SUSTAINED RECYCLING

Alejandro Taboada Esteban

Paula Abad Carvajal

Sergio Rodriguez Valbuena

Index

1. Introduction and summary of contribution

2. Steps we followed and difficulties we found

3. Description of the final result

4. Comparison with already existing approaches

5. Demo and real experiments

6. Conclusion

1. Introduction and summary of contribution

The project we're embarking on involves developing an innovative system where users can input an image of an item destined for disposal. The model would recognize the image and with the use of deep learning algorithms, the system will analyze the image uploaded, classify the object into its respective waste category and finally recommend the appropriate waste bin for disposal. This technology aims to streamline the recycling process by providing users with accurate guidance on segregating waste, fostering a more sustainable and environmentally conscious approach to waste management.

This technology holds significant potential in contemporary society, where a substantial portion of the population expresses a desire to engage in recycling but may lack clarity regarding the appropriate disposal receptacles for their waste, especially old people as most of them have no idea about this topic. Beyond its industrial applications aimed at optimizing recycling facilities, there exists the possibility of developing a mobile application, for instance, to assist individuals in effectively managing their waste disposal for recycling purposes.

The work was a collaborative effort among the three of us, ensuring a thorough process. Each individual contributed to different aspects, and we implemented a review process where one person would perform a task, followed by another person reviewing and validating the work to ensure accuracy and completeness. By doing everything between all three of us, we ensured that no mistakes were committed, and for every step we got different ideas and opinions from every group member, which helped get the best final result possible. Furthermore, it was a faster process as it is not one person thinking but the three of us.

2- Steps we followed and difficulties we found

After a long search, we found our dataset in Kaggle. The data we choose is categorized into battery, biological, brown-glass, cardboard, clothes, green-glass, metal, paper, plastic, shoes, trash, and white-glass. In the next step, we downloaded this dataset and organized it based on these distinct classes. This process will enable us to efficiently categorize and work with the data for further analysis and modeling purposes.

Initially, our data preprocessing efforts were limited, mainly focusing on data augmentation. However, as we progressed, it became evident that the performance metrics of all the models we experimented with were unsatisfactory. So, we examined the dataset manually in the database and we found that there were duplicates in the dataset (some images were the same ones) and other problems that we should address. After realizing this, we decided that we had to do a deeper data preprocessing process before doing anything to the data, because it would be crucial for the final outcome of our work.

So, we prioritized the identification and removal of duplicate entries, because there were duplicate images in the data as we previously mentioned. This process ensures the dataset's precision by eliminating redundant or repetitive information which may variate the results later. By systematically purging duplicates, our goal is to streamline the dataset, thereby enhancing its quality and optimizing the accuracy of subsequent analyses or machine learning models.

In addition to eliminating duplicate images from our dataset, we recognized the importance of broadening our image collection by incorporating relevant online images matching the categories central to our project. To achieve this goal, we delved into the utilization of Google Custom Search, facilitating the curation of additional images that precisely aligned with our project's specific categories.

During our quest to establish this functionality, we initially explored alternative methods, including the creation of an API key via Google Cloud. However, multiple challenges emerged along the way, prompting us to experiment with various approaches as these ones were not giving anything. Despite encountering several obstacles, our persistent efforts eventually paid off as we successfully implemented the current code. This code now empowers us to conduct image searches effectively and seamlessly integrate the retrieved images into our dataset. It uses google search to look for 100 images for each category, download them and add them to the dataset. With the use of Google Custom Search, we end up adding around 500 images for each one of the categories. When an image gives an error and can't be added because of its characteristics, the program just skips it and goes to the next image, and with this procedure we avoid errors that are very common when using Google Custom Search.

As a final step in our data preprocessing pipeline, aiming to further enhance our accuracy, we are set to perform data augmentation on our dataset, expanding it by 40%. In this augmented dataset, images will undergo rotations ranging from 0 to 180 degrees, and additionally, certain images will feature varying degrees of zoom. This augmentation process is anticipated to diversify our dataset, introducing variations that can assist in training a more robust and generalized model.

Moreover, leveraging this refined dataset, our aim is to construct a structured data frame containing the image paths associated with each class. This data frame will act as a systematic reference, providing smooth access to the images for processing, model training, and subsequent analysis. By linking the images to their corresponding classes via this structured framework, we anticipate facilitating more efficient data management and robust model development. Furthermore, for the following models it was easy to work with more structured data as it is a dataframe.

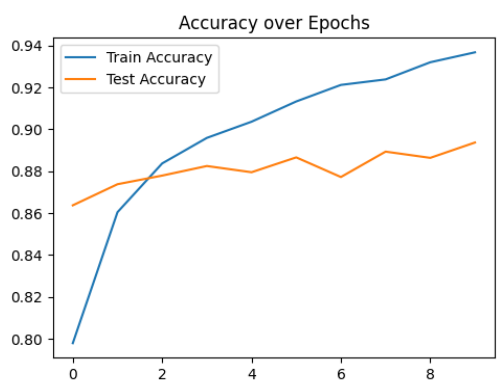
After compiling and organizing our dataset, the next step involves conducting an initial exploration and preview of the newly created dataframe. This preliminary phase of visualization aims to offer a concise overview of the dataset's structure, enabling us to grasp its dimensions, examine the distribution of labels, and gain insights into potential patterns or irregularities within the data.

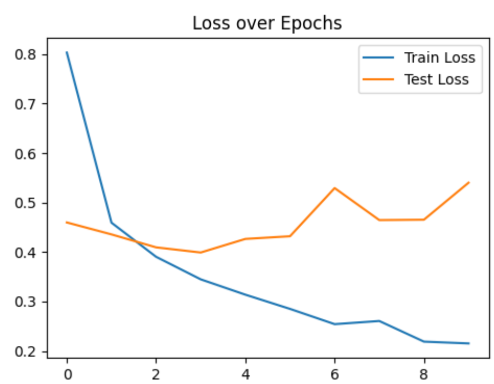
To commence our data exploration, we initiated by displaying three random images from each category, acknowledging the potential reduction in image quality due to limited RAM (we used 64 x 64 images for the data). Despite this limitation, it offered a representative glimpse into the visual content of each category, providing a foundational understanding of image characteristics.

Subsequently, we created a bar chart to analyze the distribution of images across labels. The graphical representation highlighted that categories such as 'biological' and 'clothes' stood out prominently, exhibiting a notably higher number of images compared to others. This disparity suggests a potentially broader range and abundance of visual diversity within these specific classifications. Conversely, the remaining labels demonstrated a relatively consistent count, hovering around approximately 1000 images each. This observation implies a more uniform distribution of visual content across these categories, depicting a balanced representation of images.

Model ResNet

As a first approach, we'll begin by constructing an initial model using ResNet architecture. Initially, we'll split the dataset into training and testing subsets. Post that, we'll train the model and evaluate its performance. To get this initial model, we used the pre-trained ResNet50 model, adding some new layers to enhance its improvement. Plotting the results will allow us to visualize the model's accuracy, providing insights into how effectively it classifies the test data. The results we obtained where the following, with a final test accuracy of 0.8937

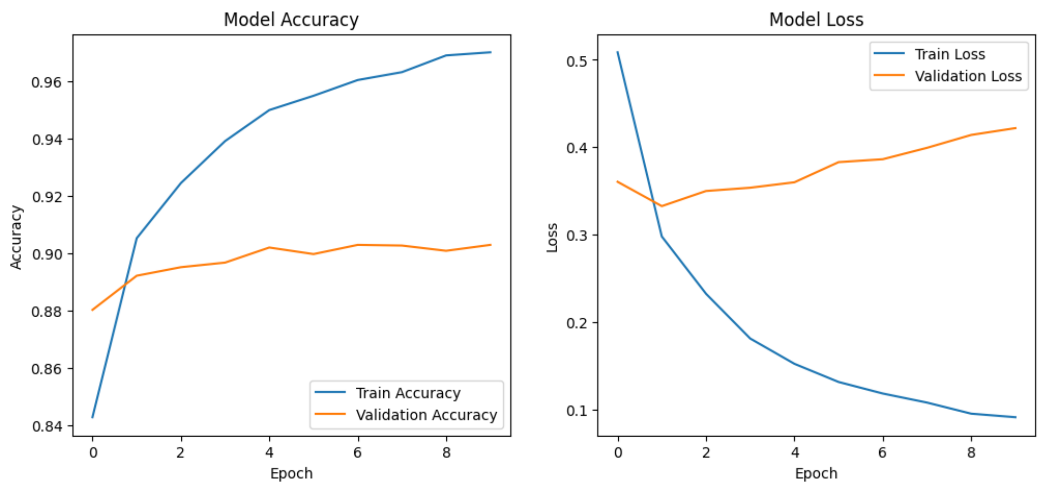




Upon evaluating the performance, we concluded that the ResNet50 model could potentially be improved by exploring other architectures. Consequently, we made the decision to create a final model, aiming to enhance its capabilities and achieve superior performance.

Model EfficientNetB0

In the upcoming phase, our focus will be on refining the deep learning model's architecture and parameters to enhance its performance. Our objective is to construct a final model leveraging a pre-trained architecture, specifically EfficientNetB0. To do this, we tried with many models, we tried with a VGG19, a MobileNetV2 and a MobileNetV3 and a DenseNet with a depth of 169 and different growth rates, pooling and batch sizes, but the one that gave us the best results was EfficientNetB0. We first employed this model to define our final model architecture and subsequently compile it. Following this, we trained the model for approximately 10 epochs, achieving an accuracy of 0.9028. We know that with more epochs we could have obtained a more accurate model, but the training would have lasted too much, and to create a demo it is not practical. Of course, to create a final application we would have used way more epochs. Finally, we plotted all the information about the model accuracy and model loss on two graphs



Finally, we developed an interactive demonstration that allows users to upload images directly to Google Colab for classification purposes. This user-friendly interface enables individuals to submit their images seamlessly, leveraging the power of our refined model to classify them based on predefined categories or labels. By harnessing the capabilities of Google Colab's environment, users can obtain instant classification results, gaining insights into the content or category to which their uploaded images belong. This demonstration not only showcases the functionality of our model but also offers a practical, hands-on experience, demonstrating its real-time applicability and providing an intuitive platform for users to engage with and understand the model's classification capabilities firsthand.

4- Comparison with already existing approaches

Recycling is a matter that affects all of us. We need to preserve our planet and this is one of the ways in which we can do it. Therefore it is not strange that the internet is full of different models and ideas of how we can make recycling easier and how we can help people increase their knowledge of it.

After conducting an online search for articles with applications similar to ours, we came across one that particularly caught our attention, known as “Humans in the Loop”.

The article provides an insightful look into the application of AI in the recycling industry, similar to the EfficientNetB0 model that we described. The “Humans in the Loop” article emphasizes the use of AI to combat pollution and climate change by promoting recycling and making it more efficient. They highlight the importance of sorting the items for recycling. This is similar to the

EfficientNetB0 model, which assists in identifying recyclable objects to streamline the recycling process.

We found another article from the University of Colorado Boulder's Environmental Center that offers a comprehensive overview of the use of AI and robotics in recycling, which shares similarities with the EfficientNetB0 model we described.

In the article, they are also worried about the future of our planet and therefore they decided to create AI and AMP robots. These innovations are positioned as solutions to address the perceived inadequacies in the recycling methods prevalent in the United States. The way they work is as follows : The AMP Cortex robot has three arms that allow for a large range of motion and fast picking and placement. Together, AMP's AI and robotics system can quickly sort through nearly all recyclables and contaminants found in single-stream recycling.

We found another article that talks about the different deep learning techniques that have been used for this same purpose in the last years. It says that in 2021 A MobileNetV3 was used, and it had a 92% of accuracy, which is very similar to ours, a little bit higher, but the model was trained with way more epochs, so if we did the same with ours the accuracy could be even higher than the MobileNets's. In 2022 Zhao et al. Created a CNN model by combining MobileNetV3 with LSTM (long-short term memory), which is the best model so far. All the work described in the article employs MobileNet, and that's why we tried to implement it too, but the model we got with it was not the better performing one we could find.

Also, we found on the internet and Kaggle other kinds of models to classificate garbage into categories. For example, we found in Kaggle a MobileNet with a 82% testing accuracy, which is significantly lower than ours, and the model has been trained with 7 epochs, so it is very similar to ours.

5- Demo and real experiments

In our demo, we leverage the EfficientNetB0 model configured with a Dropout layer of 0.5, global average pooling 2D, and a Dense layer of 1024 and ReLU activation. The demo first resizes the input image, so that the model can handle it, into a 64 x 64 image, applies the trained model, and once it has classified it in a category, it assigns it into its corresponding dumpster. It operates by allowing users to input an image of the object they wish to classify. Upon submission, the model processes the image, swiftly identifies the type of object depicted, and promptly recommends the specific recycling bin or container where the item should be discarded for proper recycling. This seamless process aids users in making informed decisions on how and where to dispose of their items, contributing to a more efficient and environmentally conscious recycling system.

The following are the result of experiments where we input different images we took of objects in our house

1/1 [==============================] - 0s 25ms/step
The image is a shoe and it has been classified as: shoes
It should go to the: pink dumpster

1/1 [==============================] - 0s 28ms/step

The image is a cardboard and it has been classified as: cardboard
It should go to the: blue dumpster

1/1 [==============================] - 0s 30ms/step
The image is a white glass and it has been classified as: white-glass
It should go to the: green dumpster

1/1 [==============================] - 0s 36ms/step

The image is a tshirt and it has been classified as: clothes
It should go to the: pink dumpster

6- Conclusion

As a summary of everything we have said, our project focused on developing an innovative waste management system employing deep learning algorithms to recognize and classify items for efficient recycling. With this technology we aimed to bridge the gap between users' intentions to recycle and the lack of clarity regarding appropriate waste disposal.

During this process, we conducted exploratory data analysis, visualizing the dataset's structure and label distributions. We initiated model development using the ResNet50 architecture, achieving a test accuracy of 0.8937. However, aiming for improved performance, we transitioned to EfficientNetB0, achieving an accuracy of 0.9028 after model refinement and training.

We demonstrated that our approach aligns with existing applications like "Humans in the Loop" and robotics systems described by the University of Colorado Boulder's Environmental Center, emphasizing AI's role in efficient recycling methods.

Finally, in conclusion, our project's innovative waste classification system, employing deep learning techniques, aims to empower individuals and industries in making informed recycling decisions. By leveraging AI, we've contributed to fostering a more sustainable and environmentally conscious approach to waste management, offering a practical solution for efficient waste segregation and recycling. The demo and experiments conducted demonstrate the real-time applicability and efficacy of our model in facilitating informed waste disposal decisions.

Now, there are some things that we could have done differently. For example, a better working model could have been obtained if we had found a better and more broad dataset, with more garbage pictures from different angles, shine and background, and of course the performance would have been better if we had trained the model with more epochs.

To sum up, during the process of creating this model and demo, we all have participated in each part, which has helped us learn some things that we did not know before. For example, we did not think that data preprocessing was so crucial, but now, after first doing a very simple one and later a way more complex one, we have seen that it has a huge impact on the precision of the final model. We all have liked this experience, because it was aimed to help with an actual problem, and we have developed a further understanding of many deep learning models and their characteristics.

7- Bibliography

1. Aref Ariyapour. *Humans in the loop.* https://humansintheloop.org/
2. University of Colorado. Kayla Vasarhelyi. *AI robotics in recycling*

https://www.colorado.edu/ecenter/2022/04/06/ai-robotics-recycling

3. Kaggle. *Dataset for garbage disposal.* https://www.kaggle.com/datasets/mostafaabla/garbage-classification/code
4. A model we found for the same database online

   https://www.kaggle.com/code/davidmax000/garbage-classification-mobilenetv2-92-accuracy
5. Another approach for this same problem we found on internet

   https://www.sciencedirect.com/science/article/abs/pii/S0956053X23001915
6. Another different article about nowadays approaches for his problem

   https://papers.ssrn.com/sol3/papers.cfm?abstract_id=3563564
7. These are some books we used for the code

   "deep learning for computer vision: expert techniques to train advanced neural networks using tensorflow and keras"

   "Computational Methods for Deep Learning Theoretic, Practice and Applications

   "Pattern Recognition and Machine Learning Christopher M. Bishop"
8. And we have to add that we also used ChatGPT to help us get our Google Custom Search credentials and Google CLoud API, and we also asked it for the best models we could try to approach our database given its characteristics and the data preprocessing we had done.
9. Finally, we also watched several youtube videos that explain how to use EfficientNet, how to create custom layers and other important characteristics of this model that could help us in this youtube channel https://www.youtube.com/@CodeWithAarohi