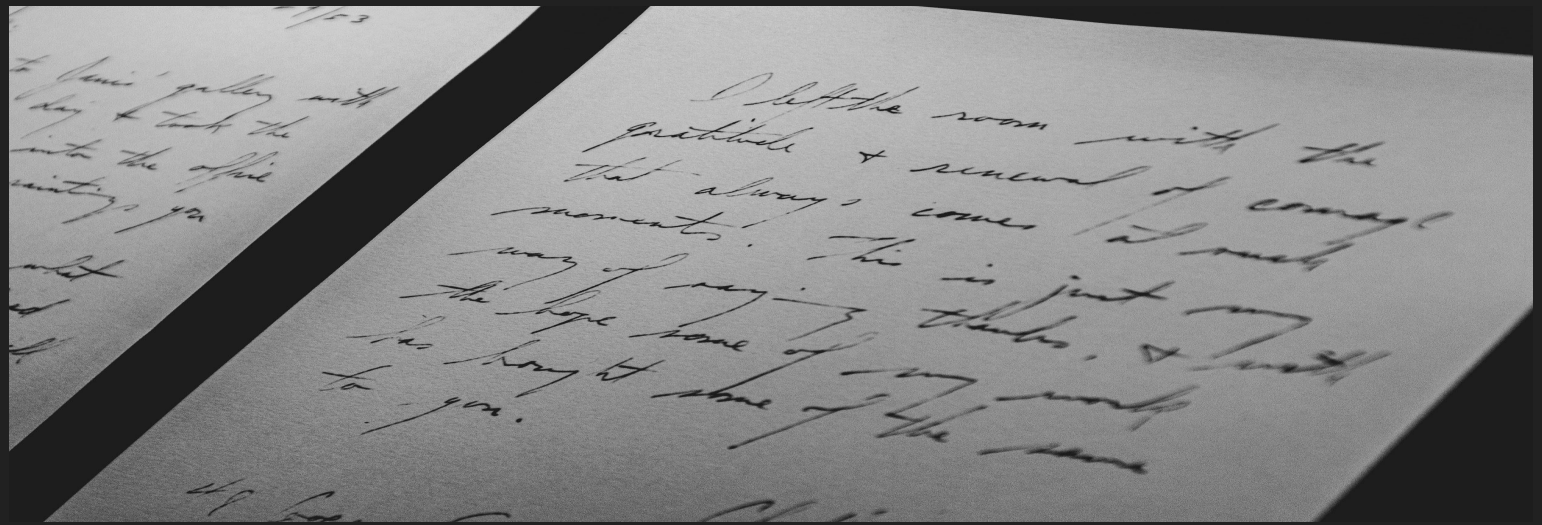
Handwritten Text Recognition



Rob Zifchak
March 2020

The Problem

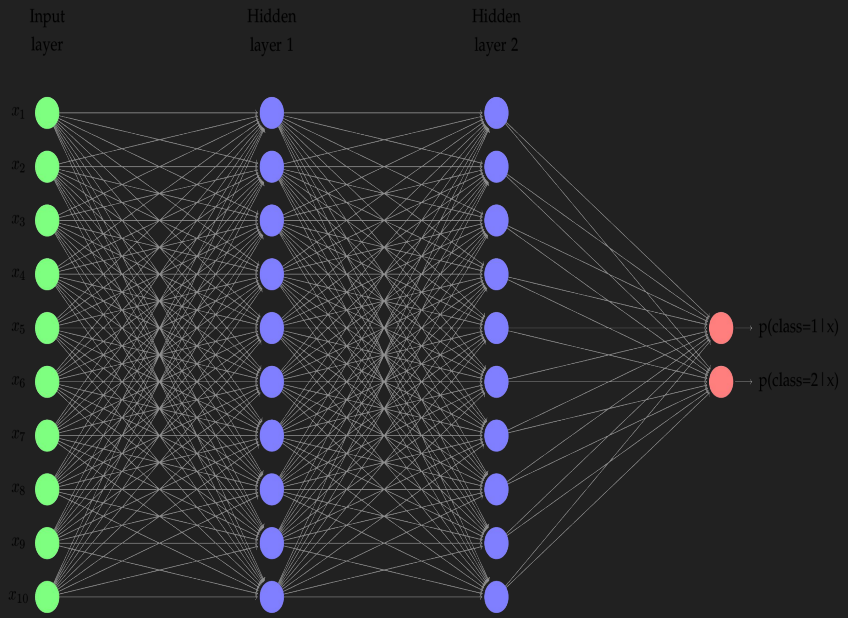
Can images of handwritten text be properly interpreted by a computer?

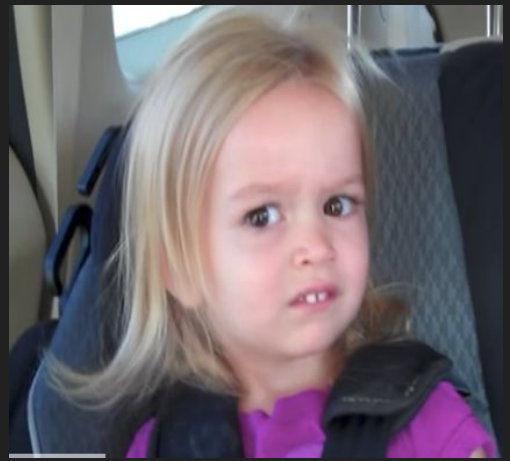
There are plenty of use cases for HTR(Handwritten Text Recognition) technologies today:

- Extracting handwritten fields from documents
- Signature verification
- Reading addresses from mail
- Scanning/reading Checks
- many more

Proposed Objective

Train Convolutional Neural Network to extract features from training images, and correctly classify handwritten words.





Data Details

IAM Handwriting Database

FKI Research Group

- 657 unique writers handwriting from 1,539 pages of scanned text containing 12,224 unique words

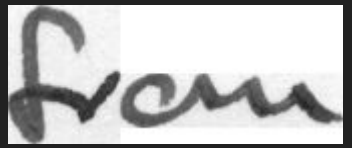
- After removing damaged samples and imbalances in class distributions, overall modeling data contains 21,308 samples, and 78 unique words.

Samples Preview

- Image samples are a variety of dimensions
- Some writers handwriting is messy and hard to classify, even to the human eye.
- Nested file structures are not conducive to word classification

“been”

“talks”



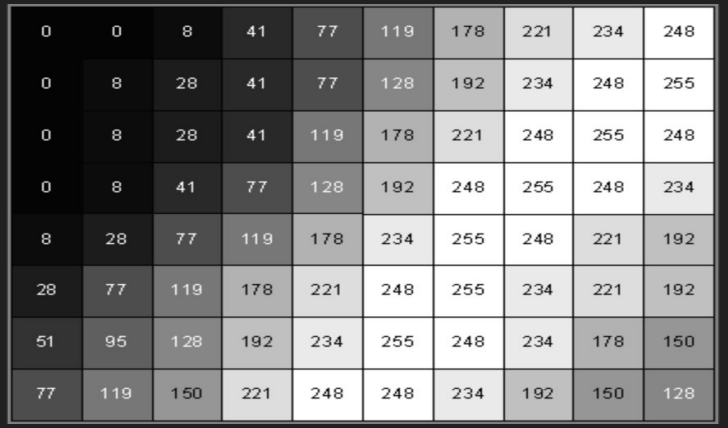
“from”

“the”

Preprocessing

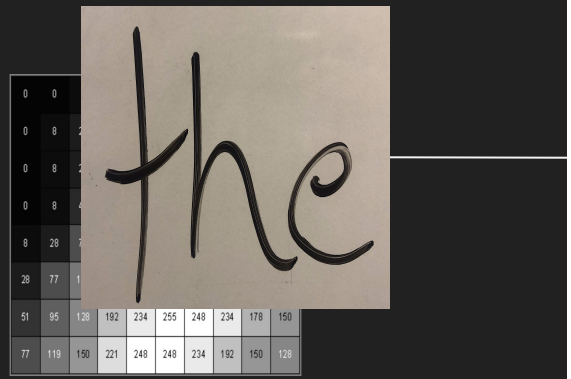
- For properly loading a CNN, images must be sorted and stored in unique directories based on their class labels.

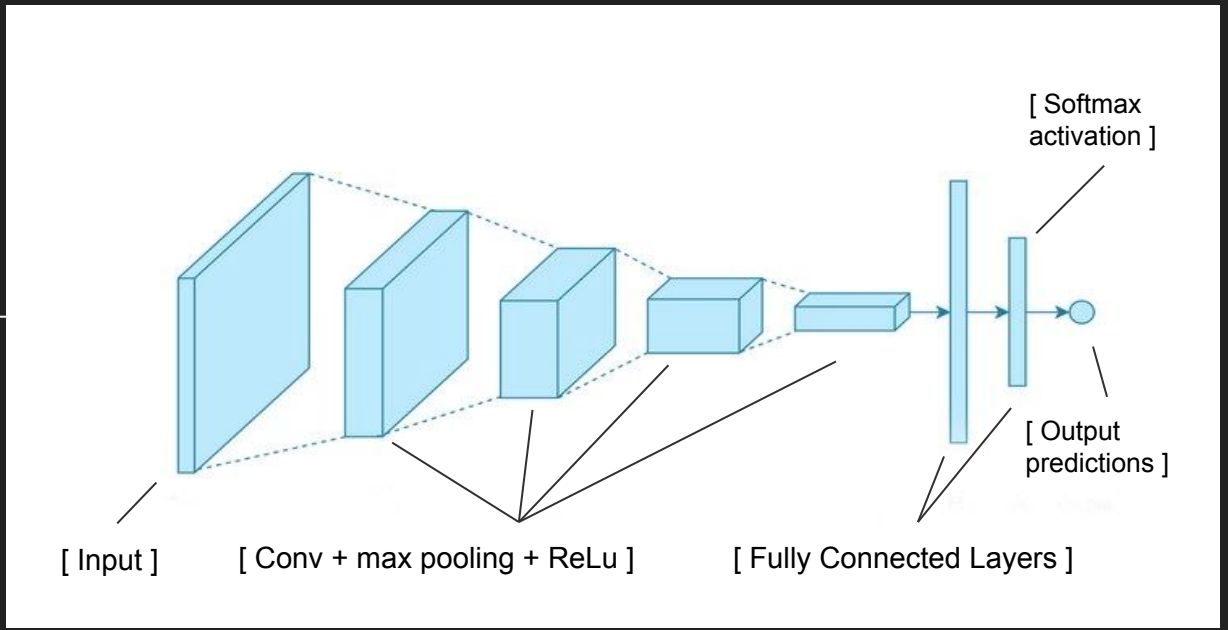
- For example: Words --/
  /--the
  /--how



- Images are then converted to grayscale, and reshaped to (64, 64, 1)
  - 1 representing color channel (grayscale = 1, RGB = 3)

ConvNet Architecture Overview

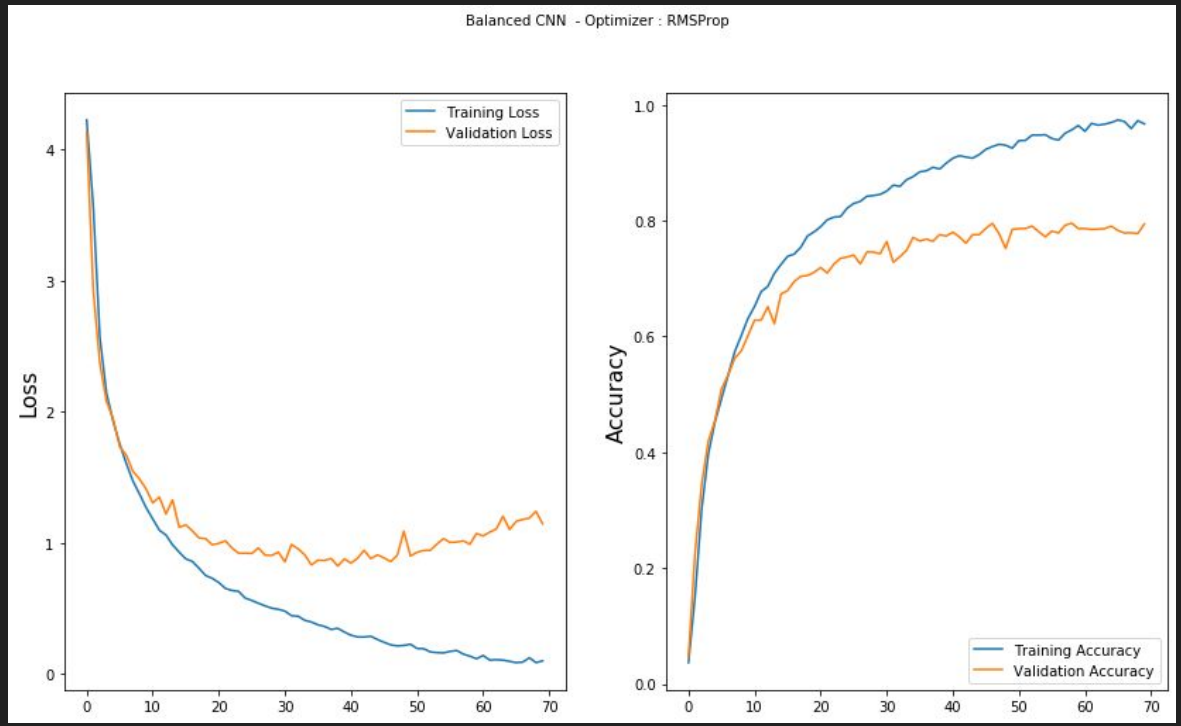




- In each convolutional layer, Convolution function, Max Pooling, and ReLu activation occur

Model Results

- 4 layers conv/pooling/ReLu
- Epochs: 70
- Batch Size: 341
- Optimizer: Adam
- Train Accuracy : 0.9669
- Validation Accuracy: 0.7943
- Test acc: 0.7872



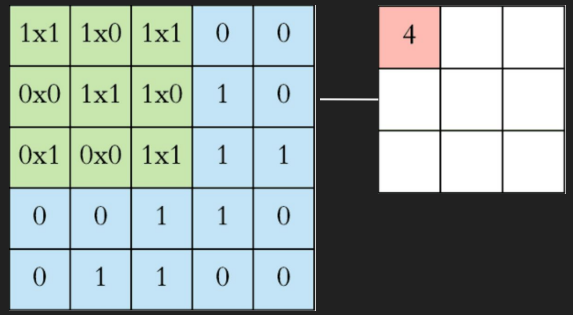
Conclusions

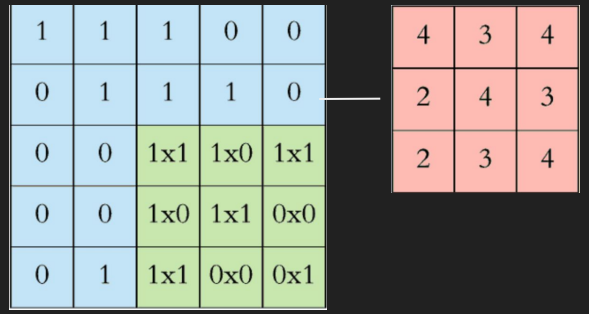
- Recognizing handwritten text image is no simple task.
- Although I do not feel optimal accuracy has been achieved, 78% is still progress considering the natural variations and similarities in handwriting.
- Future Work:
  - Implement Transfer learning with pre-trained models.
  - Integrate multi-label/multi-class approach for classification at character level.
  - Develop thresholding coordinates and bounding boxes.

A Closer Look

Convolution:

Filter(3x3) Slides over the input(5x5) taking the dot product at each stride. The sum is then added to appropriate position in feature map





Pooling:

Take the maximum value for each window. This decreases feature map while retaining important values.