HANDWRITING RECOGNITION

By: Ashley Jacob and Brendan Barnett

PROBLEM

- Handwriting recognition
- Initial Task
    - Classify an image of any given capital letter
        - 26 classes
        - ~3.8% accuracy if guessed at random
    - Originally aimed for 80% accuracy
- End Goal
    - Analyze an image of handwriting and return a string with the correct text

THE SLY FOX JUMPED OVER THE LAZY DOG

DIVISION OF WORK

- Model Creation: Brendan, Ashley
- Dataset Analysis: Brendan
- Image Preprocessing: Ashley, Brendan
- Letter Isolation: Ashley

DATASETS

- Initial dataset was too small and resulted in overfitting
  - Only 55 samples per letter
  - Data augmentation
- New dataset had tens of thousands of images
  - 28 x 28
  - More data improved accuracy

LITERATURE REVIEW (CNN)

- "CNNs are primarily used in the field of pattern recognition within images, allowing for the encoding of image-specific features into the architecture, making the network more suited for image-focused tasks."
    - An Introduction to Convolutional Neural Networks, source

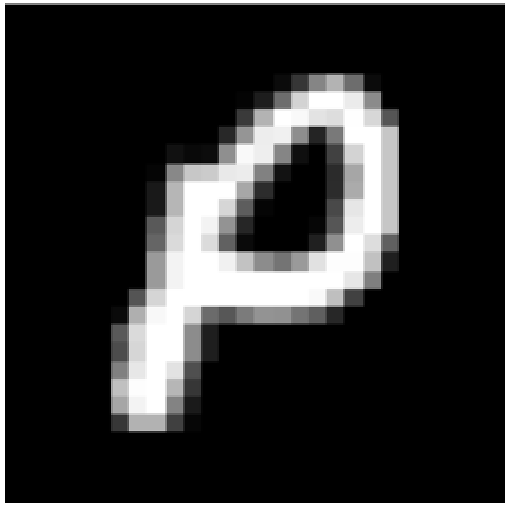
LITERATURE REVIEW (POOLING)

- "Pooling is a key-step in convolutional based systems that reduces the dimensionality of the feature maps. It combines a set of values into a smaller number of values, i.e., the reduction in the dimensionality of the feature map."
  - Pooling Methods in Deep Neural Networks, source

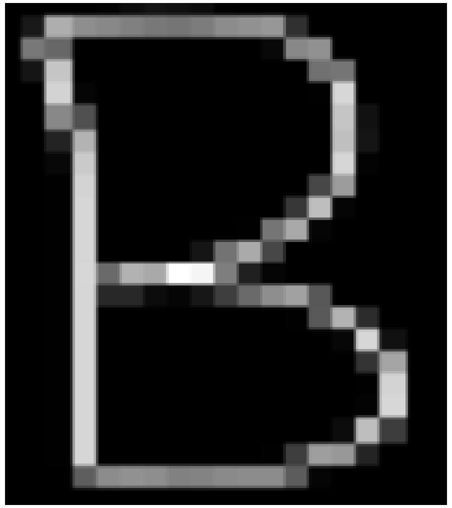
INITIAL ATTEMPT (MODEL)

- 1st, we overfit...
  - Too many epochs, no dropout layers
- Then underfit
  - Too much dropout, batch size too big
- And poor preprocessing

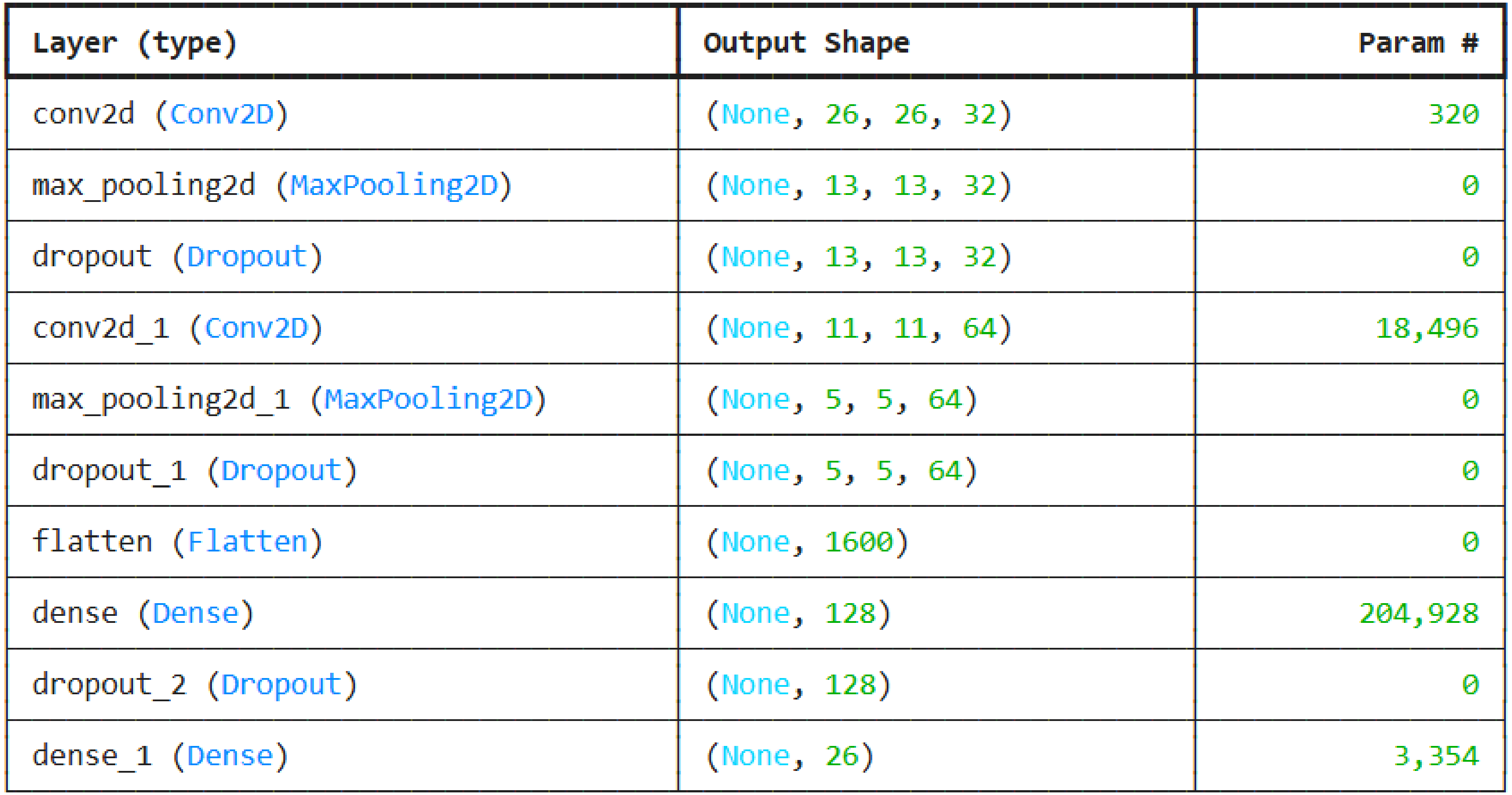
Expected



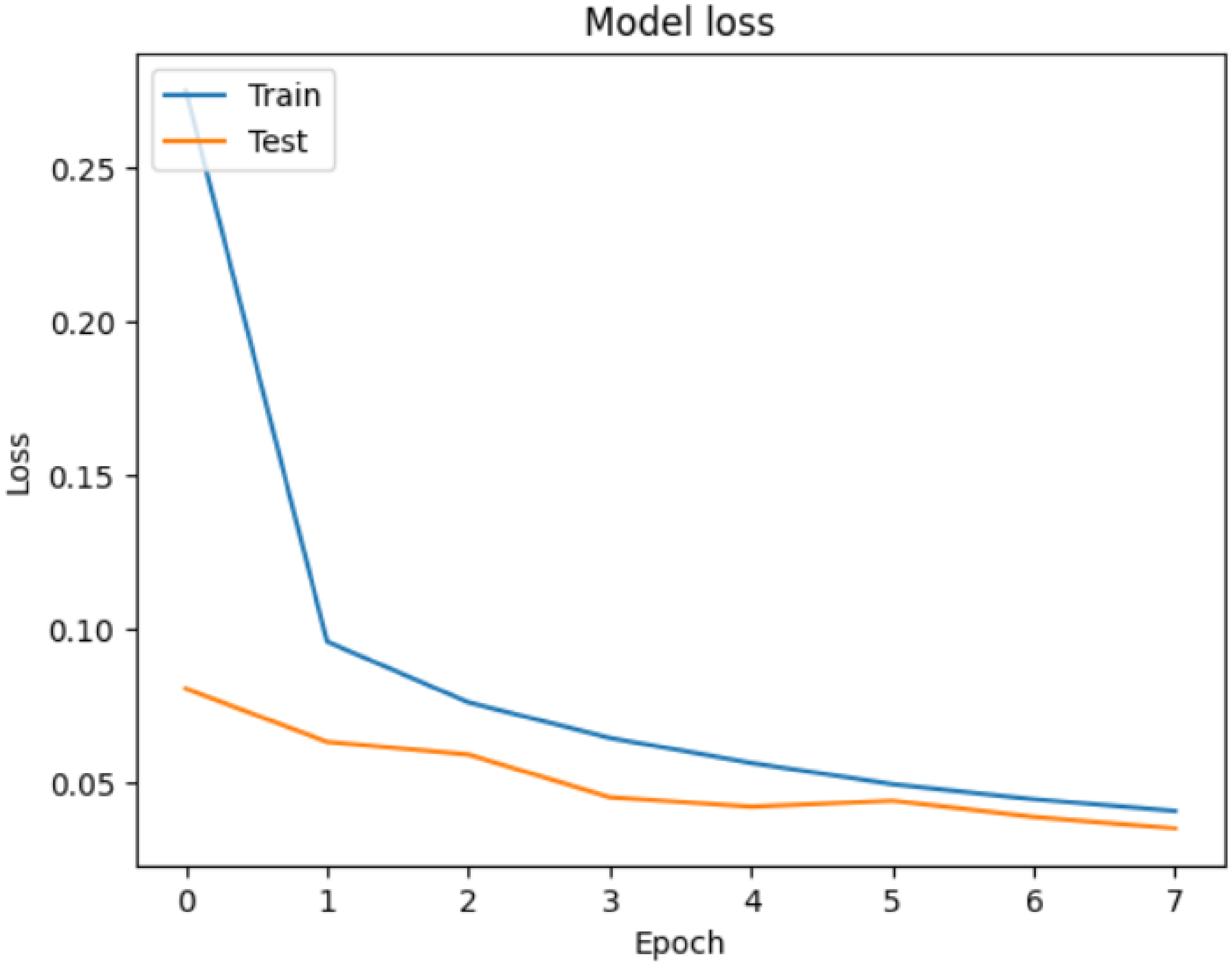
Reality (thin,
no padding)



FINAL MODEL ARCHITECTURE



Layer (type)	Output Shape	Param #
conv2d (Conv2D)	(None, 26, 26, 32)	320
max_pooling2d (MaxPooling2D)	(None, 13, 13, 32)	0
dropout (Dropout)	(None, 13, 13, 32)	0
conv2d_1 (Conv2D)	(None, 11, 11, 64)	18,496
max_pooling2d_1 (MaxPooling2D)	(None, 5, 5, 64)	0
dropout_1 (Dropout)	(None, 5, 5, 64)	0
flatten (Flatten)	(None, 1600)	0
dense (Dense)	(None, 128)	204,928
dropout_2 (Dropout)	(None, 128)	0
dense_1 (Dense)	(None, 26)	3,354

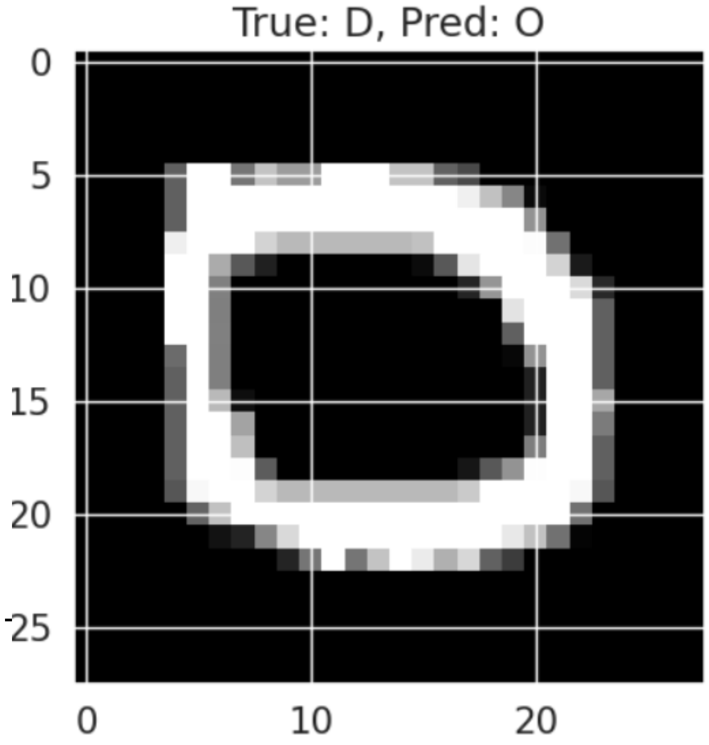

Model loss
Train
Test
Loss
0.25
0.20
0.15
0.10
0.05
Epoch
0
1
2
3
4
5
6
7

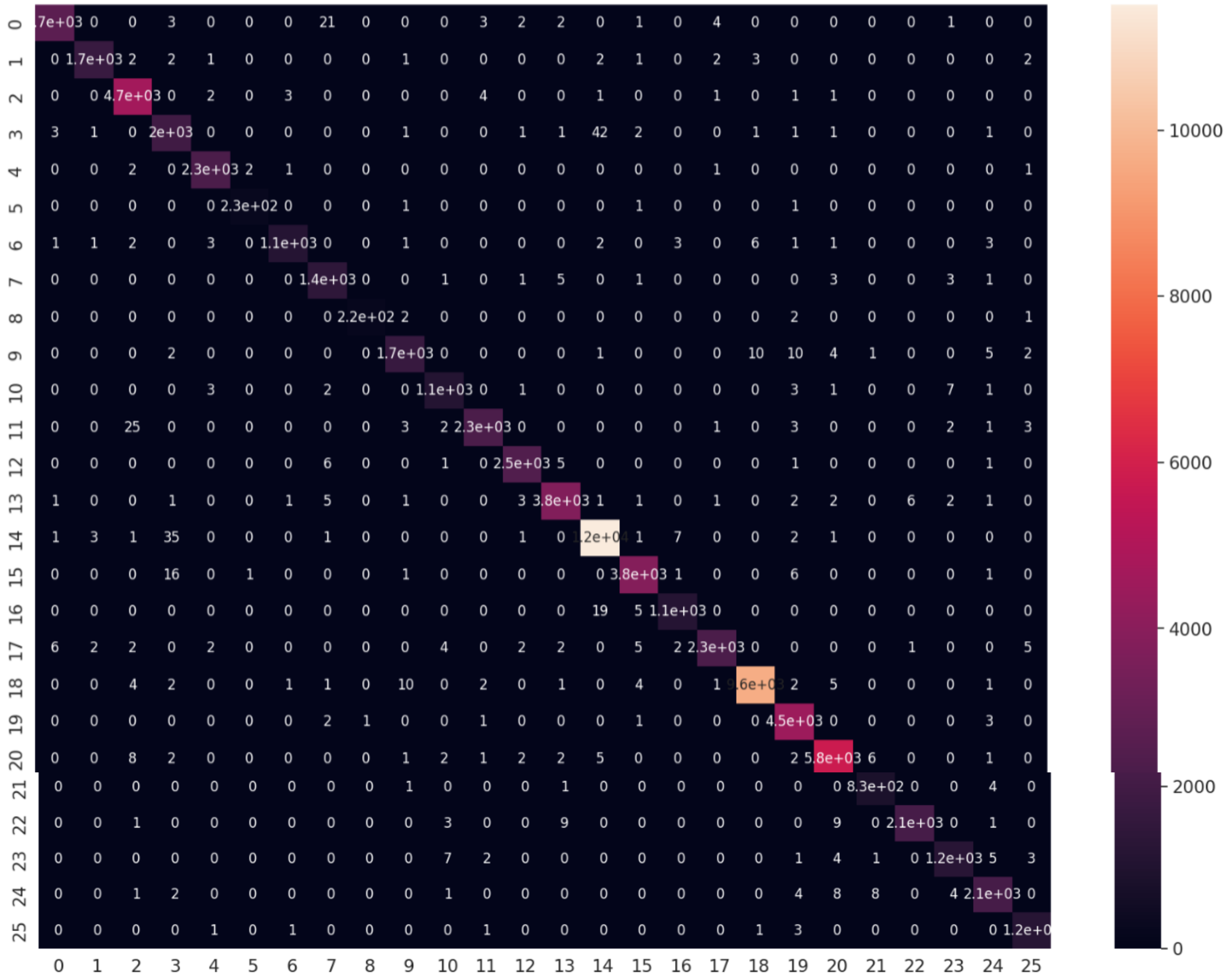
MODEL RESULTS (TEST SET)

20% holdout for testing:

- Accuracy: 0.9918512701
- Precision: 0.99186832351
- Recall: 0.99185125520
- F1-Score: 0.99185087003

```
True Label: D, Predicted Label: O, Count: 42
True Label: O, Predicted Label: D, Count: 35
True Label: L, Predicted Label: C, Count: 25
True Label: A, Predicted Label: H, Count: 21
True Label: Q, Predicted Label: O, Count: 19
```

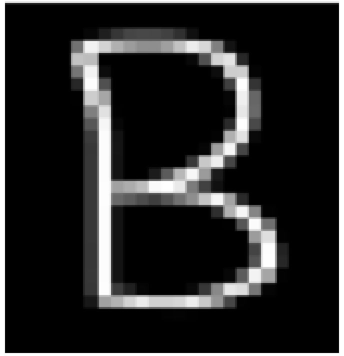


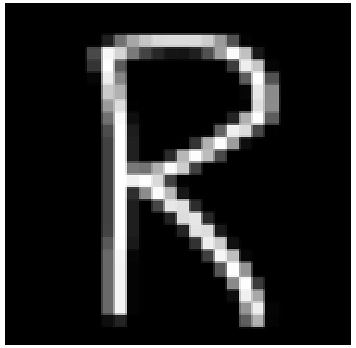


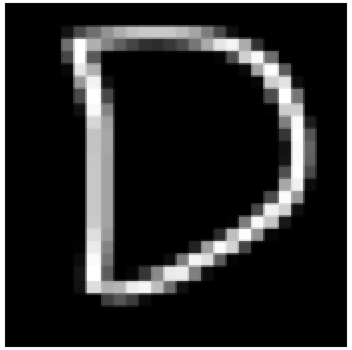
	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25
0	7e+03	0	0	3	0	0	0	21	0	0	0	3	2	2	0	1	0	4	0	0	0	0	0	1	0	0
1	0	1.7e+03	2	2	1	0	0	0	0	1	0	0	0	0	2	1	0	2	3	0	0	0	0	0	0	2
2	0	0	4.7e+03	0	2	0	3	0	0	0	0	4	0	0	1	0	0	1	0	1	1	0	0	0	0	0
3	3	1	0	2e+03	0	0	0	0	0	1	0	0	1	1	42	2	0	0	1	1	1	0	0	0	1	0
4	0	0	2	0	2.3e+03	2	1	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	1
5	0	0	0	0	0	2.3e+02	0	0	0	1	0	0	0	0	0	1	0	0	0	1	0	0	0	0	0	0
6	1	1	2	0	3	0	1.1e+03	0	0	1	0	0	0	0	2	0	3	0	6	1	1	0	0	0	3	0
7	0	0	0	0	0	0	0	1.4e+03	0	0	1	0	1	5	0	1	0	0	0	0	3	0	0	3	1	0
8	0	0	0	0	0	0	0	0	2.2e+02	2	0	0	0	0	0	0	0	0	0	2	0	0	0	0	0	1
9	0	0	0	2	0	0	0	0	0	1.7e+03	0	0	0	0	1	0	0	0	10	10	4	1	0	0	5	2
10	0	0	0	0	3	0	0	2	0	0	1.1e+03	0	1	0	0	0	0	0	0	3	1	0	0	7	1	0
11	0	0	25	0	0	0	0	0	0	3	2	2.3e+03	0	0	0	0	0	1	0	3	0	0	0	2	1	3
12	0	0	0	0	0	0	0	6	0	0	1	0	2.5e+03	5	0	0	0	0	0	1	0	0	0	0	1	0
13	1	0	0	1	0	0	1	5	0	1	0	0	3	3.8e+03	1	1	0	1	0	2	2	0	6	2	1	0
14	1	3	1	35	0	0	0	1	0	0	0	0	1	0	1.2e+04	1	7	0	0	2	1	0	0	0	0	0
15	0	0	0	16	0	1	0	0	0	1	0	0	0	0	0	3.8e+03	1	0	0	6	0	0	0	0	1	0
16	0	0	0	0	0	0	0	0	0	0	0	0	0	0	19	5	1.1e+03	0	0	0	0	0	0	0	0	0
17	6	2	2	0	2	0	0	0	0	0	4	0	2	2	0	5	2	2.3e+03	0	0	0	0	1	0	0	5
18	0	0	4	2	0	0	1	1	0	10	0	2	0	1	0	4	0	1	9.6e+03	2	5	0	0	0	1	0
19	0	0	0	0	0	0	0	2	1	0	0	1	0	0	0	1	0	0	0	4.5e+03	0	0	0	0	3	0
20	0	0	8	2	0	0	0	0	0	1	2	1	2	2	5	0	0	0	0	2	5.8e+03	6	0	0	1	0
21	0	0	0	0	0	0	0	0	0	1	0	0	0	1	0	0	0	0	0	0	0	8.3e+02	0	0	4	0
22	0	0	1	0	0	0	0	0	0	0	3	0	0	9	0	0	0	0	0	0	9	0	2.1e+03	0	1	0
23	0	0	0	0	0	0	0	0	0	0	7	2	0	0	0	0	0	0	0	1	4	1	0	1.2e+03	5	3
24	0	0	1	2	0	0	0	0	0	0	1	0	0	0	0	0	0	0	0	4	8	8	0	4	2.1e+03	0
25	0	0	0	0	1	0	1	0	0	0	0	1	0	0	0	0	0	0	1	3	0	0	0	0	0	1.2e+0

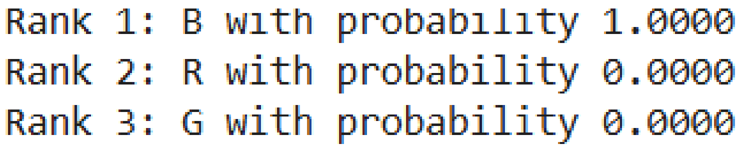
MODEL RESULTS (CUSTOM TESTS)

- Using commonly mixed-up letters with my own handwriting (rigid).
- Probabilities from softmax output activation.

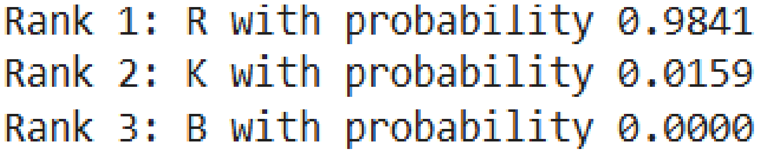




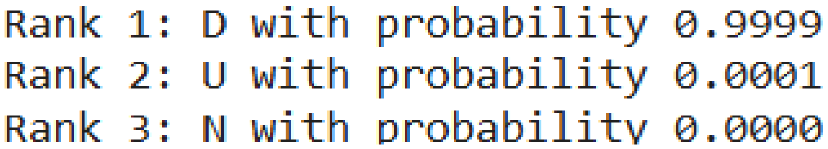




```
Rank 1: B with probability 1.0000
Rank 2: R with probability 0.0000
Rank 3: G with probability 0.0000
```



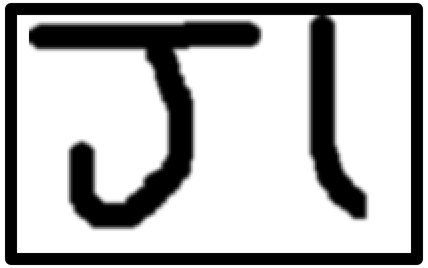
```
Rank 1: R with probability 0.9841
Rank 2: K with probability 0.0159
Rank 3: B with probability 0.0000
```

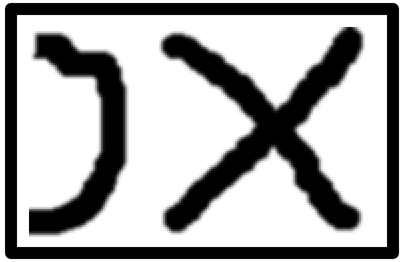


```
Rank 1: D with probability 0.9999
Rank 2: U with probability 0.0001
Rank 3: N with probability 0.0000
```

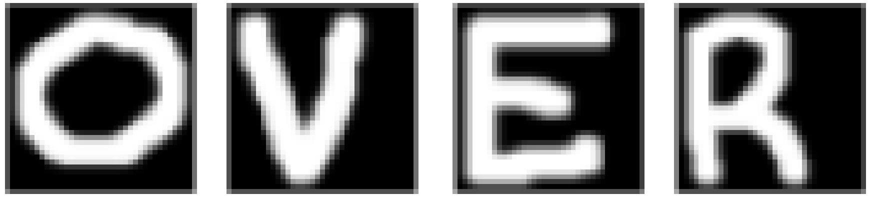
INITIAL ATTEMPTS (LETTER ISOLATION)

- Used K-means for clustering
  - Grouped pixels inaccurately
  - Inconsistent results





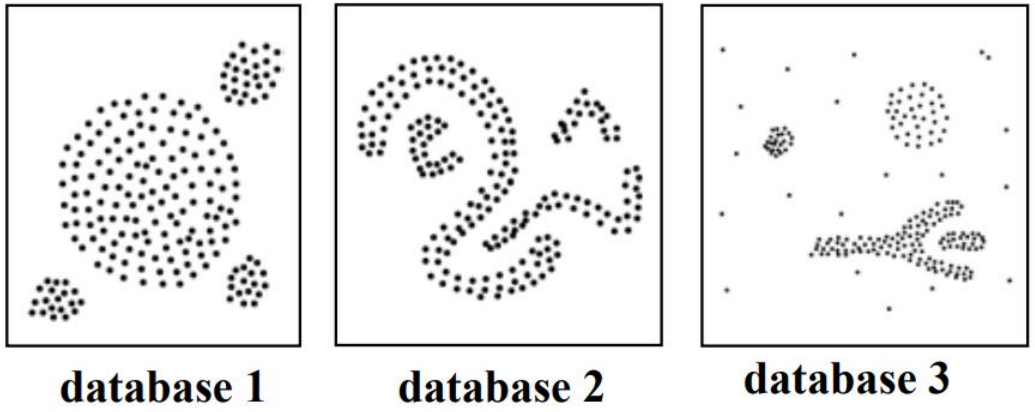
- Issues with padding and centering



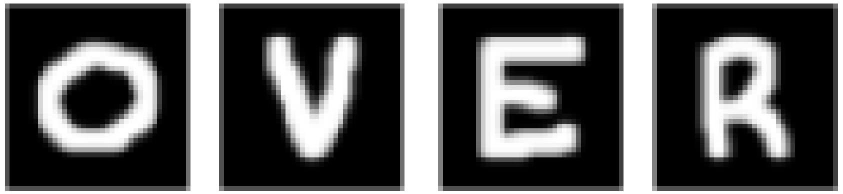
```
CVEL
```

LITERATURE REVIEW (DBSCAN)

- "...DBSCAN is significantly more effective in discovering clusters of arbitrary shape than the well-known algorithm..."
- A Density-Based Algorithm for Discovering Clusters in Large Spatial Databases with Noise, Source



LETTER ISOLATION

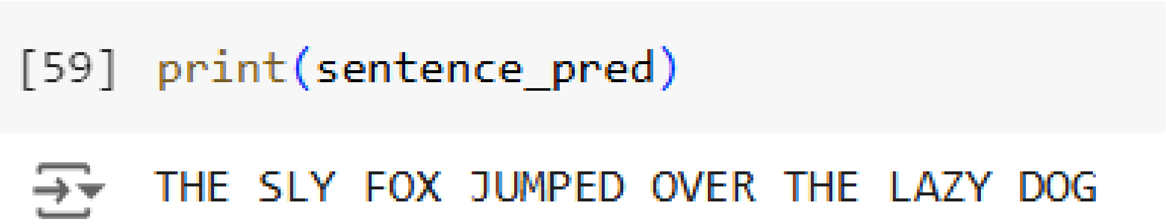


OVER

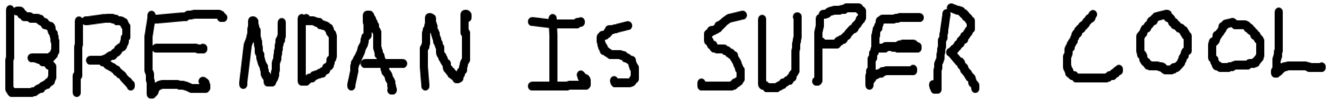
- Eventually chose DBSCAN
  - Always returns the same clusters
  - Two layers
    - One isolates words, other isolates letters
    - Sorted the cluster centers to figure out order
  - Take points from each cluster and create a new image
    - Make sure to pad and center letters to match dataset

MODEL RESULTS (CUSTOM SAMPLES)

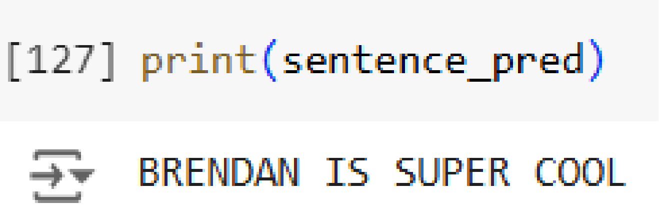
- When writing neatly, the model seems to predict the sentence very well



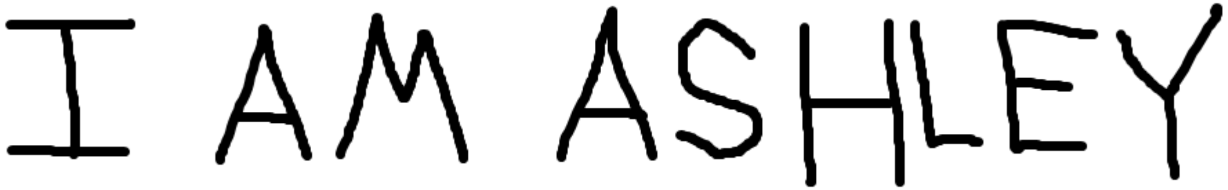
```
[59] print(sentence_pred)

THE SLY FOX JUMPED OVER THE LAZY DOG
```



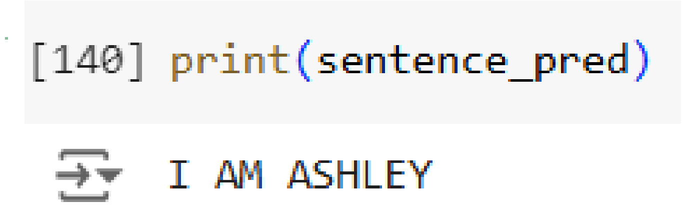
BRENDAN IS SUPER COOL



```
[127] print(sentence_pred)

BRENDAN IS SUPER COOL
```



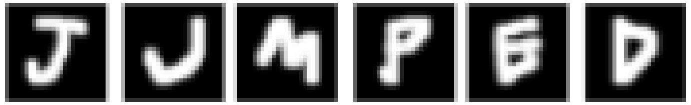
I AM ASHLEY



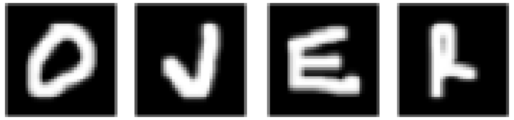
```
[140] print(sentence_pred)

I AM ASHLEY
```

ERRORS AND SHORTCOMINGS

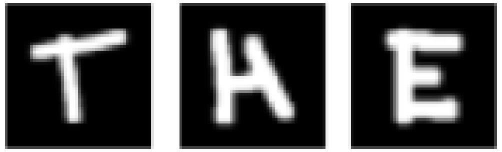
- Letters cannot touch
- Letters cannot have gaps
- Assumes certain distance between letters and words
- Some letters look very similar



```
JJMPED
```



```
OJER
```



```
TAE
```

THE SLY FOX JUMPED OVER THE LAZY DOG