Garbage Classification With CNN, XGBoost, and Logistic Regression

Yifan Sun
Department of Computer Science
Texas A&M University
College Station, United States
sunyifan1113@gmail.com

***Abstract*—This study is aimed to build three different machine learning models: CNN, XGBoost, and logistic regression to accomplish garbage image classification. CNN also uses the transfer learning from ResNet50 to make the training more efficient and accurate. The data set used for deep learning structures has a total of 2527 images with 6 different classes, which is divided into training, validation, and test sets. Finally, the performances of the three models measured in accuracy are compared. CNN and XGBoost have the potential to do well on this task, but logistic regression is not suitable.**

***Index Terms*—CNN, XGBoost, logistic regression, image classification, transfer learning**

I. INTRODUCTION

The material life of people nowadays are becoming satisfying, but more and more domestic waste with various types are generated around us. At the same time, the significance of garbage classification cannot be ignored.

Garbage classification is the prerequisite for the scientific treatment of garbage, and it is the foundation for the reduction, recycling, and harmless treatment of garbage. Modern garbage contains chemical substances. If we do not correctly classify them into waste, it will not only mutually pollute other types of garbage, but also go into our environment without harmless treatment , enter the ecosystem with the circulation of the earth, pollute the water sources and land, and ultimately affect people's health as it accumulates through plants and animals. According to MWH Environmental, 13 tons of hazardous waste is produced every second in 2021, which is 400 million tons per year. [1]

On the other hand, if we accurately classify each type of garbage, we can extract an abundance of valuable resources from them, as well as protect people's health in the long run. For example, cardboard and paper can be recycled to protect forest resources. Glass, metal, and plastic can also be recycled through processing, shredding, and melting. We can see that each type of recycling garbage has its unique way of processing that cannot be substituted, which puts emphasis on the accuracy of garbage classification. Some types of trash can even be used as fertilizer after treatment. If we just simply regard everything as trash and landfill or staking them, we not only lose a lot of valuable resources, but also put stress on our lacking land resources.

Currently, only a few countries have the technology of automated garbage classification. A lot of them are still in the progress of researching, and only small scale deployment has been made by some companies in the form of the reverse recycling vending machine [2] . Most countries are still in the phase of suggesting people to classify their waste by propaganda and media. However, only a small proportion of people really know the importance of garbage classification and classify the garbage to their best knowledge, but even those people cannot classify every piece of garbage accurately. So automated garbage classification is our future.

II. LITERATURE REVIEW

Currently, most data scientists around the world are using deep learning, especially Convolutional Neural Network (CNN) to enable computer vision and find representations of images, in combination of other machine learning models, especially Support Vector Machine (SVM) to classify images.

For example, in the study by Umut Özkaya1 and Levent Seyfi [3], they developed various fine-tuning models for garbage classification. Their approach to the garbage classification problem is to take a dataset of images, let them pass through one of the fine-turned CNN models with the last layer connected with softmax regression or take the intermediate representation to classify them using SVM. The fine-tuned CNN architectures include AlexNet, VGG-16, GoogleNet, ResNet and SquezeeNet. This is based on the concept of transfer learning. These architectures are pretrained with millions of images of various objects so that they have initialized layers of filters and weights associated with them. In this way, the network will converge more quickly and accurately.

Most other studies implement similar solutions using fine-tuned CNN, but few of them implements other machine learning models only to realize garbage classification. In this report, I will use three popular classification models in machine learning: CNN, XGBoost, and Logistic Regression to classify garbage images and compare their performances on the dataset by using the flattened images as data points and their pixels as features.

III. PROBLEM FORMULATION

In this study, the problem is to develop a machine learning model that could take a raw image of garbage taken in a well-lighted environment and classify it into one of the six categories: cardboard, glass, metal, paper, plastic and trash.

IV. Proposed Solution

This study is going to use three different classifier models: CNN, XGBoost, and Decision Tree.

A. data preprocessing

First, each category will be assigned with one of the labels from 0 to 5. Then each image in their corresponding folder (folder name is the garbage type) will be assigned one of the labels.

TABLE I
Garbage Type to Label

Garbage Type	Label
cardboard	0
glass	1
metal	2
paper	3
plastic	4
trash	5

After that, each image will be flattened into a 1-d array of numbers from 0-255, and each array will have length of $length \times width \times 3$, because these are colored images with dimension of $length \times width$ and each pixel contains a tuple of three values representing the RGB values. An example image obtained after this step is shown in figure 1.

Fig. 1.

Moreover, each image needs to be normalized in order to improve the stability of the models by making sure each pixel in a specific location has the same scale compared to all the other pixels so that they will have equal significance before training. In this study, the standard normalization with mean of 0.5 and standard deviation of 0.5 are used to transform each image. The equation is given by

$$\text{output} = \frac{\text{input} - \mu}{\sigma} \tag{1}$$

where input, output, μ, σ are the input value, output value , mean, and standard deviation of the specific channel respectively. So that the normalized image looks like the example shown in figure 2.

Fig. 2.

In addition to normalization, it is necessary to compress the image since time and hardware resources available do not allow these model to train the dataset with all the features (pixels). Some attempts were made and the CNN takes about three hours to train on uncompressed images for 10 epochs, and the accuracy is not very impressing. This is for the reason that each image contains about 600,000 features, and it is computationally expensive for a dense neural network to backpropagate on this huge dataset. At the same time, training on a subset of the images is not desirable since the dataset only contains about 2,500 images, and it will be impossible to generalize and hard to interpret meaningful results if the number of features is much higher than the number of data points.

Therefore, each image is compressed into dimension of 16×16 using average pooling algorithm, which is to take the average RGB value of all the pixels in a specific region and fill the region with this value. The resulting example is shown in figure 3.

Fig. 3.

After assembling the compressed images into the dataset, stratified sampling method is used to randomly divides the dataset into training, validation, and test sets according to the proportions of 0.8, 0.1, 0.1 respectively. Stratified sampling ensures that the training set contains enough examples for each class of garbage to train on, since the dataset has much fewer number of trash images than all the other categories, so that the model will not be evaluated on a test set that contains many examples that it has not encountered much.

B. Convolutional Neural Network (CNN)

This study utilizes a pretrained fine-tuned CNN called ResNet50 [4]. It is a convolutional neural network that is 50 layers deep. It is pretrained with more than a million images from the ImageNet database [5]. It can classify images into 1000 object categories, which is good for feature extraction of images. The goal is to extract essential features from the image. From the first few layers, the primitive features are extracted. As the training goes deeper into the network, more detailed and complex features are extracted. If a untrained shallow convolutional neural network is used for this task, it is unlikely to produce meaningful result since we have $16 \times 16 \times 3 = 768$ features even after image compression, but we only have 2274 images to train on, which is not enough for this task. As a result, the architecture of the CNN used in this study is a pretrained ResNet50 already initialized with weights connected with a fully connected layer at the end to the output.

The activation function used is sigmoid function, which is the most common activation function used for classification problems which could stabilize nodes in each layer to have values between 0 and 1. Cross entropy loss is used as the loss function, which measures how well a classification model performs. Adam optimization algorithm is used for back propagation, which is one of the most popular optimizers for training deep learning models. Adam combines the best properties of AdaGrad and RMSProp algorithms to handle sparse gradient on noisy problems.

In order to make the architecture perform better, several hyperparameters including batch size, number of epochs, and learning rate are tuned by using five-fold cross validation. Batch size refers to the number of sample images that pass through the neural network each time; number of epochs refers to the number of times the model is trained on the entire training set; learning rate refers to the rate by which the optimization function steps for back propagation, which is crucial for determining stability and convergence of the model. Because cross validation will train a separate model for each iteration, it is computationally expensive to cross validate on the entire dataset. Instead, a representative subset that is 1/10 of the size of the original dataset is selected to make the process efficient.

The model will ultimately produce a list of probabilities as prediction for an image to be one of the classes that sum to 1. Since we are only interested in knowing what class does the image most likely belong to, so the class with the highest probability is given as the prediction. This prediction is compared against the actually label associated with the image, and the performance of the model is calculated based on the proportion of the dataset that it classifies correctly.

By assumption, the training loss will decrease and the training accuracy will increase over the epochs. In contrast, the validation loss should decrease with the training loss, and then flattens or starts to slowly increase over the epochs in theory, because the latter represents how the model will overfit to the training set. And similarly, the validation accuracy is assumed to increase with the training accuracy, and then flattens or starts to slowly increase over the epochs as the model overfits. For each epoch, the current version of the model will be stored. Then from the plots of comparison of training accuracy and validation accuracy, the epoch from which the model overfits will be identified, so that the version of the model corresponds to the epoch before it will be retrieved as the final model that will be evaluated against the test set. The final performance of CNN will be evaluated with its accuracy on the test set.

C. XGBoost

XGBoost is an optimized efficient gradient boosting algorithm which provides a parallel tree boosting. Based on the gradient boosting framework, XGBoost constantly adds new decision trees to fit a value with residual multiple iterations and improves the efficiency and performance. XGBoost uses a Taylor expansion to approximate the loss function, unlike gradient boosting. It usually uses fewer decision trees to obtain a higher accuracy.

As a gradient boosting framework, XGBoost builds up the predictor using multiple weak classifiers as decision trees. It generally has higher performance than simple decision tree classifiers since it will not allow several of the strong features to dominate the decision. It has been proved to be one of the most scalable classifier in most scenarios, so that it could deal with the huge amount of data stored in images efficiently, which makes it a suitable model for image classification. Some attempts are made to fit the dataset to a decision tree classifier, and it has only an accuracy of 0.4325 on the test set.

In order to make XGBoost perform accurately, several of its parameters need to be tuned before use. In this study, the parameters to be tuned include max depth, which represents how deep a decision tree can grow before it is pruned, number of estimators, which represents how many decision trees are combined into the model, and learning rate, which represents how large the model will shrink for each step. The hyperparameters are tuned similarly using ten-fold cross validation, and the estimator with the best accuracy is used for the model. However, the problem of length of computation is encountered again, so a subset of 1/10 of the size of the original dataset is taken out for cross validation.

XGBoost's performance is also evaluated by taking the trained model and measure the accuracy against the test set.

D. Logistic Regression

Logistic Regression has been one of the most simple and interpretable classification model in machine learning. It applies the sigmoid function to the linear regression model to bound the output to be between 0 and 1 for binary classification. Since logistic regression is naturally suitable for binary class classification, we need to modify it to be several binary classification problems for multiclass classification of the study.

Similar to the previous models, logistic regression has hyperparameters to be tuned, which include the regularization constant C, and the choice of regularization parameter: either l1 or l2 norm. Regularization ensures that logistic regression does not get too complex in degrees to keep its variability in check.

By assumption, logistic regression is not suitable for high dimensional data, especially image classification problem since images usually have tremendous amount of features. However, there is chance that this simple model could produce results that are not much worse than the complex models as CNN and XGBoost.

Data Description

The dataset is taken from www.kaggle.com with six folder of images of size 512×384. Each of the six folders contain images from a certain class: cardboard, glass, metal, paper, plastic, and trash. They are all colored images taken in a well-lighted environment with white background with the garbage positioned in the center of the images. Some examples of the images are shown in figure 4.

There are 403 images of cardboard, 501 images of glass, 410 images of metal, 594 images of paper, 482 images of plastic, and 137 images of trash, summing up to 2527 images in total.

Results

Figure 5 shows the training and validation accuracy of the first 20 epochs in CNN. We can see that deviate from our assumption in many ways. The training accuracy is increasing slowly with small fluctuations, and it does not reach a high level up to this point. The validation accuracy is generally increasing with the training accuracy, but it is unstable across the epochs with some very low accuracies and some very high accuracies. This can be due to the use of a deep neural network (ResNet50) with little regularization, because the neural network could choose from many ways to classify the images, but only some of them are good generalizations, while others are overfitting on the training set. It is consistently increasing with the training set because the validation set is very similar to the training set.

Figure 6 shows the training and validation loss of the first 20 epochs in CNN. This shows similar result as figure 5. The training loss is consistently decreasing, while the validation loss is generally decreasing but fluctuating a lot. This is also due to the depth of the neural network used and the abundance of choice. Since we have compressed the image into 16×16 pixels, the image might be too noisy to be classified into a specific category of object.

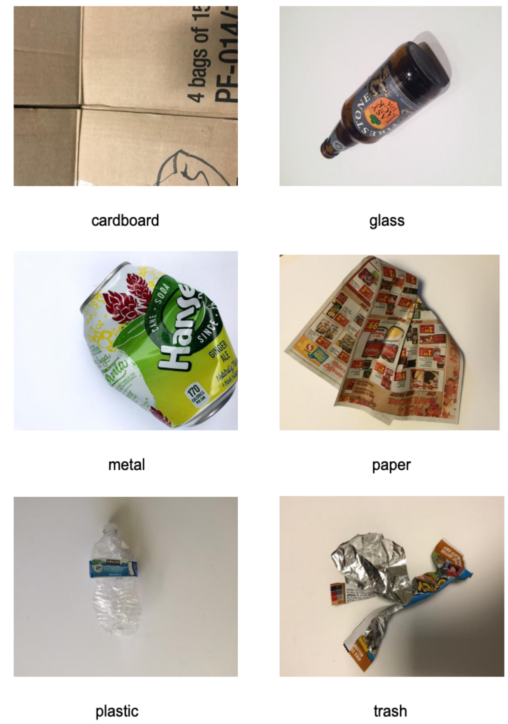

Fig. 4.

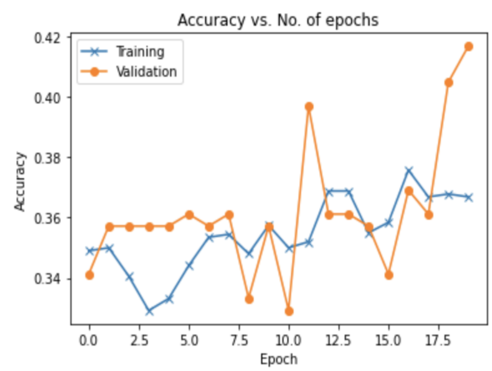

Fig. 5.

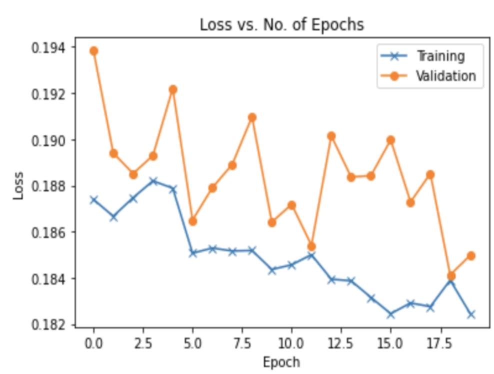

Fig. 6.

By training this model with another 50 epochs with only training set, because validation and test set are very similar to the test set so the model does not overfit much, the model ultimately reach a test accuracy of 0.5992.

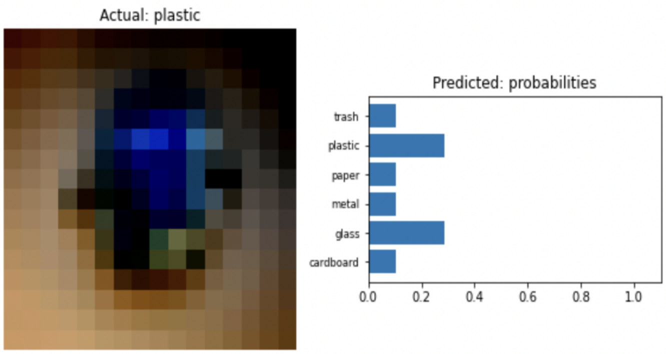

Fig. 7.

Figure 7 shows how the trained CNN classifies images. By plotting the softmax of the prediction probabilities, we can see that the model can hardly decide whether the image is plastic or glass, which is reasonable because we have compressed the image too much so that it is too noisy to make accurate decisions due to limited time and hardware resources. The image is actually plastic, so the model is very close to the correct answer.

The three models use the same split of training, validation, and test sets and all use accuracy as the metric to measure performance. The resulting accuracy is shown in table 2.

TABLE II
ACCURACY OF MODELS ON TEST SET

Model	Accuracy
CNN	0.5992
XGBoost	0.6825
Logistic Regression	0.4444

From the result we can imply that XGBoost is a good model for garbage classification as it has the highest accuracy among all the three models, which proves that CNN is not required to do garbage classification. The low accuracy of CNN is expected because of limited time and hardware does not allow for training on clearer images. More epochs can also be run to further improve the accuracy of CNN. It might have a much higher accuracy if it can be trained on uncompressed images, which is more complex, and the deep neural network can certainly handle the task.

Similar to the assumption, logistic regression is too simple for the task, as it does not perform well with high dimensional data.

Limitations and Future Directions

The major limitation for this work is the limitation of time and hardware resources which do not allow for a comprehensive training on the dataset, especially with the CNN that takes the longest to be fully trained. In addition, the hyperparameter tuning process using cross validation can be run on the entire dataset if unlimited time and hardware are available so that it can give a more comprehensive result on the entire dataset.

Since XGBoost has shown great potential to do well in garbage classification, a more complex model can be built by combining XGBoost with CNN. Basically CNN will be used for deep feature extraction, which will feed in to XGBoost to do classification. This more complex model might be capable of classifying garbage images with even more characteristics.

Further data can also be collected to extend the dataset, so that the model will have more information to be trained on, resulting in potential even higher accuracy.

In order to make this model practical and functioning in real life, there are still many steps to take on. One potential method to collect more data is to hire garbage classification experts or trained volunteers to manually classify incoming garbage from people living in a region, and the machine will automatically take pictures of these garbage and put label on them so that the size of the dataset will continue to increase, and the model will also train on its dataset continuously.

Conclusions

The task of classifying garbage is realizable using various machine learning models. Among these models, CNN and XGBoost have the potential to classify garbage accurately, while logistic regression does not perform well with high dimensional data. Although the study does not achieve high classification success due to limiting time and hardware resources, the models can be trained on larger dataset with more characteristics and clearer images if resources are available. In addition, more data can be collected continuously in the future to extend the dataset, and more classes of garbage can also be added so that the trained model will provide more accurate information. XGBoost and CNN can also be combined to make a even more complex model for more accurate results in the future.

References

[1] Facts and statistics about waste for 2021. HWH Environmental. (2021, March 31). Retrieved November 13, 2021, from https://www.hwhenvironmental.com/facts-and-statistics-about-waste/.

[2] The role and significance of Garbage Classification. INCOM TOMRA Recycling Technology (Beijing) Co., Ltd. (n.d.). Retrieved November 13, 2021, from http://www.incomrecycle-rvm.com/news/significance-of-garbage-classification.html.

[3] Özkaya, U.; Seyfi, L. (n.d.). Fine-tuning Models Comparisons on Garbage Classification for Recyclability. Retrieved December 13, 2021, from https://arxiv.org/pdf/1908.04393.

[4] Deep Network designer. ResNet-50 convolutional neural network - MATLAB. (n.d.). Retrieved December 13, 2021, from https://www.mathworks.com/help/deeplearning/ref/resnet50.html; jsessionid=1357daa05814956906709900a82e.

[5] ImageNet. http://www.image-net.org