other embeddig. Because we are calculate in Horizontly.