SIGNIFICANCE OF DATA QUANTITY ON MODEL CENTRIC AI FOR A GARBAGE CLASSIFIER USE CASE

Wenyuan Zhong, Rohinesh Ram, Yutong Wang, Emily Wang, Yingcheng Kou, Michael Ah-Kiow

University of Calgary

ABSTRACT

The purpose of this paper is to explore the significance of convolutional neural networks (CNN) models on a garbage system using different images and classifying them into different coloured garbage bins. The three different datasets being compared is firstly, the dataset obtained from class, secondly, the dataset obtained from class plus additional data obtained from Kaggle, and lastly, the dataset from class plus data obtained from Kaggle, and 100 images manually collected by the team from various sources. We also used three different models on the dataset: VGG16, EfficientNet, and the teams own CNN. The best performing model was the EfficientNetB0 model with transfer learning for an accuracy of 0.819. It performed the best since it leverages information gained from the ImageNet dataset which contains over one-million images. This model was then used to be trained and tested on datasets of increasing size. There were three different datasets being compared. Firstly, is the dataset obtained from class which was a total of 5931 images. Secondly, it was the dataset obtained from class plus additional data obtained from Kaggle which contained a total of 7326 images. Lastly, the dataset from class plus data obtained from Kaggle, and images manually collected by the team from various sources which resulted in a dataset of a total of 7626 images. The total accuracy increased 6% from the first dataset at 0.797 to 0.858 in the last dataset. The results ultimately align with the theoretical assumption since with increasing data size, the more accurate the model and the better the performance in deep learning. In conclusion, the best performance overall was achieved by the EfficientNetB0 model with the full dataset.

1. INTRODUCTION

In recent years, machine learning and more specifically deep learning have become quintessential for innovation. AI is now used for natural language processing, computer vision and finding complex correlation [1]. Furthermore, with the rise of new sensors and algorithms, AI is on its way to achieving complex tasks that were initially thought impossible (i.e., self-driving).

However, as Melanie Mitchell points out in her book Artificial Intelligence, A Guide for thinking Human, AI has had its fair share of struggles. For instance, AI has gone through 2 bust famously known as AI winters [2]. The most recent being due to lack of processing power and data. However, with the rise of big data and faster processing powers, those limitations are now a thing of the past. Every day, there's more questions about using AI to leverage the untapped potential of big data and finding hidden correlation to solve complex problems.

Those factors have allowed machine learning and deep learning in becoming one of the hottest fields in recent years. A fairly common problem that is addressed by deep learning is computer vision. Through this paper, our team wants to investigate the performance of various models using a garbage classification system with convolutional neural net. Furthermore, our team will be using this experiment to investigate the correlation between quantity of data and model performance.

2. PROPOSAL AND METHODOLOGY

Through the development of this paper, the group is trying to address 2 following questions

- How do different models perform with a garbage classification use case?
- The effectiveness of data for model centric AI [3].

How different models perform for any use case is one of the common questions that occur on machine learning or AI engineering topics. In most cases, the goal for engineers is to find the optimal model for each use case and ensure it is ready for production. Throughout the academic semester in ENEL 645, the group has been evolved with various models such as convolutional neural networks, Efficient Net, VGG 16, VGG16 with trainable weights, models with dropouts and et cetera. Being equipped with many tools in our toolbox, the group aims to innovate the garbage classification problem with a model centric approach to find which models perform best. This has 2 goals, one to allow us to get more familiar with each model and gain a better understanding on the trade-offs between each model and how it affects model performance.

The problem or use case we are looking to solve is garbage classification. Household garbage in Calgary is usually categorized into three different bins including:

1. Green bin for food and yard waste
2. Blue bin for recycling
3. Black bin for garbage

At landfills, garbage is usually manually filtered by workers, the goal of this project is to develop and compare the performance models when it comes to classifying garbage into three different bins to help automate this process.
The goal in this case will be to compare the performance of different models in correctly predicting the bin color for garbage objects belonging to three different datasets.

Figure 1 below shows a list of models of interest that the group will be going through.

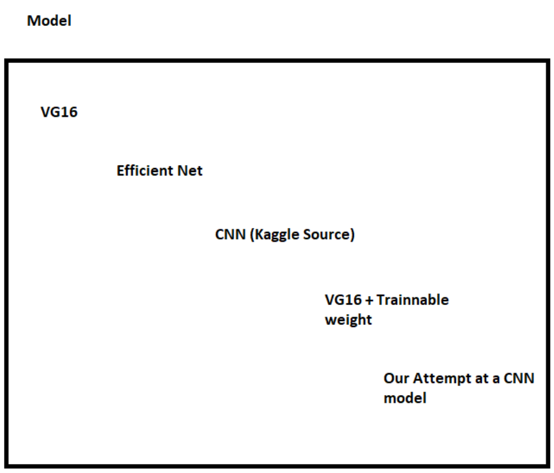

Figure 1: List of models of interest

The list of models utilized will include a combination of tutorial models, Kaggle models and our own CNN model. Refer to section 5 for more information related to each model.
On top of comparing different models, we are also looking to focus on experimenting with data. The group is looking to see if the data provided will have an effect on the performance of each model, to validate this, the group has collected 3 different sets of data, including:

1. Garbage classification data set from class
2. Garbage classification dataset from Kaggle
3. Manually collected garbage classification dataset

The group is looking to utilize these sets of data by adding the datasets cumulatively into 3 different aggregates as following:

1. Dataset from class on its own
2. Dataset from class + dataset from Kaggle
3. Dataset from class + dataset from Kaggle + manually collected dataset

The group will ultimately visualize the performance of models on each dataset and determine if more data correlates to better performance for our use case as well as whether the group was able to develop a convolutional neural network which demonstrated superior performance compared to the rest of the models on the list.

3. MOTIVATION AND SIGNIFICANCE

The intent behind this experiment is for 2 primary reasons. The first being to gather all of the theoretical knowledge and models explored through the semester and apply them into a practical and all-in one-use case for garbage classification. It's a complex enough scenario where the need of deep learning is necessary yet it is simple enough to deploy different models and understand the end-to-end pipeline. Despite the constant dynamic and rapid changes of AI throughout history, model centric AI is still regarded as the bread and butter of machine learning. Like Melanie Mitchell points out in her book [1] , the software and algorithms used in AI has barely change in recent years. As such the group hopes to gain relevant and hands on experience with model centric AI and applying all the cool models explored throughout ENEL 645.

Another significant intent behind this experiment is to gain a deeper understanding of the effectiveness of data. With the rise of data centric AI and other non-model centric approach, the group believe it's important to truly understand model centric before moving on to other approaches. We hope to investigate how adding data, some good, some bad data and how it affects the performance of our models (if any).

The significance of data centric and model centric AI is paramount. It's currently one of the hottest questions being ask in industry. Which is more important, cleaner data or more data? The general trend of AI has always been that with more data and with more processing power, deeper and complex models could be trained and deployed to solve problems that otherwise wouldn't be solved. However, as AI is being democratized, it has become painfully obvious that not every company can afford to deploy a big data engine and train such expensive neural network.

Andrew Ng, a key figure in AI says it best in his interview with Landing AI. "In many industries where giant data sets simply don't exist, I think the focus has to shift from big data to good data. Having 50 thoughtfully engineered examples can be sufficient to explain to the neural network what you want it to learn." [4]. This capstone project (especially part 2) will serve as a small experiment on how amount and quality of data can affect performance of an AI. Like proposed in the methodology, we'll be adding layers of data, some with good and some with bad quality and see how it affects performance. We hope that the result should clearly show whether data quality or data quantity was more favorable for our small garbage data collection classification use case. Ultimately it doesn't mean we hope to prove that one is better than the other, but to point that based on the data we collected, the experiment we set and the quantity of data, there was a clear better answer.

4. RELATED WORK

4.1. The significance of data

Scaling to very very large corpora for natural language disambiguation is one of the most significant and popular paper in the field of Artificial Intelligence, Its findings led to the revolution large data collections for artificial intelligence [5]. It was written by Michele Banko and Erik Brill and was released by Microsoft [5]. The study focused on implementing several models on a natural language problem called disambiguation. The paper speaks of how effective data is in comparison to models. Their result shows that adapt more complex or "better" models had less significant impact than adding more data [3]. Their findings showed that data is the most fundamental driving force for AI performance and that models aren't as significant relatively speaking. This philosophy is considered one of the most fundamental ideologies in artificial intelligence.

Figure 2 below is a plot from [5], that shows how significant the model improvement as more data is added to each model. As shown, initially most models have similar performance despite different complexities. And that as more data is added even simpler models can outperform complex models (with less data). For instance, it's clear that even a Naïve bayes can outperform a memory-based model just by adding more data. In fact, once there were a billion words, all the models had similar performance.

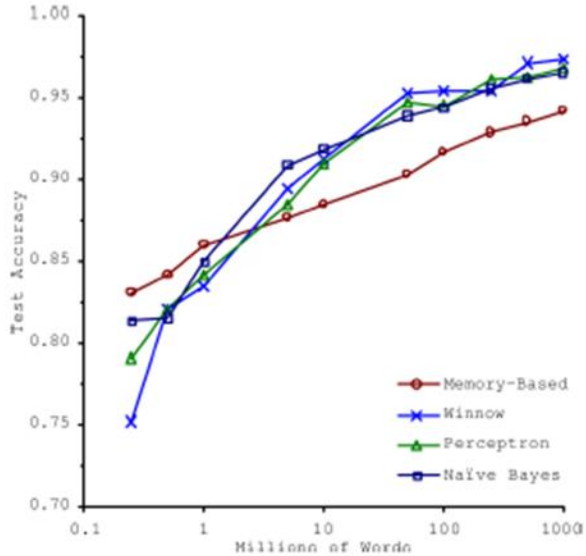

Figure 2: Data Quantity vs Model Comparison for a NLP use case [5]

5. MATERIALS AND METHODS

5.1 Data

5.1.1 ENEL645 Data

This data was provided by the professor containing 5391 images in total. The data was originally used for illustrating the process of transfer learning by using a pre-trained model on ImageNet for one of the tutorials. The team used this data to compare the results with 2 other datasets. For the preprocessing step, image augmentation was performed. The main method for image augmentation is the ImageDataGenerator, where upon fitting an image, it will slightly modify the image based on the provided parameters. The shortcoming with this dataset was that the 5391 images were further categorized into different bin colors to classify the trash, and that is not a large dataset for deep learning to achieve accurate results.

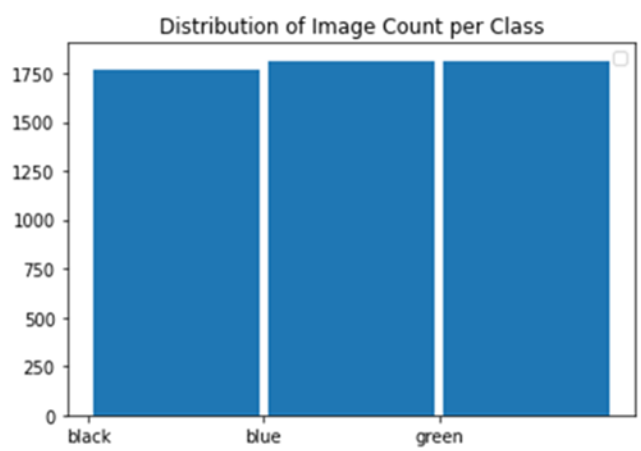

Figure 3: ENEL645 Data distribution

5.1.2 Kaggle Data

The Kaggle dataset is obtained from a Kaggle activity for a garbage classification problem where twelve different classes of household garbage, paper, cardboard, biological, metal, plastic, green-glass, brown-glass, white-glass, clothes, shoes, batteries, and trash are classified. The dataset has 15,150 images in total with size of 250MB. The initial breakdown for the number of images for each class is shown in the Table 1 below. As shown it's obvious that the distribution of each class is not balanced. The twelve classes are then organized into three bins (green, blue and black) to match our desired format for dataset. The dataset can be found at [6].

Table 1: Kaggle dataset breakdown

Class name	Number of images
Batteries	945
Biological	985
Brown-glass	607
Cardboard	891
Clothes	5325
Green-glass	629
Metal	769
Paper	1050
Plastic	865
Shoes	1977
Trash	697
White-glass	775

5.1.3 Our own Data and Full Dataset

For our own data, we manually added 100 images for each bin from google and it has a size of 35 MB. This dataset was added to the data given by the professor and Kaggle data to form a full dataset with size of 1.2 GB. This full dataset was used to test the performance of our best model to prove that with increasing amounts of data the performance also increases.

5.2. Models

5.2.1 VGG16

VGG16 is a convolutional neural network (CNN) architecture used for visual object recognition. It won the ImageNet project competition(ILSVRC) in 2014 and it's one of the most outstanding vision model architectures ever created [7]. The 16 in VGG16 means that it comprises 16 layers, each of which has a different weight. The most distinctive feature of VGG16 is that, rather than having a huge number of hyper-parameters, they concentrated on having 3x3 filter convolution layers with a stride 1 and always utilized the same padding and maxpool layer of 2x2 filter with stride 2. Throughout the design, it maintains this convolution and maxpool layer layout. Finally, it has two fully connected layers(FC) for output, followed by a softmax layer. Figure 3 below shows the architecture structure of VGG 16.

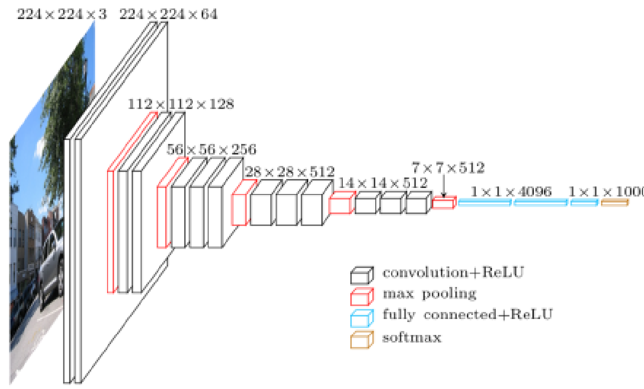

Figure 3: VGG16 architecture [7]

The inputs are fixed-size 224 by 224 images with three channels (RGB). The images are processed through a stack of 2 convolutional layers with filters of size 3x3 followed by ReLu activations. There are 64 filters in each of these two levels. The padding is 1 pixel, while the convolution stride is fixed at 1 pixel. Then they are fed to a maxpool layer with size of 2x2 and stride of 2 pixels. The activations after the first stack of layers have a size of 112x112x64, which is half of the original. Then these activations flow through similar stacks of layers followed by a softmax activation layer at the end for categorical classification [7].

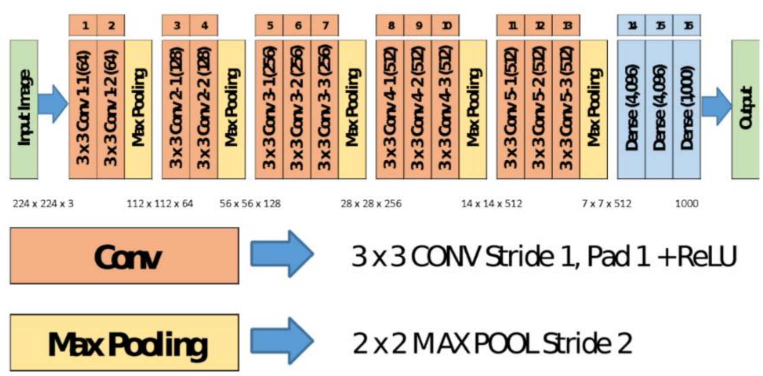

Figure 4 VGG16 layers

The advantages of VGG16:

1. Simple to understand and explainable.
2. Results are promising with score of 92.7% on the ImageNet dataset, which is about 150 GB of images.
3. Works great on classical problems like cats vs dogs classification and can achieve a great enough baseline of 80% accuracy.

The disadvantages of VGG16:

1. The model is big due to the enormous number of weights parameters, which are close to 550MB in size.
2. VGG-16 contains just 16 layers, but if we build it deeper, like VGG-19, the network gets deeper, the model gets heavier, and inference time increases.

5.2.2 EfficientNet

EfficientNet is a convolutional neural network (CNN) that enforces uniform scaling in dimensions of width, depth, and resolution using a compound coefficient which can ultimately lead to better performance. It is known to be one of the most powerful CNN architectures. [10] introduces this model as a simple but also extremely powerful compound scaling method.

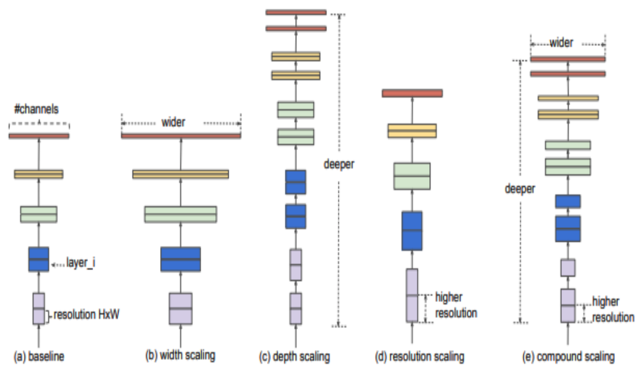

Figure 5 Types of Model Scaling

The compound scaling method as seen above in Figure 5 can be done by increasing the depth by α^N, width by β^N, and image size by γ^N. α, β, γ are constant coefficients that can be obtained from a grid search on the original model. The figure shows the difference between this compound scaling method in comparison to the conventional methods. Compound scaling is based on the log that if an input image is larger, the network requires more layers and channels to not only increase the receptive fields, but also capture the fine-grained patterns. [10]

To obtain EfficientNets which is a group of deep learning models, a neural architecture search was used to outline a base network which was then scaled up to obtain the result. The neural architecture search helps to automate the design of neural networks to improve efficiency based on floating-point operations per second. As a result, it was seen to achieve better accuracy than previously designed convolutional neural networks.

The EfficientNet model was originally proposed by two researchers of Google Research in the Brain Team, MingXing Tan and Quoc V. Le. With the increase in popularity of computer vision applications, researchers have been eager to optimize and improve performance in this field. EfficientNet is one of the convolution neural networks that are widely accepted in this field of study[10].

EfficientNet is currently one of the most accurate models in the world of CNN's; It not only has high accuracy accuracy, but also great efficiency in comparison to existing CNN models.

5.2.3 Personal CNN

A convolutional neural network (CNN) is an artificial neural network used in images to process pixel data. They utilize multiple filters to determine specific image features that can be used in object categorization. There are multiple kernels responsible for extracting features. A kernel refers to a method that allows the application of linear classifiers to inherently non-linear problems to transform them into a higher-dimensional space. Scientifically, CNN's are structured like the frontal lobe in the brain where image and visual processing occurs in humans. Neurons are arranged to account for the entirety of the visual field which eliminates the problem of piecemeal image processing found in traditional neural networks. CNN layers consist of input, output, and hidden layers that include multiple convolutional layers, pooling layers, normalization layers, as well as fully connected layers. The applications that often require CNN include image processing, machine vision (image and video recognition) and natural language processing. [11]

Our CNN is designed according to the principles recommended by [12] and referencing the architecture of vgg16. The filter size is set to be 3 to pick up information rigorously at the first several layers and increases to 5 as more layers are added. The number of channels starts at 64 to learn more "low-level features", this number is then increased to 128 and 256 in the following layers. Max-pooling is added after each convolutional layer, the pooling size is 2 considering the input images sizes are not very large. Dropout is applied before number of filters increases, a relatively low dropout rate 0.1 is used, we found this layout can reduce overfitting down to 0.5%-3%. Multiple versions of CNNs were created with minor variations on details. They were trained and tested on ProfsData with 30 epochs, the accuracies were approximately between 0.65-0.7. We chose the one that has the highest accuracy of 0.7048, the following table is the summary of the model.

5.3. Implementation

The methodology and proposal were implemented using the following sequence of actions:

1. Data Aggregation and Composition.

2. Model Formation.
3. Train Test Split.
4. Image Augmentation/Preprocessing.
5. Training & Collecting Results: See Section 7 for more information.

5.3.1 Data Aggregation and Composition

In this section, the different data sources were first collected and then aggregated into different “datasets” for training, validation, and testing. The idea is to have a small and clean data as baseline, larger aggregated and still relatively clean dataset, and the largest dataset with some noisy data. Note, clean in this case means data provided in lecture or with Kaggle. Unclean data is a subjective term used to refer to data collected by the group found on Google. They are of lower quality, noisy and sometimes not isolated images meaning that there is background noise or other items present in the image. Table 2 below shows the dataset available to us and a brief description of what they consist of while Figure 6 shows an illustration of the aggregation.

Table 2: Dataset & Composition

Dataset	**Dataset Composition**	**Number of Images**
Our data	Data collected by us	300
Kaggle data	Kaggle Data	1936
ProfsData	Prof Data	5931
Profs_Kaggle	Prof Data + Kaggle Data	7326
Full	Prof Data + Kaggle Data + Our data	7626

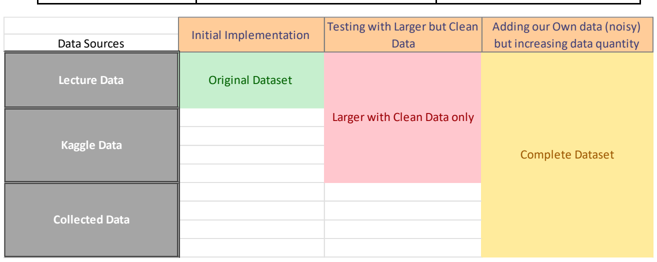

Figure 6: Data Aggregation

5.3.2. Model Implementation

Table 3 below shows a summary of the models employed as well as their implementation details. Refer to the attached source code for the specific step.

Table 3: Model Implementations and Summary

Model	Implementation/Layer Details
CNN(best)	• 9 Convolutional Layers in total • 9 Max Pooling in total • 3 Dropout in total • Pattern: Conv+Max+Conv+Max+Cov+Max+Dropout • 3 Dense Layers: 2 relu + 1 softmax • Total params: 22, 387, 715 • Non-trainable params: 0
VGG16	• Import VGG16 model • Include top • Exclude external weights • Set classes to be 3 • Created generators • Created callbacks • Train model
EfficientnetB0	• Import EfficientnetB0 model • Include top • Exclude external weights • Set classes to be 3 • Create generators • Create callbacks • Train model
VGG16 + Transfer Learning	• Import VGG16 model • VGG16 Classifier is excluded • Initial weights were imported from Imagenet • Set weights to be untrainable • Create generators • Create callbacks • Added our own dense layer • Trained model with low learning rate • Save model • Set weights to be trainable • Reduce learning rate • Train model
EfficientnetB0 + Transfer Learning	• Import EfficientnetB0 model • EfficientnetB0 Classifier is excluded • Initial weights were imported from Imagenet • Set weights to be untrainable • Create generators • Create callbacks • Added our own dense layer • Trained model with low learning • Save model • Set weights to be trainable • Reduce learning rate • Train model

5.3.3. Train test split

The next step consisted of splitting the dataset into a training, validation, and test set. The training set was used to train the model while the validation was used to help verify and improve performance during the training. Finally, the test set was hidden until the very end to verify the performance of the models. Each dataset listed in 5.1 is split into train, validation, and test sets at a ratio of 0.7, 0.2 and 0.1. Every split set contains three folders of black, blue, and green.

5.3.4 Image Augmentation/Preprocessing

Image augmentation was used to preprocess all images that will later be used for model training, verification, and testing. The augmentation process was based on the parameters introduced in ENEL645 tranfer_learning_imagenet.ipynb to ensure the same standards are applied. For VGG16 and EfficientnetB0, images undergo additional processing steps provided by Kera.

6. RESULTS & DISCUSSION

6.1. Training Model Architectures on ENEL645 Data

Figure 7 shows the performance of the five model architectures on the ENEL645 garbage dataset. The best performing model was the EfficientNetB0 model with transfer learning at an accuracy of 0.819 This model performs best as it leverages knowledge gained from training on the ImageNet dataset which contains over one-million images. The VGG16 model with transfer learning has the next accuracy at 0.749 which indicates that transfer learning serves to effectively increase performance. Refer to the code for training and validation accuracy plots for all models.

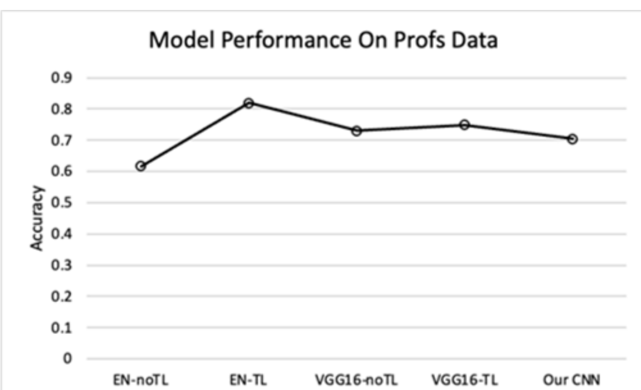

Figure 7: Plot of different model architectures performance on the ENEL645 data set

6.2. Model Accuracy on Increasing Dataset Sizes

Figure 8 shows a plot of the performance of the EfficientNetB0 model with transfer learning vs increasing dataset sizes. The ENB0 model with transfer learning was chosen for this step because it exhibited the best performance in the previous sections. The model was trained and tested on datasets of increasing size. The first dataset was the ENEL645 garbage classification data containing 5931 images. The next dataset contained all the images from the first dataset and additional data from Kaggle, the total size of the second dataset was 7326 images. The final dataset contained all the images from the second dataset plus data collected by the team; the final dataset contained 7626 images. The models performance increased 6% from the first data set at 0.797 accuracy to 0.858 in the final data set. This aligns with the thesis of the paper that adding more data, despite the noisy and non-clean data collected by the team, improves model performance.

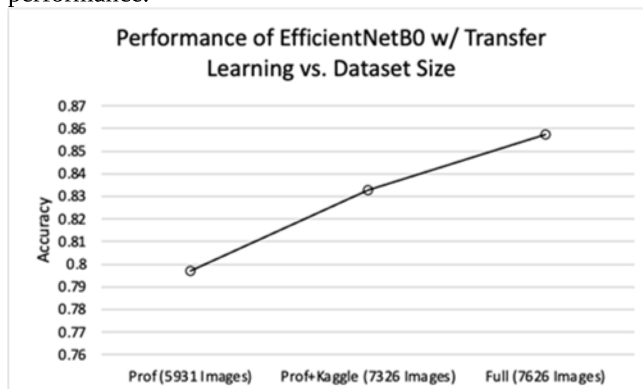

Figure 8: Plot of the EfficientNetB0 model with transfer learning as data size increases

6.3. Performance of Model Architectures on Different Dataset Sizes

To extend the ideas discussed in previous sections, the team tested different datasets on all five of the models and the results of this experiment are summarized in Figure 9. In general, the models all performed better as more data was added except for the VGG16 model with and without transfer learning which performed the best on the second data set consisting of all the images from the ENEL645 garbage classification data and all the images from the Kaggle dataset. This indicates that the VGG16 architecture may be sensitive to noisy data because of the final dataset including data collected by the group. The best performance overall is achieved by the EfficientNetB0 model with transfer learning on the full dataset.

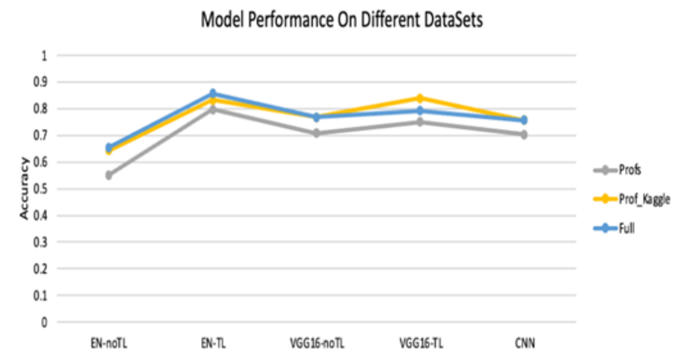

Figure 9: Plot of all model performances on different dataset

6.4 Results

All results discussed in sections 6.1, 6.2, and 6.3 are summarized in Table 4.

Table 4: Experiment Results

Model	Data Set	Performance (Accuracy %)
CNN(best)	Dataset #1 (Small)	0.703
	Dataset #2 (Medium)	0.756
	Dataset #3 (Large)	0.757
VGG16	Dataset #1 (Small)	0.708
	Dataset #2 (Medium)	0.769
	Dataset #3 (Large)	0.769
EfficientnetB0	Dataset #1 (Small)	0.552
	Dataset #2 (Medium)	0.644
	Dataset #3 (Large)	0.655
VGG16 + Transfer Learning	Dataset #1 (Small)	0.751
	Dataset #2 (Medium)	0.840
	Dataset #3 (Large)	0.769
EfficientnetB0 + Transfer Learning	Dataset #1 (Small)	0.797
	Dataset #2 (Medium)	0.833
	Dataset #3 (Large)	0.857

7. CONCLUSION

The purpose of this project is to understand the significance of convolutional neural networks and build our own CNN model from scratch on a garbage classification problem. The three classes are black for landfill garbage, blue for recyclables and green for organic wastes. Through the development of the project, two additional questions are brought up. The first one is how do different models perform with a garbage classification use case and the second one is the effectiveness of data for model centric AI. The second question helps the group to gain a deeper understanding of the effectiveness of data two different aspects, which are cleaner data vs. more data. There are three different datasets, and they are combined to make a full dataset. The first dataset is from the professor and the data is intended to illustrate transfer learning using a pre-trained model on ImageNet for one of the lessons. The second dataset is from a garbage classification problem on Kaggle, and the third dataset is three hundred garbage images from google. The datasets are preprocessed using image augmentation before model training. To determine the best model for this garbage classification use case, five different models are implemented and tested using the dataset provided by professor. The models are our best version of CNN model, VGG16, EfficientnetB0, VGG16 with transfer learning and EfficientnetB0 with transfer learning, which is the best model with accuracy of 0.819. Since it 's trained with data from the ImageNet dataset, which comprises over a million images. This model is then trained and tested on bigger datasets to solve the problem whether it's more important to have high quality data or more data. The dataset given by professor is combined with the kaggle dataset for training. The data given by professor is then combined with the kaggle data and finally with the image dataset from google. The result is that larger dataset produces higher accuracies than dataset with better quality since the model's performance increased 6%. Overall, the best model is EfficientnetB0 with transfer learning with the largest (full) dataset and the accuracy is 0.857.

8. REFERENCES

[1] M. Mitchell, *Artificial Intelligence: A Guide for Thinking Humans*.

[2] "Are we in an AI summer or AI winter?," *Big Think*. https://bigthink.com/13-8/are-we-in-an-ai-summer-or-ai-winter/ (accessed Apr. 02, 2022).

[3] S. Singh, "Importance of Data vs Algorithm in Machine Learning," *Deep Learning University*, Mar. 28, 2020. https://deeplearninguniversity.com/importance-of-data-vs-algorithm-in-machine-learning/ (accessed Apr. 02, 2022).

[4] "Andrew Ng: Farewell, Big Data," *IEEE Spectrum*, Feb. 09, 2022. https://spectrum.ieee.org/andrew-ng-data-centric-ai (accessed Apr. 02, 2022).

[5] M. Banko and E. Brill, "Scaling to very very large corpora for natural language disambiguation," in *Proceedings of the 39th Annual Meeting on Association for Computational Linguistics*, USA, Jul. 2001, pp. 26–33. doi: 10.3115/1073012.1073017.

[6] "Garbage Classification (12 classes)." https://www.kaggle.com/dataset/126ab2c7f7e22add276bc29e44b97f635e3f6a04368afb20130a83518a9056b9 (accessed Apr. 02, 2022).

[7] R. Thakur, "Step by step VGG16 implementation in Keras for beginners," *Medium*, Nov. 24, 2020. https://towardsdatascience.com/step-by-step-vgg16-implementation-in-keras-for-beginners-a833c686ae6c (accessed Apr. 02, 2022).

[8] G. L. Team, "Introduction to VGG16 | What is VGG16?," *GreatLearning Blog: Free Resources what Matters to shape your Career!*, Oct. 01, 2021. https://www.mygreatlearning.com/blog/introduction-to-vgg16/ (accessed Apr. 02, 2022).

[9] "VGG16: A Brief Summary | Data Science and Machine Learning." https://www.kaggle.com/getting-started/a (accessed Apr. 02, 2022).

[10] M. Tan and Q. V. Le, "EfficientNet: Rethinking Model Scaling for Convolutional Neural Networks," *ArXiv190511946 Cs Stat*, Sep. 2020, Accessed: Apr. 02, 2022. [Online]. Available: http://arxiv.org/abs/1905.11946

[11] "What is convolutional neural network? - Definition from WhatIs.com," *SearchEnterpriseAI*. https://www.techtarget.com/searchenterpriseai/definition/convolutional-neural-network (accessed Apr. 02, 2022).