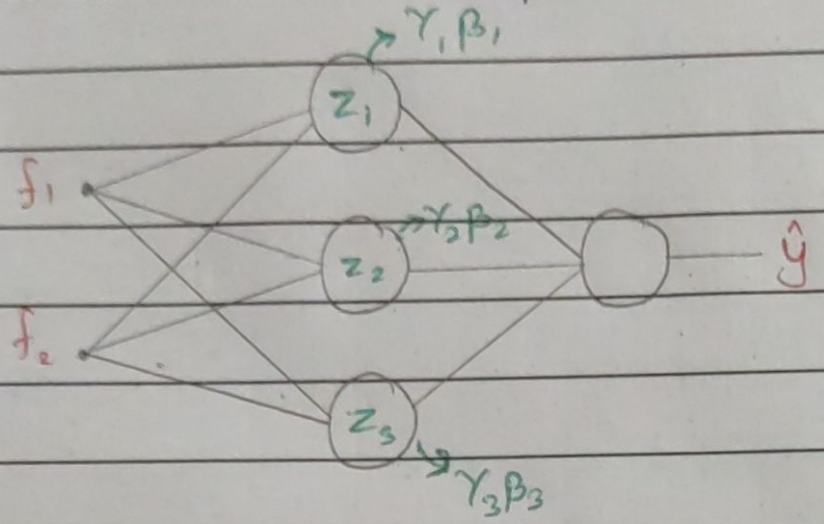
Consider same setup as of Batch Normalization



		$\gamma_1\beta_1$ z_1	$\gamma_2\beta_2$ z_2	$\gamma_3\beta_3$ z_3	
f_1	f_2				
2	3	7	5	4	$\mu_1\sigma_1$
1	1	2	3	4	$\mu_2\sigma_2$
5	4	1	2	3	$\mu_3\sigma_3$
6	1	7	5	6	$\mu_4\sigma_4$
7	1	3	3	4	$\mu_5\sigma_5$

In layer normalization μ & σ are calculated across features (row wise) & are readjusted using the γ & β values of the respective columns (values of the preactivation) ($z_1 \to \gamma_1\beta_1$, $z_2 \to \gamma_2\beta_2$, $z_3 \to \gamma_3\beta_3$)

In same way for normalizing the values of text embedding matrix we use Layer Normalization

	$\gamma_1\beta_1$	$\gamma_2\beta_2$	$\gamma_3\beta_3$	
Hi	6.5	2.41	3.21	$\mu_1\sigma_1$
Probal	2.21	0.4	3.6	$\mu_2\sigma_2$
<Pad>	0	0	0	$\mu_3\sigma_3$
<Pad>	0	0	0	$\mu_4\sigma_4$
How	7.5	9.2	1.5	$\mu_5\sigma_5$
are	2.2	1.1	6.7	$\mu_6\sigma_6$
you	2.9	6	9	$\mu_7\sigma_7$
today	9.9	2.3	6.5	$\mu_8\sigma_8$

Here, as we can see, each embedding vector has it's own statistical representation (μ & σ) & the zero padding embedding vectors are not affecting the true statistical representations of the embedding matrix.

Eg. For 1st dimension of embedding vector of word "How" -

$$\frac{7.5 - \mu_5}{\sigma_5} = 0.7 \to \text{Hypothetical}$$

{ Row wise μ_5 & σ_5
{ Column wise γ_1 & β_1

Readjustment: $0.7\gamma_1 + \beta_1$