

Capstone 3 Project Proposal

Digit Recognition using Deep Learning

1. Problem: Train neural networks (NNs), including convolutional ones, for recognizing handwritten images. Enhance the performance of NN models by allowing a user to provide feedback on classification results.

2. Data Wrangling

- **Data Collection:** upload from MNIST database, `from keras.datasets import mnist`.
- **Data Organization:** determine the data format and shape required by conventional and Convolutional Neural Networks, and reshape the dataset accordingly. Create folder tree for input and output data, data for model enhancement, and results.
- **Data Definition:** write helper functions to display and convert between numerical and image file representations of the digits.
- **Data Cleaning:** not expected, as there should be no missing or incorrect values or duplicates.

3. Coding of GUI window with Tkinter Python library

- This graphical interface allows handwriting an image.
- Display the result of its classification as one of the digits, and its predicted probability.
- In case if the digit was misclassified, a user can specify a correct label, save the misclassified image and its label for subsequent model re-training.
- Keep track of the actual prediction accuracy for digits handwritten in the GUI.

4. Coding integration of manually labeled data with MNIST.

Additional training data generated by the user in the GUI will be merged with MNIST training data.

5. Train usual NN and convolutional CNN models, test GUI.

Use Keras, and then optimize performance with TensorFlow.

6. Collect performance metrics

- AUC and accuracy vs training epoch for train and validation datasets for different NN architectures.
- Training time vs number of nodes and layers.
- Actual accuracy running average for different NN architectures.
- Study the actual accuracy of the handwritten digits.
- All of the above for MNIST only and for MNIST + manually generated training data.

7. Summary statistics and hypothesis testing of the difference between MNIST and combined datasets.

8. Visualizations of main findings and **storytelling** regarding the improvement of accuracy of the digit classifier using its Tkinter GUI. See if performance significantly improves after providing sufficiently many manually labeled handwritten digit images.

9. Slides

Problem identification (1-2 slides)

Recommendation and key findings (1 slide)

Modeling results and analysis (3-4 slides)

Summary and conclusion (1 slide)