Cassy Prescott-Cormier

Computer Vision

Lab 06 - Chihuahua or Muffin Workshop Instructions

Professor Anna Rachapudi

March 28th, 2025

Reflective Journal: Chihuahua or Muffin Classifier

Sometimes, the most interesting computer vision tasks are the ones that seem simple. In this lab, I'll be training a basic neural network to solve a playful yet educational problem: categorizing images of chihuahuas and muffins. As humans we can easily tell the difference, but this task is a perfect example of how machines need to learn from data and patterns, especially in image classification. By the end of this lab, we'll see how neural networks can learn to distinguish between two objects using real-world data and fundamental machine learning techniques.

Reflecting on this project, I realize how much there is to learn about neural networks and computer vision. I'm filled with a sense of curiosity and excitement as I dive deeper into deep learning, especially exploring convolutional neural networks, image classification, and transfer learning. There were some reservations initially when I saw the complexity of the code. As I start reading through the code, I am excited to work on my coding skills by reading the provided code.

This report covers various topics, such as convolutional neural networks and transfer learning. A neural network is a subset of machine learning that involves deep learning among multiple layers of nodes. Convolutional neural networks (CNNs) are vital for computer vision tasks such as classification. CNNs contain a convolutional layer, a pooling layer, and a fully connected (FC) layer (IBM, n.d.). Transfer learning refers to the ML technique of re-training a pre-trained model for a new task. It's effective for enhanced efficiency, performance, and accessibility (What is Transfer Learning?, n.d.).

The training accuracy remains constant at 96.67% in all three epochs. It suggests that the model fits the training data well and is likely to learn the patterns of the training set. The validation accuracy starts at 93.33% in epoch 1, drops to 90% in epoch 2, and then fluctuates back to 93.33% in epoch three before falling back to 90% in epoch 4. The accuracy does not show consistent improvement, which indicates issues with overfitting. The training error steadily decreased from 0.5311 in epoch 1 to 0.4941 in epoch 4. The validation error decreases from 0.5652 in epoch 1 to 0.5454 in epoch 4 but decreases only slightly. The model seems to be overfitting the training data. While it performs well on the training set (96.67% accurately), it does not show consistent improvements on the validation set (ChatGPT). This model would have performed better if I had included a pooling layer.

I learned that machine learning involves teaching computers how to recognize patterns in data and make predictions, which can excel through algorithms like neural networks. Last semester, I learned the core concepts of the machine learning pipeline, including neural network building, data loading, model training, and data visualization. This knowledge has provided a solid foundation for my current studies in computer vision, where I'm applying neural networks, particularly Convolutional Neural Networks, to image processing tasks like image classification and object detection. What worked well? What didn't? Why? What could have been done differently? Many things worked well, and others did not work well. Decreasing the learning rate to 0.003 improved the validation accuracy percentage to almost 1. Increasing the learning rate to 0.001 and 1 did not perform as well. Decreasing the batch size from 100 to 36 assisted with the model's performance. There was a suggestion made by ChatGPT that a pooling layer would help reduce the risk of overfitting by reducing the spatial dimensions of the photos.

The experience I gained with PyTorch, the open-source machine learning library, has significantly contributed to my academic growth. Working through convolutional neural networks gave me more practice with Python, as I encountered and fixed various coding challenges along the way. I gained new expertise in PyTorch, parameter tuning, data preprocessing, and visualization, becoming a more proficient code reader. Additionally, I gained a deeper understanding of how to start building my neural networks from scratch.

As I continue to study computer vision, I look forward to applying these concepts to more advanced problems like image segmentation and image captioning, where I can combine neural network techniques with computer vision-specific methods. The techniques explored in this lab have wide-ranging real-world applications, from enhancing recommendation systems in marketing and advancing radiology technology in healthcare to improving visual search capabilities in retail. The knowledge gained has the potential to simplify complex tasks and drive significant improvements across various industries.

References

ChatGPT. (n.d.). Retrieved from https://chatgpt.com/

IBM. (n.d.). *What are convolutional neural networks?* Retrieved from IBM: https://www.ibm.com/think/topics/convolutional-neural-networks

What is Transfer Learning? (n.d.). Retrieved from AWS: https://aws.amazon.com/what-is/transfer-learning/