Enhancement Two: Algorithms and Data Structure

Tuyet Tran

Prof. Joseph Conlan

Southern New Hampshire University

November 21, 2024

This week I’ve been focused on enhancing my handwriting recognition artifact by transitioning from the MNIST dataset to the IAM dataset. This change has been a significant upgrade, as it allows me to work with a larger and more complex dataset. While making this transition, I discovered several useful resources that provide guidance on using the IAM dataset for handwriting training. The file I’m currently working with was created in 2022 and has been recently updated, which ensures I’m working with relevant and up-to-date tools.

I chose handwriting recognition as my artifact because I wanted to explore the scalability of working with larger datasets. Through this week’s project, I’ve evolved the artifact from recognizing simple digits to recognizing full characters and even constructing words. To achieve this, I’ve incorporated various processes, including splitting the dataset into training and validation sets, string splitting, reading text files, and using ground truth text to train the model. I also had to modify the original code to adapt to the IAM dataset by replacing modules, tweaking parameters and sizes, and simplifying functions to better suit my project needs. The main focus this week has been transitioning from the MNIST dataset to this more advanced and comprehensive dataset. The basic structure of handwriting recognition involves several key steps. First, the database is imported into the program. Next, the images are adjusted to properly fit the model and split into training and validation samples. A neural network is then created, and various attributes are implemented for the training process. After training is complete, the learned model is used to run the validation and assess its performance.

One of the course outcomes I’ve been working toward through this enhancement is designing and evaluating computing solutions to solve specific problems using algorithmic principles and computer science practices. This involves managing trade-offs in design choices while leveraging various resources. I feel I’ve met this goal, as the changes I’ve made ensure efficient implementation across different operating systems while optimizing memory allocation.

Initially, I had planned to host the program on a website, but I’m now considering creating a downloadable version that can run directly on a machine. This shift in direction is aimed at improving usability and accessibility.

Throughout this process, I’ve gained valuable insights into new algorithms and neural network structures, particularly recurrent neural networks (RNNs) and CTC. I also gained hands-on experience with modules like cv2, pickle, numpy, and pathlib. However, through my research, I found that using pickle could increase the program's complexity. The biggest challenge I faced was navigating the steep learning curve of new training techniques, which required significant time and effort to fully understand and implement effectively.

Overall, this week has been a valuable step in refining my artifact and expanding my technical skills.