**Capstone Project System Administration Guide:**

**Using Machine Learning to Perform Handwriting Recognition on Mathematical Equations**

Brendan Turner

Grand Canyon University

February 16, 2022

**General Information**

The inspiration for this assignment comes from the growing popularity of online learning, whether by choice or compulsory due to the nature of the COVID-19 pandemic, and the subsequent need for modules to aid mathematical learning in place of in-person instruction. This project aims to identify and solve images of handwritten mathematical equations. Successful implementation of this project builds a model that will take any image of a mathematical equation, whether on paper or drawn on a digital sketchpad, and identify each of the individual characters contained in that image. It uses a classifier that will recognize what each character is, and turn these results into an equation string. This string will then be solved by a custom-built function, and the solution is displayed below the input image for the user.

**System Summary**

The program is written in Python 3.6, and the Jupyter Notebook was used to write and test the code. There are two main files: one for designing, running, and saving the image classifier, and a main.py file, which runs the GUI and performs the main functionality of the program. This entails gathering the input image, segmenting the characters of the image, running each segmented character through the classifier, and using the results to put together the equation string to be solved. The main program uses PySimpleGUI as the user interaction model, OpenCV and imutils to perform segmentation, and SymPy to perform the equation solver. The classifier’s libraries include NumPy and Pandas for data handling, Pillow for image extraction, and Keras through TensorFlow to build the model itself. All of these libraries must be installed prior to running the files. If they are not installed locally already, the program will download them for you automatically, but this will take a couple of minutes.

**Getting Started**

Everything that the user needs is displayed immediately upon loading the webpage. This application is very simple in its nature, as since it is designed for one purpose, it only has one user function. This is to click on the “Choose File” button to pick which image you’d like to upload (future versions may include a sketchpad to draw your equation in the app). The system will then decode the symbols in the image in the background, and will produce a text string showing your equation. For testing purposes, a second button exists to display a sample image to show the efficacy of the model.

**System Configuration**

Upon loading the program in a web browser, all the necessary libraries will load to run the model. If any of them are previously not installed (especially Python itself) then this will take some time to locally load. The system is configured for optimal runtime, and to maximize the ease at which the user goes through the program. The CNN is pre-made and ready to run, saved as an .h5 file. Should the user wish to rerun the classifier or modify it, this will require some time to run.

The image that you upload must be either a JPEG or PNG file. If you choose an incompatible image, or one that does not display a mathematical equation, you will get an error message and be returned to the main screen. The program is not configured to accept any other image types. This part of the model uses the Pillow library, which can use other image types, so the program may be modified if the user wishes to use something other than these two filetypes.

**System Maintenance**

To ensure that this program works as intended, it is necessary to have all the software requirements met. This includes having Python installed and all the necessary libraries available to load. They can be found at each of their respective locations below (in the Legal Notice section). Please note that building the CNN from scratch takes some time. The program was written on a Surface laptop with 8GB of RAM, so it will run on similar systems, but may be quicker on more powerful computers. The model is built with the desired speed and accuracy in mind, but these can be configured to the user’s preferred specifications if they so choose. Beware that changing any of the layers in the CNN, any steps in the cleaning process, or the number of chosen classes can potentially cause errors and/or substantially slow down the model or distort the accuracy of the character detection algorithm.

**Security**

The main benefit of the simplicity of this app is that beyond uploading an image, there is nothing else that the user needs to do. The program will analyze the image itself and produce a solution in the window beneath the image itself. This image that you upload must be either a JPEG or PNG file. If you choose an incompatible image, or one that does not display a mathematical equation, you will get an error message and be returned to the main screen.

**Legal Notice**

This project is the personal intellectual property of the developer. The program is written in Python 3.6 and can be accessed here: <https://www.python.org/downloads/> The data used to train the classifier is from the HASYv2 dataset, which can be found at its original location here: <https://arxiv.org/pdf/1701.08380.pdf>. All the libraries necessary to build the classifier and run the final product can be found at their respective locations here:

* + [NumPy](https://r.search.yahoo.com/_ylt=AwrEzedsBwJiatgAnpJXNyoA;_ylu=Y29sbwNiZjEEcG9zAzEEdnRpZAMEc2VjA3Ny/RV=2/RE=1644328941/RO=10/RU=https%3a%2f%2fnumpy.org%2finstall%2f/RK=2/RS=GTmV_EM34E4e4QUEuT8lC_pBZQc-)
  + [Pandas](https://r.search.yahoo.com/_ylt=AwrE1.EpCAJi4gkA4fZXNyoA;_ylu=Y29sbwNiZjEEcG9zAzEEdnRpZAMEc2VjA3Ny/RV=2/RE=1644329129/RO=10/RU=https%3a%2f%2fpandas.pydata.org%2fpandas-docs%2fversion%2f0.23%2finstall.html/RK=2/RS=jyaP3xY_ZZ3GVFVXc5_QX_lbCFQ-)
  + [TensorFlow](https://r.search.yahoo.com/_ylt=AwrJ7JRKCAJiSvwAOUJXNyoA;_ylu=Y29sbwNiZjEEcG9zAzEEdnRpZAMEc2VjA3Ny/RV=2/RE=1644329163/RO=10/RU=https%3a%2f%2fwww.tensorflow.org%2finstall%2f/RK=2/RS=OMLOMq5IVoYKjXey4CY0ZTd692w-)
  + [MatPlotLib](https://r.search.yahoo.com/_ylt=AwrE19VrCAJiaSEADhZXNyoA;_ylu=Y29sbwNiZjEEcG9zAzEEdnRpZAMEc2VjA3Ny/RV=2/RE=1644329195/RO=10/RU=https%3a%2f%2fmatplotlib.org%2fstable%2fusers%2finstalling%2findex.html/RK=2/RS=aXwIk4qCvcdi3jl4ju0kVDuvGHc-)
  + [Pillow](https://r.search.yahoo.com/_ylt=A0geJaB4CAJia7AACJtXNyoA;_ylu=Y29sbwNiZjEEcG9zAzEEdnRpZAMEc2VjA3Ny/RV=2/RE=1644329208/RO=10/RU=https%3a%2f%2fpillow.readthedocs.io%2fen%2fstable%2finstallation.html/RK=2/RS=X05ugeEQ08A0_tJ0Ug9x4HUIdSM-)
  + [SymPy](https://r.search.yahoo.com/_ylt=A0geJaGDCAJianEAXhZXNyoA;_ylu=Y29sbwNiZjEEcG9zAzEEdnRpZAMEc2VjA3Ny/RV=2/RE=1644329219/RO=10/RU=https%3a%2f%2fdocs.sympy.org%2flatest%2finstall.html/RK=2/RS=fkS2M5VAIqPoZjTU2U2vTaF7pBo-)

**Help and Contact Details**

If you have any further questions about this product, the developer can be reached at:  
 [brendanrturner1@gmail.com](mailto:brendanrturner1@gmail.com) or by phone at 908-665-8056.