Exploring Convolutional Neural Networks for Image Classification

Varit Kobutra

Houston Community College

Author Note

This journal is being submitted on June 30, 2024, for Professor Patricia McManus for Computer Vision ITAI 1378: 12461 at Houston Community College by Varit Kobutra.

Introduction

This reflective journal documents my experience with Lab 07, which involved setting up, modifying, and running a Convolutional Neural Network (CNN) for image classification to distinguish between chihuahuas and muffins. The lab was conducted using Google Colab, and the primary objective was to understand the CNN architecture, evaluate model performance, and explore potential real-world applications and ethical considerations. I would like to acknowledge Professor Patricia McManus for providing the notebook and instructions for this exercise.

CNN Architecture

Convolutional Neural Networks (CNNs) are a class of deep learning models specifically designed for processing data with a grid-like topology, such as images. Unlike traditional neural networks, CNNs employ convolutional layers to automatically and adaptively learn spatial hierarchies of features through backpropagation (Alzubaidi et al., 2021). The architecture typically includes convolutional layers, pooling layers, and fully connected layers, which work together to extract and classify features from input images (Voulodimos et al., 2018).

In this lab, the CNN architecture used included multiple convolutional layers followed by pooling layers and fully connected layers. Specifically, the `` `ChihuahuaMuffinCNN` `` model consisted of three convolutional layers with ReLU activation and max pooling, followed by two fully connected layers. The convolutional layers applied filters to the input images to detect features such as edges and textures, while the pooling layers reduced the spatial dimensions of the feature maps, thereby decreasing computational complexity (Scherer et al., 2010).

Model Performance

The performance of the CNN was evaluated based on accuracy and patterns in misclassifications. Overall, the CNN yielded higher accuracies compared to traditional neural networks, which aligns with existing literature on the superior performance of CNNs in image classification tasks (Alzubaidi et al., 2021). However, the CNN also demonstrated some interesting behaviors:

- **Training Time**: The CNN took longer to train compared to traditional neural networks. This is expected due to the increased computational complexity associated with the convolutional operations and the larger number of parameters (Voulodimos et al., 2018). Additionally, Google Colab used CPUs instead of GPUs during the lab, which further extended the training time.
- **Performance Fluctuations**: During some epochs, there were noticeable drops in performance, which quickly recovered in subsequent epochs. This could be attributed to the stochastic nature of the training process and the learning rate adjustments (Scherer et al., 2010).
- **Misclassifications**: Interestingly, the model made no wrong predictions for muffins but showed some inaccuracies in classifying chihuahuas. This could be due to the

higher variability in the appearance of chihuahuas compared to muffins (Nagpal & Dubey, 2018).

Comparison with Traditional Neural Networks

Compared to traditional neural networks, CNNs offer several advantages in terms of performance and feature extraction. Traditional neural networks require manual feature extraction, whereas CNNs automatically learn features from the data, making them more efficient and effective for image classification tasks (Alzubaidi et al., 2021). Additionally, CNNs are more computationally intensive, requiring specialized hardware such as GPUs for efficient training (Voulodimos et al., 2018).

Challenges and Solutions

During the lab, I encountered several challenges:

- **Computational Resources**: Google Colab complained about the number of workers requested in the provided project file. Reducing the number of workers to 2 resolved this issue.
- **Code Modifications**: Ensuring all necessary imports were present and correctly replacing placeholder code required careful attention to detail.
- **PyTorch Upgrades**: Upgrading PyTorch in Google Colab took longer than expected, which delayed the initial setup and execution of the notebook.
- **Hardware Limitations**: Google Colab used CPUs instead of GPUs, which significantly increased the training time and limited the computational efficiency of the model.

To overcome these challenges, I consulted the course materials and sought help from peers when necessary. Additionally, running the notebook in smaller batches and monitoring resource usage helped manage computational constraints.

Real-World Applications

CNNs have numerous real-world applications, particularly in the field of computer vision. They are widely used in medical image analysis, autonomous driving, facial recognition, and security systems (Alzubaidi et al., 2021). The ability of CNNs to automatically learn and classify features from images makes them invaluable for tasks that require high accuracy and efficiency.

Ethical Considerations

The development and deployment of CNN-based models raise several ethical considerations. Issues such as bias in training data, privacy concerns, and the potential for misuse in surveillance and security applications must be carefully addressed (Alzubaidi et al., 2021). Ensuring transparency in model development and implementing robust data privacy measures are essential steps in mitigating these ethical challenges.

Conclusion

Lab 07 provided valuable insights into the architecture and performance of CNNs for image classification tasks. The hands-on experience with setting up, modifying, and running the CNN model highlighted both the strengths and challenges associated with this powerful deep learning technique. By reflecting on these experiences and considering the broader implications, I gained a deeper understanding of the potential and responsibilities associated with deploying CNN-based models in real-world applications.

References

Alzubaidi, L., Zhang, J., Humaidi, A. J., et al. (2021). Review of deep learning: Concepts, CNN architectures, challenges, applications, future directions. Journal of Big Data, 8, 53. https://doi.org/10.1186/s40537-021-00444-8

Nagpal, C., & Dubey, S. R. (2018). A performance evaluation of convolutional neural networks for face anti-spoofing.

Scherer, D., Müller, A., & Behnke, S. (2010). Evaluation of pooling operations in convolutional architectures for object recognition. In International Conference on Artificial Neural Networks (pp. 92-101). Springer, Berlin, Heidelberg.

Voulodimos, A., Doulamis, N., Doulamis, A., & Protopapadakis, E. (2018). Deep learning for computer vision: A brief review. Computational Intelligence and Neuroscience, 2018, 7068349. https://doi.org/10.1155/2018/7068349