**Handwriting detection**

1. **Beyond State-of-the Art (expected contributions)**

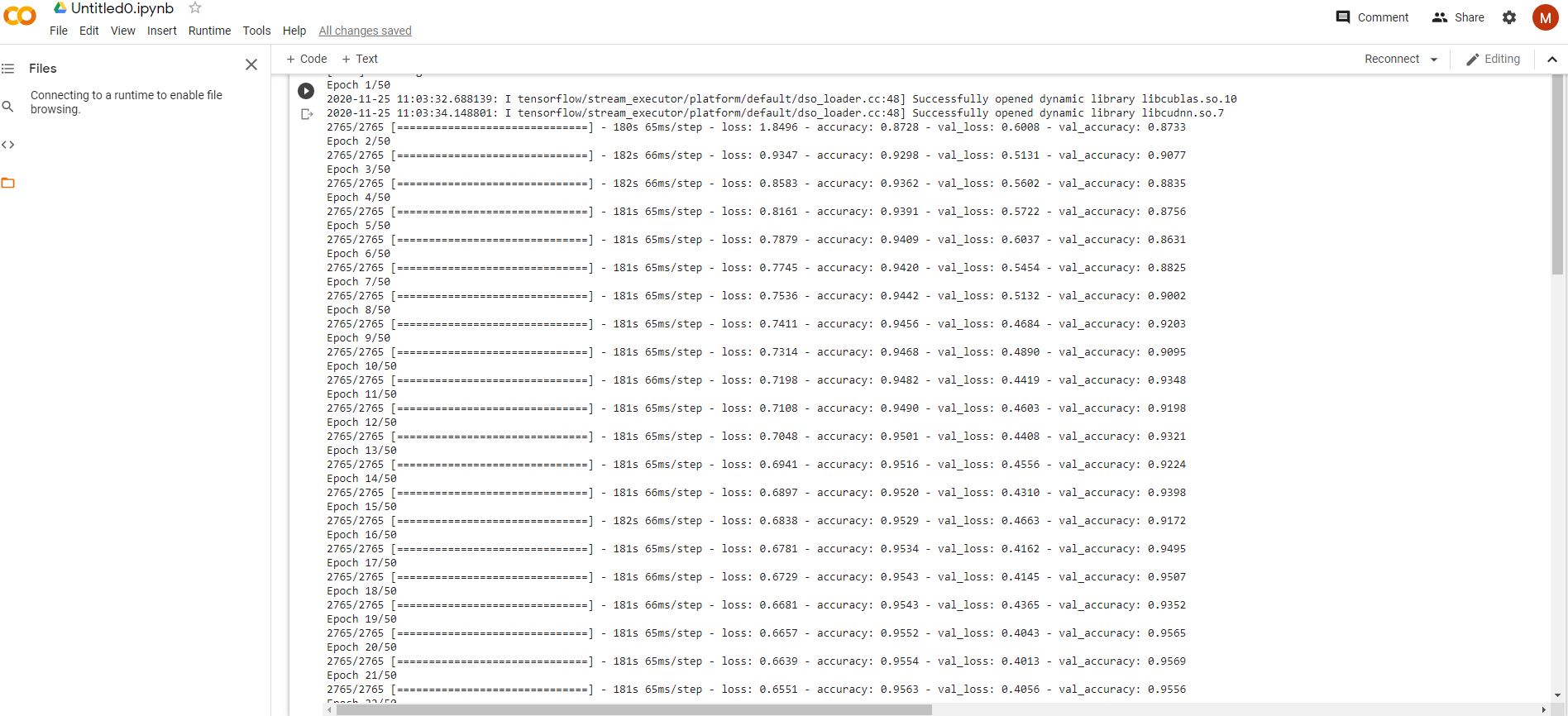
In the case of our project as a state of the art, we turned to two implementation solutions. The first solution was based on Markov models and the second is based on the use of neural networks. After extensive research, we decided to focus more on artificial intelligence combined with OCR, because the Markov model involves difficult-to-apply mathematics. For beyond state of the art, the introduction of Romanian special characters is a plus and also maximizing the accuracy of the network is a starting point.

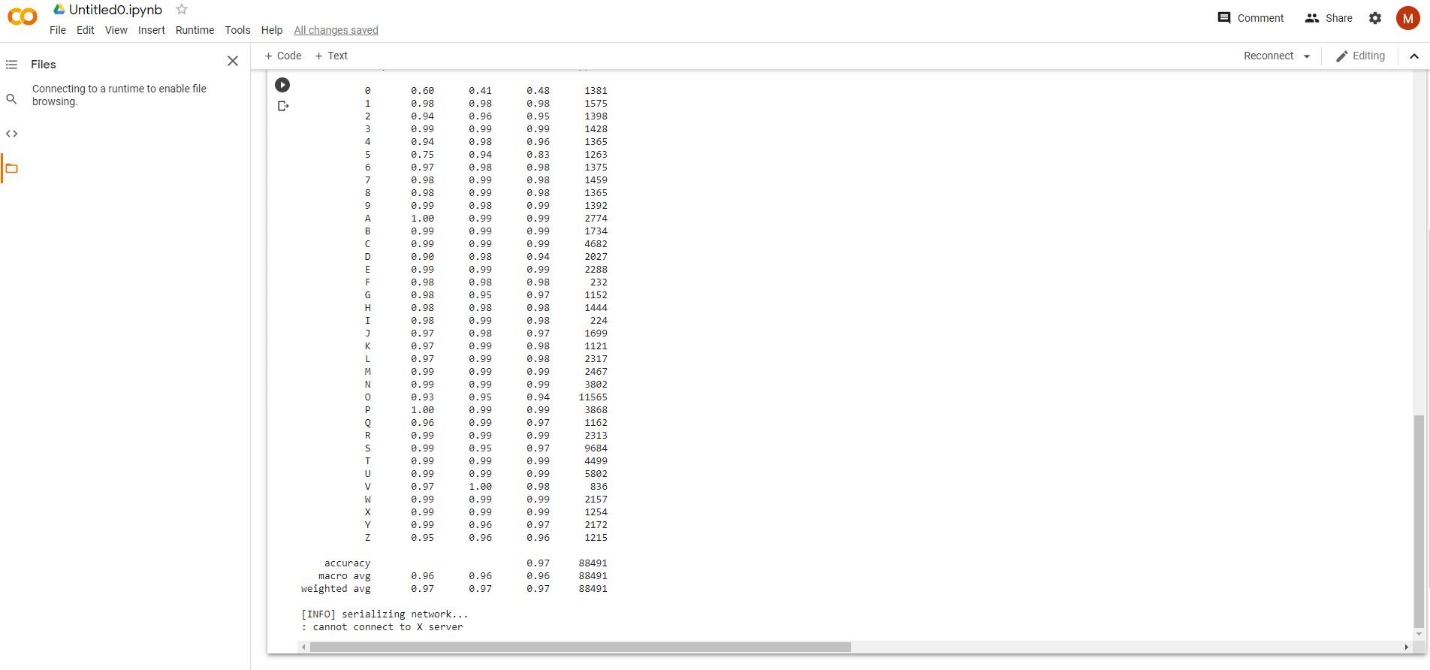
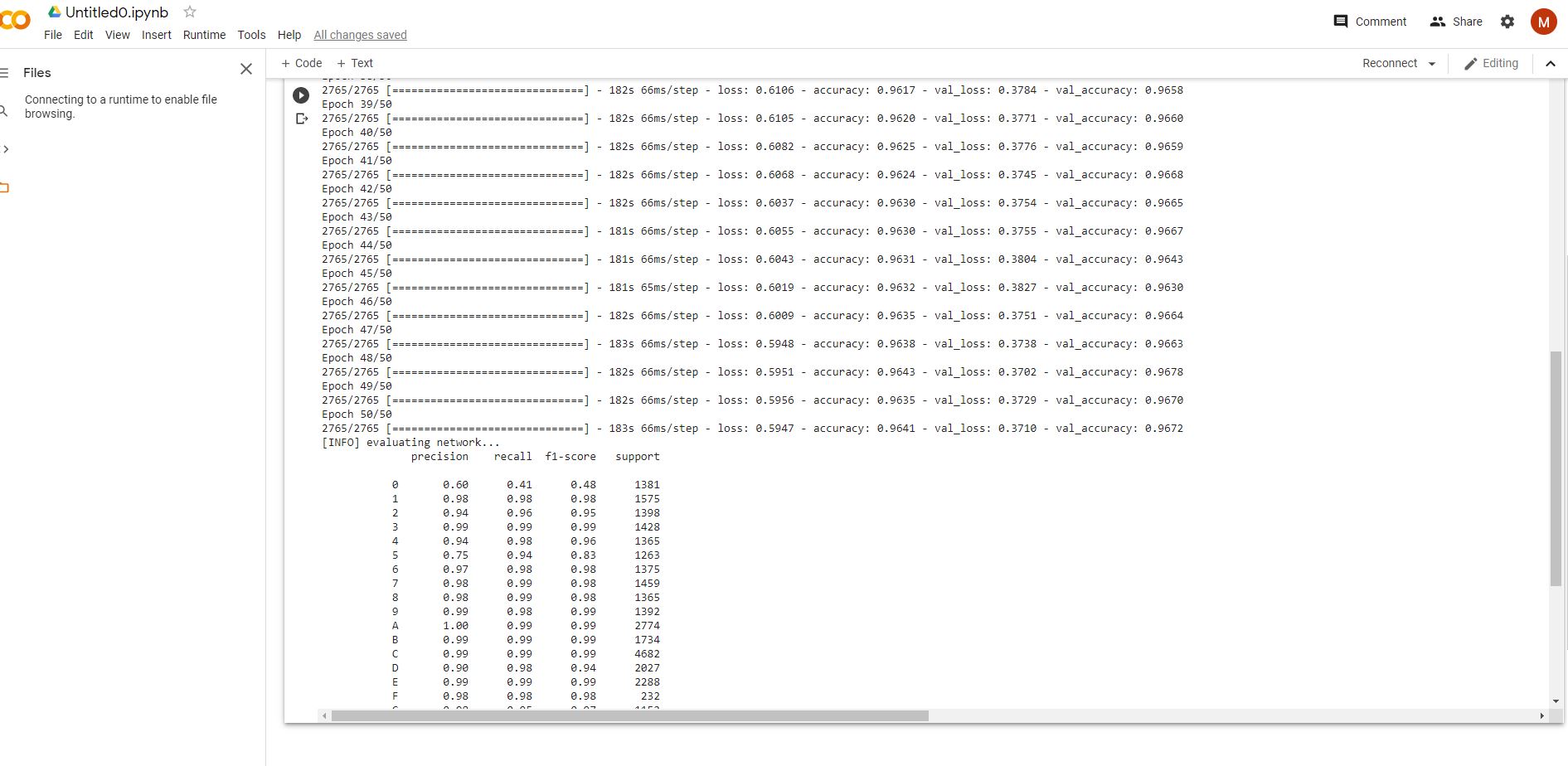
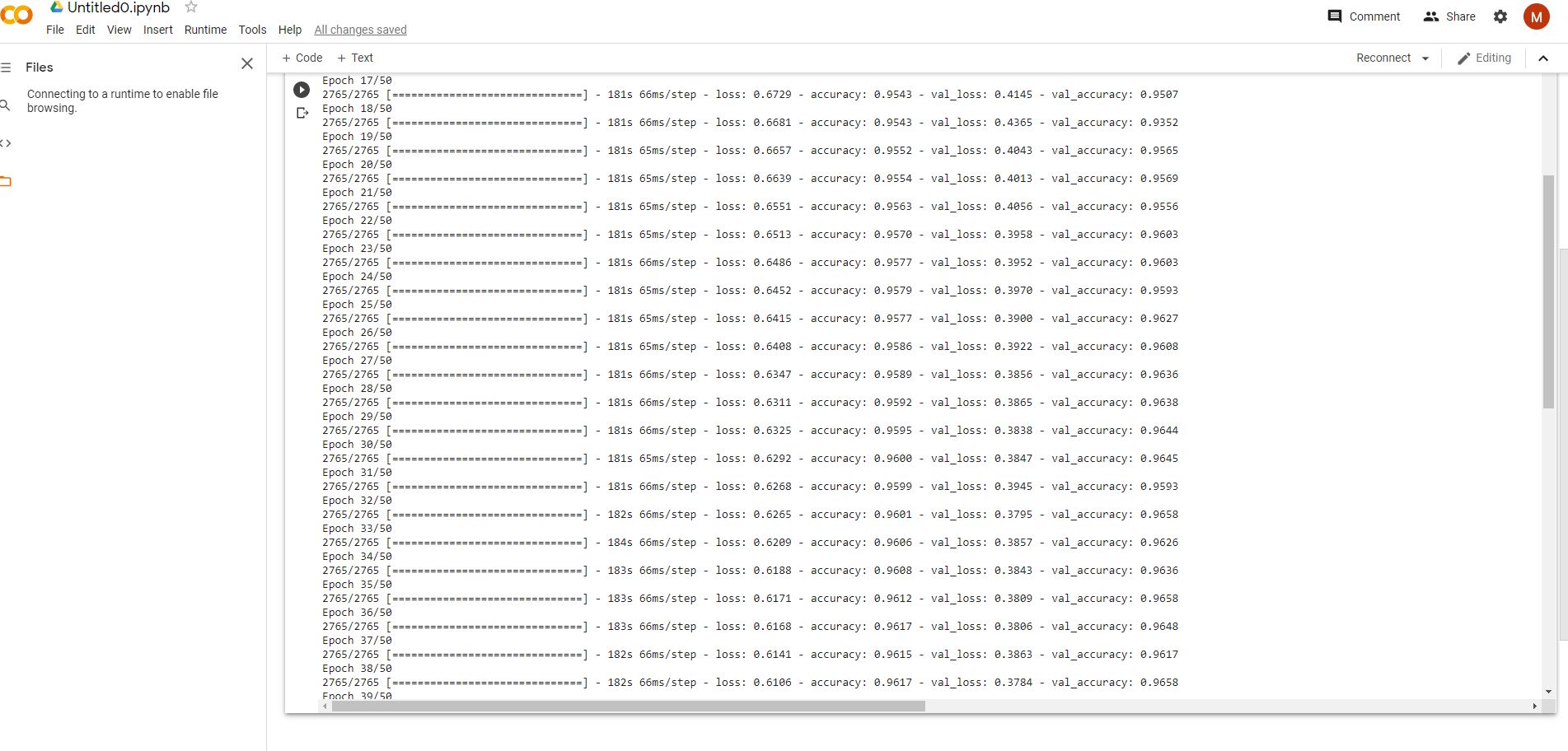
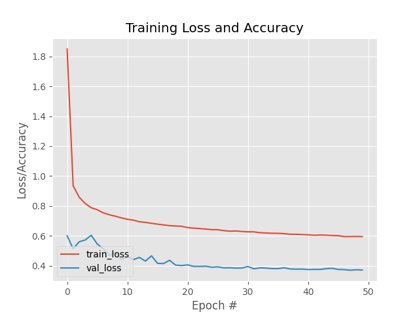
1. **Proof of concept demonstrator application**

As a concept, we intend to:

Load both the datasets for MNIST 0-9 digits and Kaggle A-Z letters from disk, combine these datasets together into a single, unified character dataset, we are then going to loop over each row of our CSV file and parse out the label and the associated image and casts it as a NumPy array of unsigned 8-bit integers, which correspond to the grayscale values for each pixel. We will have successfully train a Keras and TensorFlow model on the combined dataset using a custom implementation of the very popular and successful deep learning architecture, ResNet. Finally we will plot the results of the training and visualize the output of the validation data by saving the model to disk, plot the results of the training history, and save the training history.

1. **Preliminary results (using demonstrator application)**



1. **Preliminary conclusions**

After seeing the results, we foresee faster and better progress as we move forward to the next challenges. As it can be seen, the model averages around ~96% accuracy on the training set. Also, judging by the plot there are few signs of overfitting, implying that the model is performing well at our basic OCR task. For the second sprint, the project focused largely on training an OCR model with Keras, Tensorflow and Deeplearning to recognize handwritten characters. Training a model is very important in detecting the handwriting that we will extract in the next step. For model training we have two types of data sets for recognition: alphanumeric and alphabetic. The MNIST 0-9 standard was chosen for the alphanumeric set. The standard MNIST dataset is embedded in the popular deep learning framework, including Keras, TensorFlow and PyTorch. For letters we used the special NIST 19 database, which was resized to 28 x 28 pixels in grayscale to be in the same format as our MNIST data set.

1. **Update Bibliography:**

* [MNIST handwritten digit database, Yann LeCun, Corinna Cortes and Chris Burges](http://yann.lecun.com/exdb/mnist/)​
* <https://www.pyimagesearch.com/2019/07/08/keras-imagedatagenerator-and-data-augmentation/>​
* <https://www.pyimagesearch.com/deep-learning-computer-vision-python-book/>​
* <https://www.nist.gov/srd/nist-special-database-19/>​
* <https://www.tensorflow.org/>​
* <https://opencv.org/about/>