

Ink to LaTeX

Zirui Li, Jianxiong Li, Youjia Qian, Haozhuo Liu

Motivation

$$n \cdot \log(d-1) - n \cdot \log(d_{\min}) - \alpha \left(\left(\sum_{i=1}^n \log(d_i) \right) - \left(\sum_{i=1}^n \log(d_{\min}) \right) \right)$$

Introduction

$$A^{(m+1)} = \psi^{(m)}(A^{(m)})$$

Expression Image



\$\$

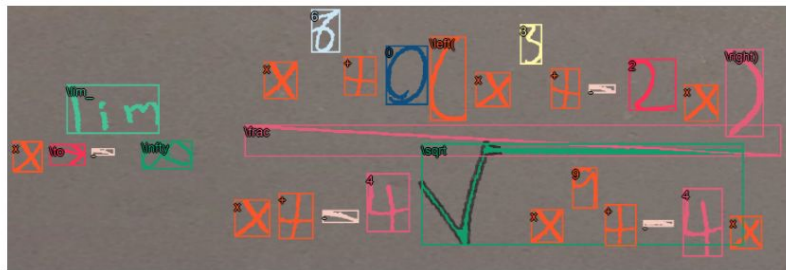
`A^{(m+1)} = \psi^{(m)}(A^{(m)})`

\$\$

LaTeX Code

Calculus Math Handwriting Recognition Dataset

$$\lim_{x \rightarrow -\infty} \frac{x^8 + O(x^3 + -2x)}{x + -4\sqrt{x^9 + -4x}}$$

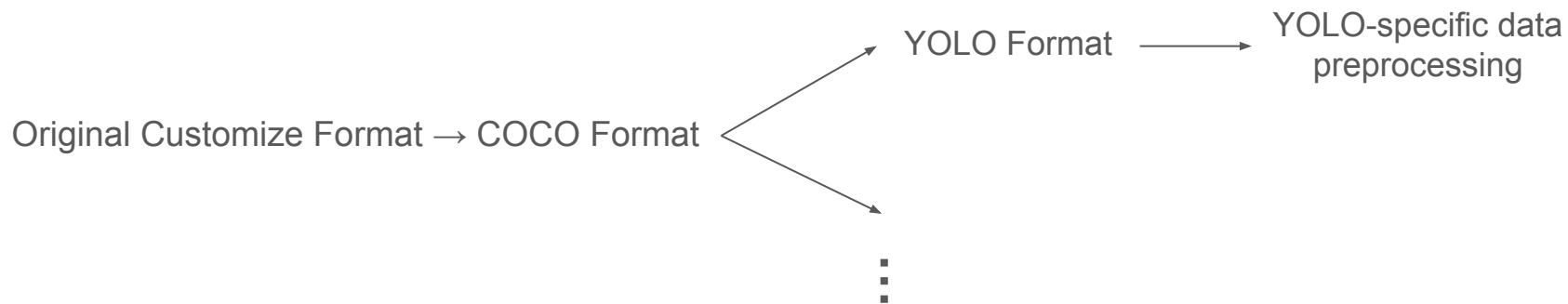


- 100,000 images in total
- Split with 6:2:2

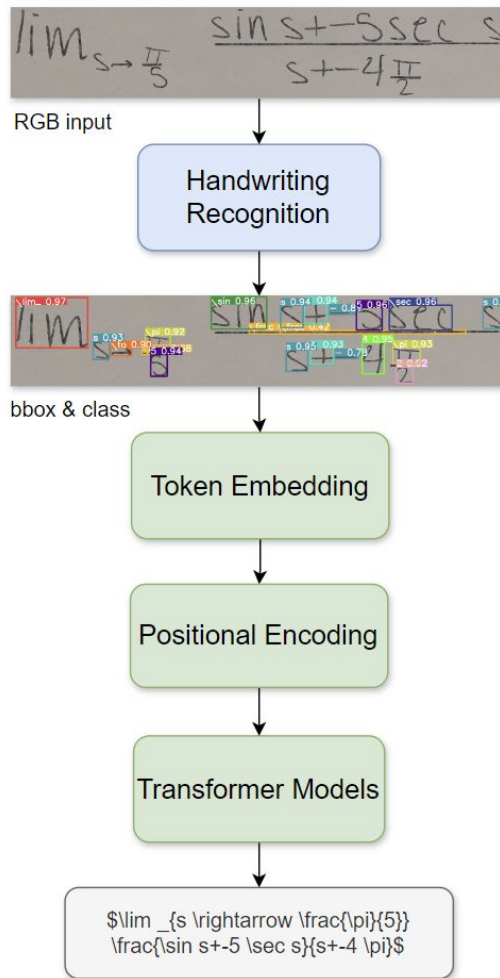
- Bbox
- Class
- Segmentation Mask

<https://www.kaggle.com/datasets/aidapearson/ocr-data>

Dataset Preprocess



Model Outline

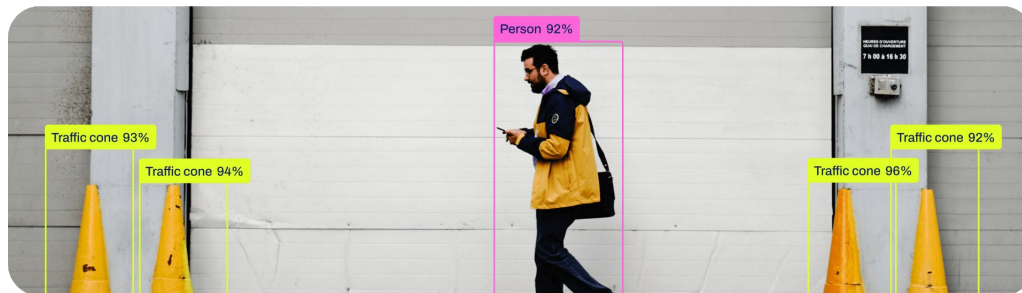


Stage 1: Handwriting Recognition

Need spatial relationship
between characters

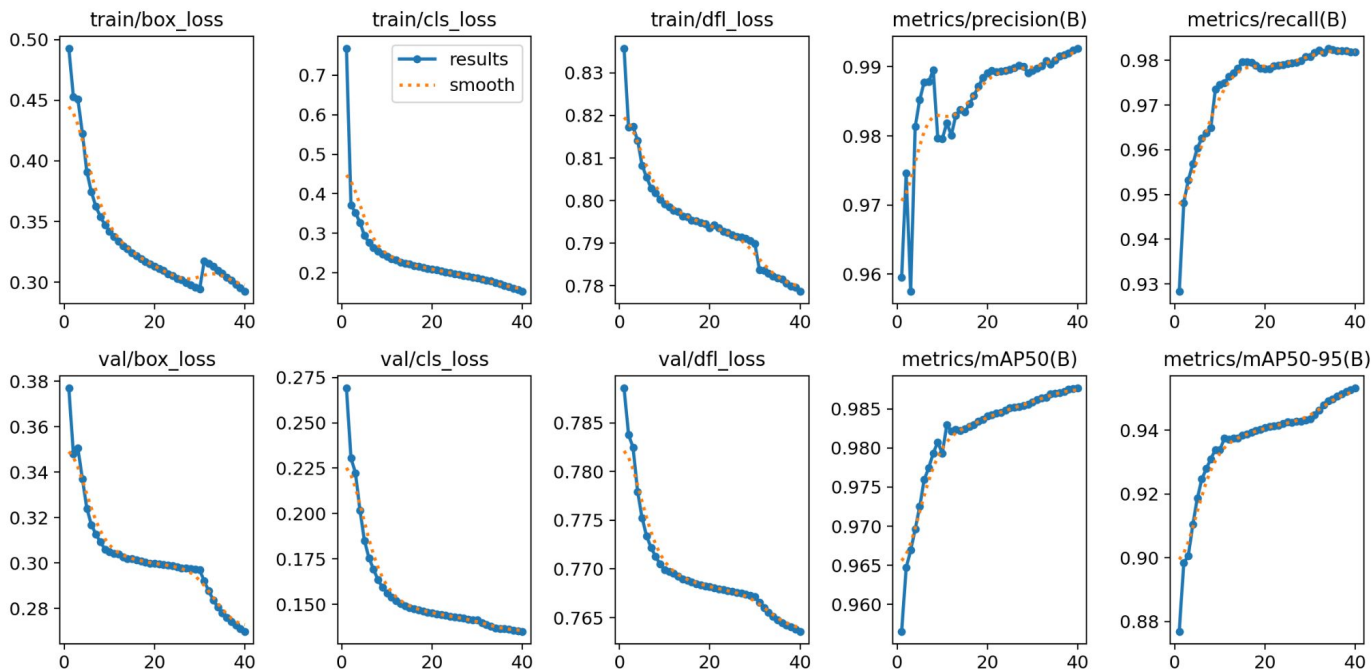
YOLOv8

- Anchor Free
- Predict Diverse Shapes



<https://docs.ultralytics.com/tasks/detect/>

Stage 1 Result



Stage 1 Result

$$\lim_{r \rightarrow 2} \frac{\sin r}{r} \sqrt{r(2 + -2r)}$$

The image shows the same handwritten equation as above, but with various digital annotations in different colors. A purple box encloses the entire expression. A red box highlights the limit symbol and the variable r . A green box highlights the \sin function and its argument 0.95 . A blue box highlights the square root symbol and its argument. A yellow box highlights the fraction bar. A pink box highlights the variable r in the denominator. A cyan box highlights the $2 + -2r$ part of the square root argument. A magenta box highlights the 2 and $-2r$ terms. A light blue box highlights the $0.95 + 0.96$ part of the square root argument. A dark blue box highlights the 2 and -0.83 terms. A light green box highlights the 0.95 and 0.96 terms. A light orange box highlights the 0.82 term. A light purple box highlights the 0.896 term. A light blue box highlights the 0.91 term. A light green box highlights the 0.90 term. A light orange box highlights the 0.94 term. A light purple box highlights the 0.90 term.

Stage 2 : Composing bbox to LaTeX

Loss function

- Cross entropy loss: $L_{CE} = - \sum_{i=1}^N y_i \log(\hat{y}_i)$

The output is a sequence of tokens from a defined vocabulary.

It measures the difference between the predicted probability distribution for each token in the vocabulary and the actual distribution

Stage 2: Composing bbox to LaTeX

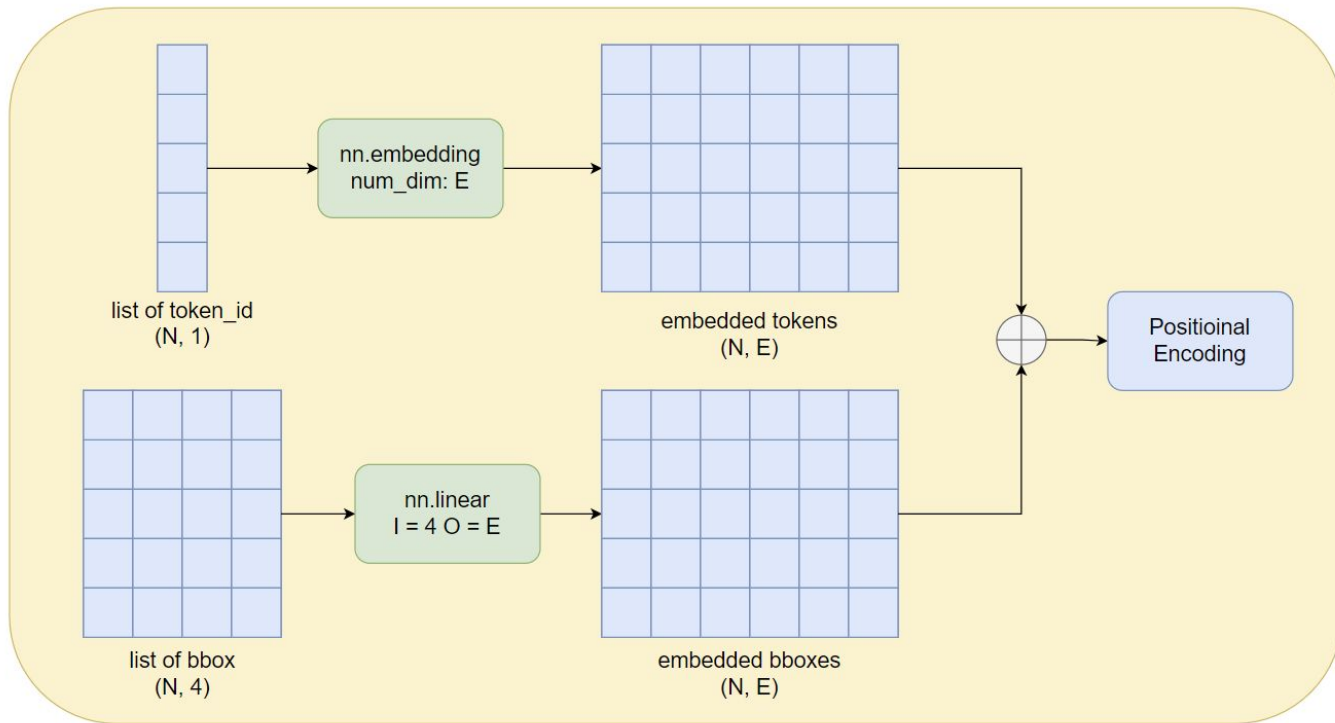
The Transformer model:



Step 1: Token Embedding

N = The number of bbox
 $E = 512$
Dimension of Transformer

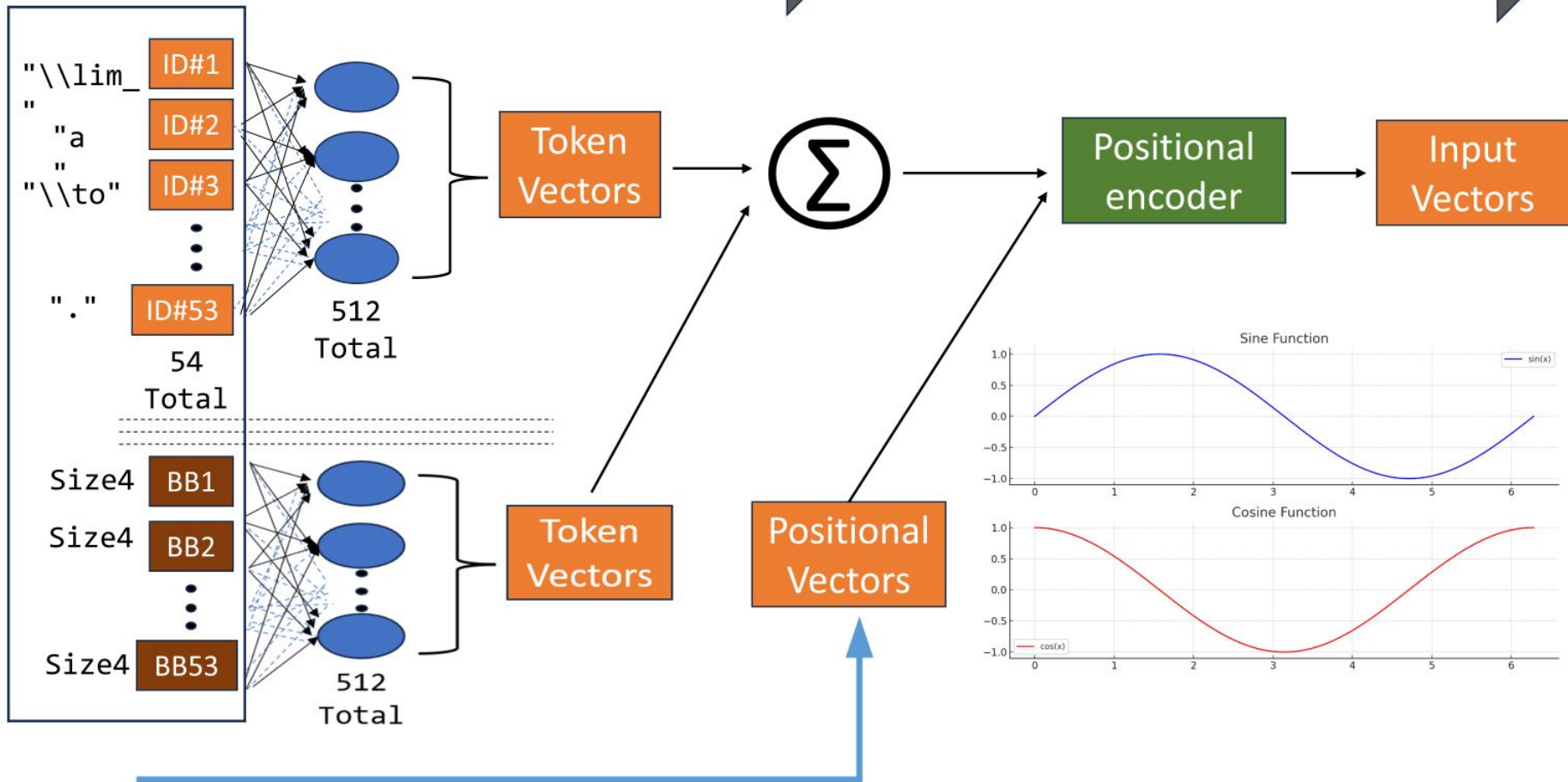
In order to make the model understand the spatial relationship of each character in an image...



Tokens

Token embedding

Positional embedding



Positions

From 0 to max_len - 1]



Positions
Tensor

[max_len, 1]



Frequencies

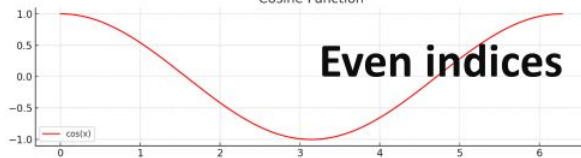
```
torch.exp(torch.arange(0, d_model, 2) * -  
(math.log(10000.0) / d_model))
```



Sine Function



Cosine Function



Input
Vectors



Token
Vectors

Encoder & Decoder

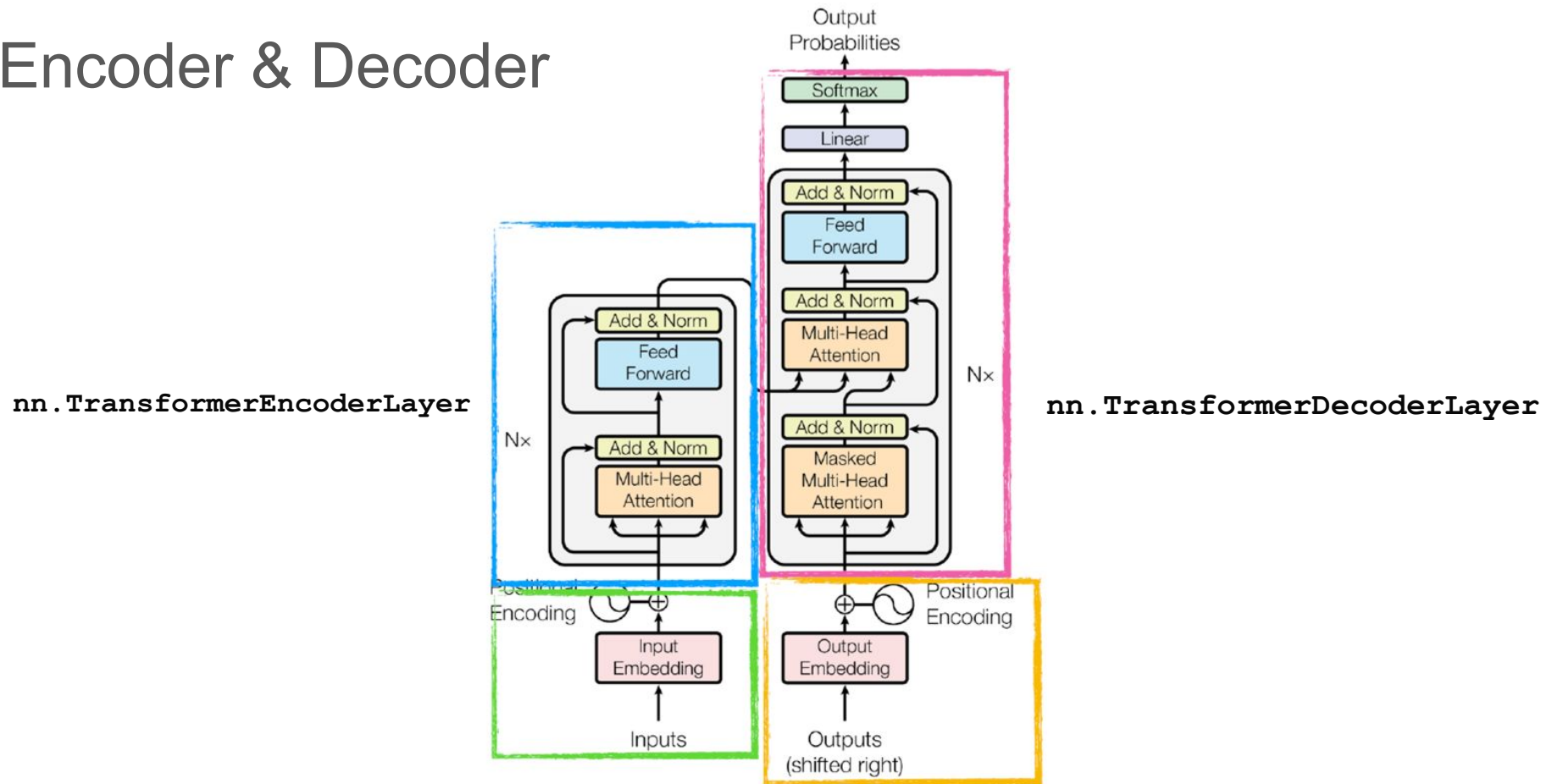


Figure 1: The Transformer - model architecture.

Evaluation– Precision, Recall, F1-score

True/False Positives and Negatives

True Positives (TP): These are the tokens that are correctly predicted by the model.

False Positives (FP): These are the tokens that the model predicted incorrectly

False Negatives (FN): These are the tokens that the model should have predicted but did not.

Simple example

Actual LaTeX: $\lim_{y \rightarrow \infty} (y - 3)$

Model's Prediction: $\lim_{y \rightarrow \infty} (y - 2)$

True Positives (TP): These are: $\lim_{y \rightarrow \infty} (y - 3)$

Total TP: 10

False Positives (FP): Since 2 is predicted instead of 3, it is a false positive.

Total FP: 1

False Negatives (FN): The 3 was missed because 2 was predicted instead.

Total FN: 1

Evaluation

WHY them?

Precision

$$\frac{\text{True Positives}}{\text{True Positives} + \text{False Positives}} = \frac{\text{N. of Correctly Predicted Positive Instances}}{\text{N. of Total Positive Predictions you Made}}$$

-accurate

Recall

$$\frac{\text{True Positives}}{\text{True Positives} + \text{False Negatives}} = \frac{\text{N. of Correctly Predicted Positive Instances}}{\text{N. of Total Positive Instances in the Dataset}}$$

-identified

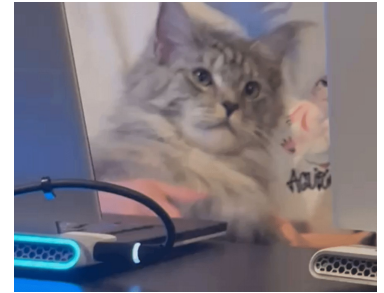
F1-score

$$2 * \frac{\text{Precision} * \text{Recall}}{\text{Precision} + \text{Recall}}$$

-balanced

Thanks for Listening!

Q & A





&

