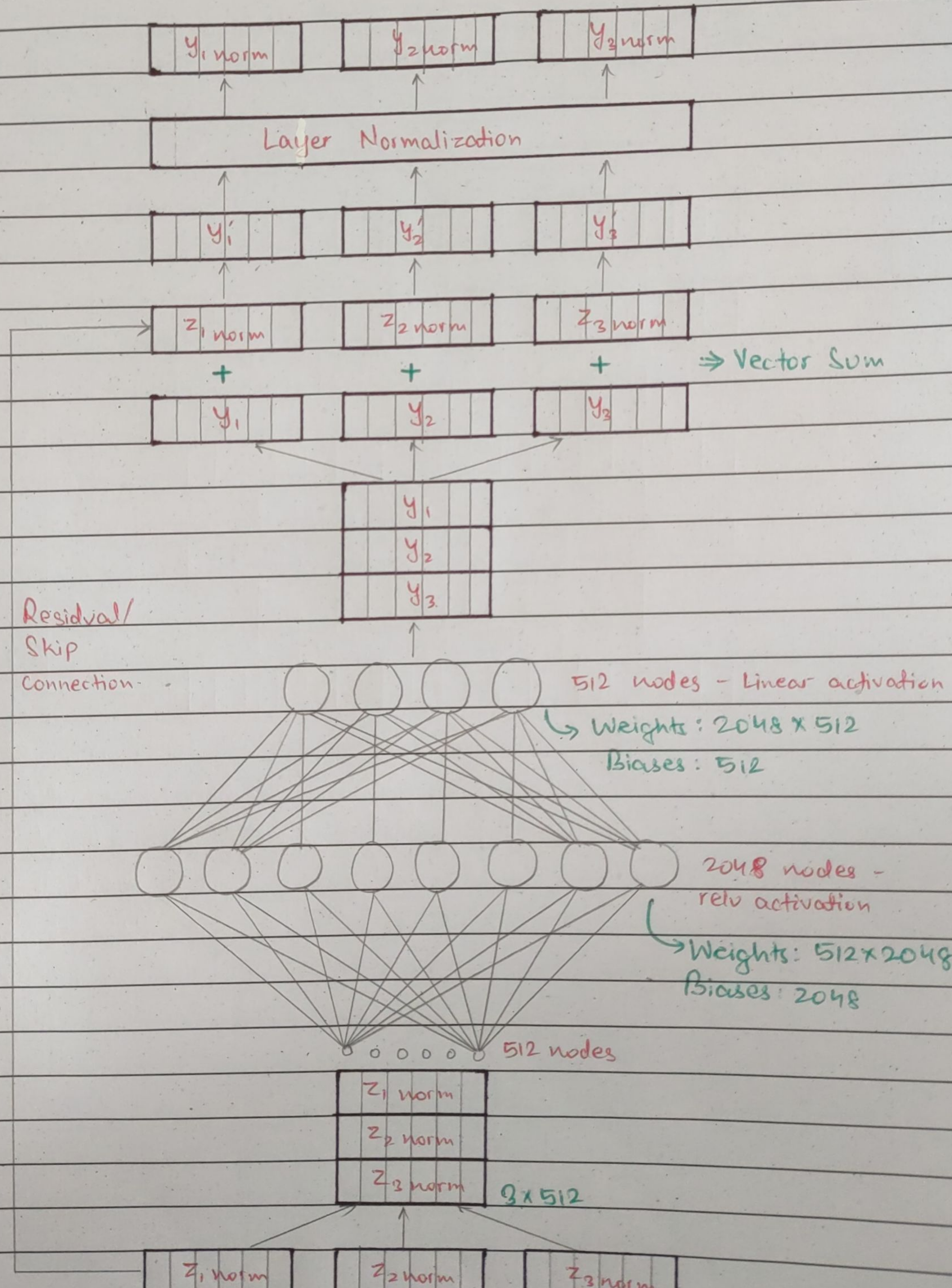
y1 norm
y2 norm
y3 norm
Layer Normalization
y1'
y2'
y3'
z1 norm
z2 norm
z3 norm
+
+
+
⇒ Vector Sum
y1
y2
y3
y1
y2
y3
Residual/
Skip
Connection
512 nodes - Linear activation
Weights : 2048 x 512
Biases : 512
2048 nodes -
relu activation
Weights: 512x2048
Biases: 2048
512 nodes
z1 norm
z2 norm
z3 norm
3x512
z1 norm
z2 norm
z3 norm

* z_{1norm}, z_{2norm} & z_{3norm} from previous stage is taken as input. They are stacked as a 3×512 matrix that is fed into a Feed Forward NN.
* The NN has 2 layers (excluding input layer). The hidden layer has 2048 neurons all with ReLU activation & the output layer has 512 neurons all with Linear activation.
* Hidden layer has 512×2048 weights & 2048 biases & output layer has 2048×512 weights & 512 biases.
* The input is of 3×512 so the calculation will be -

$$\text{ReLU}[(3\times512)\times(512\times2048)+2048] \quad \text{At hidden layer}$$

$$\Rightarrow \text{ReLU}[\underbrace{(3\times2048)}_{\text{Resultant matrix}}+2048]$$

* For the output layer the input is of 3×2048 dimension, so -

$$\text{Linear}[(3\times2048)\times(2048\times512)+512]$$

$$\Rightarrow \text{Linear}[\underbrace{(3\times512)}_{\text{Resultant matrix}}+512]$$

* Through these operations we get the ~~orig~~ output of original dimensions of 3×512 matrix. This can be broken down into 3 512 dimensional vectors → y_1, y_2, y_3
* Through the residual/skip connection we add original z_{1norm}, z_{2norm}, z_{3norm} to y_1, y_2 & y_3. The resultant vectors are y_1', y_2' & y_3'.
* y_1', y_2' & y_3' are then passed to Layer normalization to normalize the embedding values.
* The final output we get are y_{1norm}, y_{2norm} & y_{3norm}
* y_{1norm}, y_{2norm} & y_{3norm} are given as input to the next encoder block & the whole process repeats for 6 encoder blocks.
* The output of the final encoder block is given as input to the decoder.
* The output of the encoder block is 3×512 dimensional.