Probability of the entire token sequence is therefore given by Equation 4.

$$\boldsymbol{p}_t \;:\; \{1,\ldots,V\} \rightarrow [0,1] \quad ;\; \boldsymbol{Y}_t \sim \boldsymbol{p}_t \tag{2}$$

$$\boldsymbol{p}_t(\boldsymbol{y}_t) \;:=\; \mathbb{P}(\boldsymbol{Y}_t{=}\boldsymbol{y}_t|\boldsymbol{y}_{<t}, \boldsymbol{I}) \tag{3}$$

$$\mathbb{P}(\boldsymbol{y}|\boldsymbol{I}) \;=\; \prod_{t=1}^{\tau} \boldsymbol{p}_t(\boldsymbol{y}_t) \tag{4}$$

As is typical with sequence generators, the training objective here is to maximize probability of the target sequence $\boldsymbol{y}^{GT}$. We use the standard per-word cross-entropy objective (Equation 5), modified slightly for the mini-batch (Equation 6). We did not use any regularization objective, relying instead on dropout, data-augmentations and synthetic data to provide regularization.

$$\mathcal{L}_{seq} \;=\; -\frac{1}{\tau}\sum_{t} \ln\left(\boldsymbol{p}_t(\boldsymbol{y}_t^{GT})\right) \quad ;\; \tau \equiv \text{sequence length} \tag{5}$$

$$\mathcal{L}_{batch} \;=\; -\frac{1}{n}\sum_{batch}\sum_{t} \ln\left(\boldsymbol{p}_t(\boldsymbol{y}_t^{GT})\right) \quad ;\; n \equiv \#\text{ of tokens in batch} \tag{6}$$

The final Linear layer of the decoder (Figure. 3a) is a 1x1 convolution function that produces logits which are then normalized by softmax to produce $\boldsymbol{p}_t$.

Combination of Vision and NLP One of the strengths of our architecture is in the combination of Vision and Language models. CNNs such as ResNet are considered best for processing image data. And Transformers are considered best for Language Modeling (LM) and Natural Language Understanding (NLU) tasks [7, 24, 25], posessing properties that are very useful in dealing with noisy and incomplete text that often occurs in real handwriting. Having both the visual feature map and a language model, the model can do a much better job than one relying on visual features alone.

Inference We use simple greedy decoding, which picks the highest probability token at each step. Beam search decoding [8] did not yield any accuracy improvement indicating that the model is quite opinionated / confident.

4 Training Configuration and Procedure

The base configuration uses grayscale images scaled down to 140-150 dots per inch. Higher resolutions yielded slightly better accuracy at the cost of compute and memory. We use the 34-layer configuration of ResNet, but have also successfully trained the 18-layer and 50-layer configurations; larger models tending to do better in general as expected.

The following is the base configuration of the Transformer stack:

- N (number of layers) = 6