Module 8: Portfolio Project

Brady Chin

Colorado State University Global

CSC510-1: Foundations of Artificial Intelligence

Dr. Bingdong Li

March 9th, 2025

Portfolio Project

This portfolio project is a computer vision project that uses deep learning to sort recyclables. It seeks to create an AI model that can recognize and classify different kinds of trash to aid in the user correctly recycling their waste. By using machine learning, the project demonstrates how to utilize AI to assist in sustainability as well as establish baseline competence in model training and classification.

Dataset

The model is trained on the TrashNet dataset, which is an open-source dataset and contains over 2,500 images of trash and recyclable material (Garythung, 2023, June 1). The images are labeled into six categories: plastic, paper, metal, glass, cardboard, and general trash. The dataset includes a vast amount of labeled images to allow the model to distinguish between the materials correctly making it a well suited dataset for material classification tasks.

Programs

It is a two-program project. The first program, ecoscan.py, processes the data, expands it, and divides it prior to building, training, and testing a deep learning model. The model that is trained is saved and with a backup model_backup.keras provided. The second program, evaluate.py, allows users to classify one image by passing it through the trained model and picking out the material that has the highest probability. Instructions on running both programs are also included in the submission.

Tools, Libraries, and APIs

The project heavily utilized Python machine learning and data manipulation packages like TensorFlow, Keras, NumPy, Pandas, and Scikit-learn. TensorFlow and Keras were employed as the primary tools for model development and training, while Pandas and NumPy assisted in handling datasets and numeric calculations. The MobileNetV2 model was borrowed from the Keras applications to use a pre-trained deep model in order to accelerate the training process. Furthermore, Matplotlib was utilized when visualizing the outcomes.

Search Methods

Efficient data management was critical to deal with the dataset that had six recyclable materials. Directory scanning from the os library was used to load the images to Pandas DataFrames for this project. Then the dataset splitting was done using the train_test_split function from Scikit-learn, thus assuring a well-distributed training set, validation set, and testing set. With ImageDataGenerator API-based real-time data augmentation, various rotations, flipping, and shifting are considered to increase model generalization (Chollet, F., 2016, June 5).

Deep Learning Models

This project utilizes a convolutional neural network (CNN) built upon MobileNetV2, a lightweight yet powerful deep learning model (Keras Documentation, 2025). Instead of training the model from scratch, transfer learning was employed by freezing the pre-trained layers of MobileNetV2 and fine-tuning the later dense layers. The network container custom preprocessing layer, batch normalization, and regularization methods such as L2 weight decay for improved performance and to prevent overfitting. The model was trained with an Adam optimizer and categorical cross-entropy loss function, employing class weighting to counteract data imbalance.

Expert Systems

A rule-based system was drawn in to assist users with proper recycling after deep learning predictions were completed. After classifying an image, decision rules define step-by-step instructions on how to recycle. Thus, it suggested, once detected as 'glass', rinsing bottles and making sure they weren't broken before disposal. This expert system provided a practical applicability of the AI model to bridge the gap between what it predicts and the actual recycling practices.

Knowledge and Development Challenges

During the development stage, I found this project very challenging since I did not have any machine learning experience. To add to the complication, I decided to utilize a convolutional neural network rather than find a project that would use a more simple neural network. I spent the first few weeks of this project doing research and understanding the theory behind CNN's. I found some helpful resources such as geeksforgeeks.org (2024, October 10) and Deeplizard (2017, December 9).

During the development stage, I created my own CNN from scratch but I found it to be very inaccurate and it took a long time to train. I used the weights and biases for Keras version of MobileNet v2 (Xhlulu, 2019) and used it to largely improve the accuracy and reduce the loss of my model.

Early in the development process, I found that the accuracy of my model was very high and my loss was very low. While I initially thought this was a good thing, I noticed that the validation accuracy was much lower than the accuracy and the validation loss was very high. This is a sign of overfitting. I then had to add data augmentation to counter this. Afterwards, I was getting much better results.

Future Developments

While this project is now complete, there can be future developments that make it more user friendly. For example, it is difficult to run the program and test an individual image. The user will have to place an image in the correct folder then properly type in the image name and extension when prompted. A future development could be to make a UI for this project. It can consist of a "drag-and-drop" section where the user can place their image into it to have the model predict it. This UI will also display the prediction in a much cleaner way than printing out the results in the terminal.

Furthermore, instead of accepting still images, a live view can be used to recognize items and make predictions. This will give the user real-time predictions based on what their laptop or phone camera can see.

I would also like to add more images to the dataset. The “trash” class had significantly less images compared to the other classes which resulted in lower accuracy and F1 scores when evaluating. I would also like to add more diversity to the images. The TrashNet dataset had the items on a clear, white background so that the material was easily recognizable. In the real world, this might not be the case as I noticed lower accuracy when I would load my images with varies backgrounds.

Conclusion

This project successfully demonstrates the use of deep learning for recyclable material classification. Through material learned in this course, such as search methods, neural networks, expert systems and API’s and libraries, I was able to create a model that can accurately classify different waste items. While there are improvements that can be made, I am proud of this project as it was my first successful and useful AI project.

References

Chollet, F., (2016, June 5) *Building powerful image classification models using very little data.* The Keras Blog. https://blog.keras.io/building-powerful-image-classification-models-using-very-little-data.html

Deeplizard, (2017, December 9) *Convolutional Neural Networks (CNN's) explained.* YouTube. https://www.youtube.com/watch?v=YRhxdVk_sIs&t=1s

Garythung, (2023, June 1) *trashnet.* GitHub https://github.com/garythung/trashnet

geeksforgeeks.org, (2024, October 10) *Introduction to Convolution Neural Networks.* GreeksforGeeks. https://www.geeksforgeeks.org/introduction-convolution-neural-network/

Keras Documentation, (2025) *MobileNet, MobileNetV2, and MobileNetV3.* Keras. https://keras.io/api/applications/mobilenet/

Xhlulu, (2019) *MobileNet v2 Keras Weights.* Kaggle. https://www.kaggle.com/datasets/xhlulu/mobilenet-v2-keras-weights