Angel Candelas

Prof. P. McManus

ITAI 1378

July 5, 2024

L06 Chihuahua or Muffin Journal

In L06 Chihuahua or Muffin, the notebook walked me through the process of building an image classifier with PyTorch. The main objective was to teach a neural network to successfully tell the difference between pictures of chihuahuas and muffins. A neural network module from PyTorch was imported to define the network layers. Next, the torchvision library was used to load the data from the images in the folders. This library allowed us to perform preprocessing steps such as resizing, normalizing and tensor conversion. After, the notebook went through the training process where we could update the number of epochs and learning rate parameters for more accurate results. Finally, the notebook went through the validation process after each training epoch.

There were several key concepts that this notebook went through, and I learned about. The main concept was to build image classifiers using neural networks. I was able to learn about linear layers and activation functions that are used in processing image data that make up the core of a neural network. Another key concept of this notebook was to learn how to load and preprocess data. Additionally, the code showed how to organize datasets with the imported torchvision library from PyTorch. I think one of the key concepts that I found to be very important in this notebook was the training process. Being able to update the epochs and rates showed how you could achieve different results and reach proper optimization. The validation process was also an important key concept because it gives an estimate of how the machine will work on new data.

This lab was easier for me to follow than the previous one. As I have mentioned in previous journal entries, this is all brand new to me, but I am beginning to gain an understanding going through these a few times as well as following along with the class recordings. Whereas the previous lab presented lots of challenges for me, I was able to swiftly move through this one with minor challenges encountered that I was able to resolve quickly. The first error I encountered was in the spacing of the code when attempting to clone the notebook. Having watched the class recording prior to beginning the lab, I was expecting this and knew how to resolve it immediately. Most of my other “challenges” came from me not reading the code before trying to run it. For example, when building the neural network, I did not put in height and width values, so I received an error when I tried to run the code. I started with a small number (8) because I thought this would help with “keeping things simple.” I changed this value eventually, but that was after I had gone through the notebook completely. Another example of an error that I came across was when I attempted to run the code to initialize the network. The model was not defined, and I was not entirely sure how to fix this one, so I had to ask Gemini for assistance. The fix was rather simple, and I added in the the MySkynet line to get the code to run properly. Next up, yet another case of me not reading the code before trying to run it. This time, I left the batch size as a question mark, so I received an error. I changed the batch size to 32 because I

read that a good batch size is usually a power of 2 like 16, 32 and 64. Overall, there were not many challenges in following this lab and I was very happy that I was able to follow along without getting completely lost.

I was able to gain a lot of insight about machine learning and image classification. Honestly, just about everything that I have read throughout this course has been brand new information for me (I know I keep saying that, but it helps keep me motivated) but what I learned in this lab has been the most interesting for me, so far. Using neural networks for image classification is a very strong tool for learning patterns in images to be able to tell differences and appropriately classify an image (chihuahua and muffin). I also gained insight into the importance of data preprocessing. With preprocessing, you can change different parameters in the image date to make the training process better. Insight was also gained on the training and validation processes. Seeing how the data is split into training and validation sets makes it so that we can see the network's ability to differentiate between images.

I think that there can be so many ways that these techniques can be used in the real world. The first that comes to mind is with self-driving card. I feel like image classification for self-driving cars is crucial to recognize things like other cars, pedestrians and all the different street and traffic signs. Another way that these techniques can be used in the real world is with surveillance or security systems. Using image classification in this way can help by using facial recognition or being able to alert when there are out of place objects in an image. Image classification techniques can also be used in the medical field with things like x-rays or different kinds of body scans.

I kind of mentioned a few of my personal reflections on my learning experience in this lab throughout this journal, but to sum up and reiterate, this experience was a good one. This lab really piqued my interest in machine learning and image classification. I even went through the lab a few additional times changing the parameters to get different results each time. When it came to the height and width, I started with 8 on both and then also tried 16 and 32 on each in subsequent runs. I also updated the batch sizes to 16 and 32 and the epochs from 3 to 16 and then 64. With each increase in parameters my results became better and better. One image of a chihuahua went from being 42% chihuahua and 48% muffin in the first run to 99% chihuahua and 1% muffin by my final run. Hoping to keep up this momentum with the next lab!