Automated Retail Product Checkout

Abstract

In the present days, the automated checkout (ACO) is booming in retail company sectors. At this time, we implement Convolution Neural Network (CNN) models to pre-train the images to automatically categorize the products. Henceforth, it reduces manual work so that the retailer can utilize the labor force on things that matter. To achieve this real-time demand, in our project, we will discuss the relevant theory and evaluate the performance of the existing datasets like the Vegetable Image dataset, Freiburg groceries dataset, and Grocery store image dataset by implying three classification models for each using robust CNN architectures like Resnet18, Googlenet, and Alexnet. We also performed image augmentation techniques where input images are transformed using different transformations like horizontal flip, vertical flip, rotation, and normalization to increase the number of images passed through the network, and then feature extraction, map pooling, non-linear functionality, batch normalization are performed to get an output from a CNN architecture. Finally, we have used Adam Optimiser, a combination of RMS Prop and SGD momentum, to back-propagate the error gradients. Overall, our project proposes multiple classification techniques for the extensive training of varying and complex datasets to attain promising classification results with a relatively high degree of accuracy. Moreover, during our project, we dealt with various challenges such as training data, Imbalanced Data, Interpretability of data, Overfitting, Vanishing gradient problem, and this report further details how we tackled those problems in exploring this field.

1. Introduction

Recently, machine learning (ML) has gained wide popularity in research and is being integrated into various applications such as text mining, spam detection, video recommendation, image classification, and multimedia concept search. Among various ML algorithms, deep learning (DL) is very widely used in these applications.

Image classification is supervised learning in which images are tagged with classes from a predefined set of classes. Due to the effectiveness of deep learning models for image classification, especially convolutional neural networks, researchers have achieved error rates much lower than those of humans. H. 2.251 percent. A convolutional neural network (CNN) is a deep neural network defined with special operations called convolution operations in which a kernel is convolved over an input to generate a feature map.

1.1. Problem Statement and its importance in current field of applications

In the present days, the automated checkout (ACO) is booming in retail company sectors such as Amazon GO [9]. So we wanted to explore this field as the data requirements may be changing with additional new products coming in the market very frequently. Therefore, we felt that this domain would help us explore different CNN algorithms and deep learning techniques because of various datasets that are readily available and also because of its recent innovation that is yet to generate an impact on the market. Henceforth, we implemented ConvolutionNeuralNetwork (CNN) models to pre-train the images from 3 different datasets in an attempt to automatically categorise the products. Because, as previously quoted it not only reduces the manual work but also can effectively track all the products being sold to the customers which makes tasks easier, efficient and convenient for retailers. In this report, we will be talking about various techniques and evaluation metrics to compare the performance of different models to determine the best approach in solving this interesting problem. Through this process, we dealt with varying illumination of the images and difficulty in collecting training images due to frequent product updates, which needed to be addressed in order to prevent bias. Deep CNNs have proven to perform exceptionally well in classifying images; thus, it seems to be an excellent fit for resolving this problem.

1.2. Literature Review

There is some research on automating the checkout system in the retail industry. These studies consider the topic from different perspectives. One of them proposes attaching

Radio Frequency Identification (RFID) tags to monitor the quantity of products on shelves [12], but this approach requires implementing the technology and integrating it into existing systems and is not cost-effective to use. Some of them applied traditional image processing techniques to detect the presence or absence of products, and some used deep learning approaches for object detection on shelves. Examining all this previous work, these approaches have advantages and disadvantages. When conventional image processing methods are used, Histograms of Directed Gradients (HoG) [6] for feature extraction and Support Vector Machines (SVM) for classifiers have limited performance even on large datasets and are difficult to improve. Additionally, visual similarities between different products of the same brand can lead to misclassification. On the other hand, deep learning (DL) approaches such as convolutional neural networks (CNN) can be used to achieve high levels of accuracy. Our report proposes a classification technique for the extensive training of varying and complex datasets to attain promising classification results with relatively high degree of accuracy. [17]

The performance of the existing datasets such as Vegetable Image dataset, Freiburg groceries dataset and Grocery store image dataset are evaluated by implying three classification models for each. The classification model architectures we have chosen are Restnet18, Googlenet, and AlexNet. We will discuss in detail describing the concepts, theory, and state-of-the-art architectures. Our contributions are outlined as follows:

This is the first review that looks at CNN models in near-detail. The most popular deep learning algorithm, CNN, is explained in detail through exploring this retail sector while explaining its concept, theory, and state-of-the-art architecture. We review current challenges (limitations) of Deep Learning including lack of training data, Imbalanced Data, Interpretability of data, Overfitting, Vanishing gradient problem.

2. Methodologies

2.1. Datasets

We have chosen 3 datasets which are in the same domain space. You can view the specifications of these three datasets in Table 1. We performed various data augmentation and transformation techniques such as Centre Crop, Resize,Random Rotation, Random Horizontal flip, etc on our datasets to increase the number of training examples so that our models can learn better. To visualize our datasets we used TSNE Data Visualization technique [8], for example let us see the tsne distribution for vegetable images dataset, which is our best dataset. Moreover, we also implemented transfer Learning on Freiburg dataset using Resnet and Googlenet pretrained weights to observe any key differences because the yielded results earlier without using pretrained weights was not satisfactory.

Dataset Specifications	1	2	3
No of classes	25	39	15
No of images in Training set	4749	2640	15000
No of images in validation set	-	672	3000
No of images in testing set	-	2485	3000
Image Resolution	256x256	348x348	224x224
Image Format	JPEG	JPEG	JPEG
Image type	RGB	RGB	RGB

Table 1. Specifications of all three datasets(numbered in table as follows 1.Freiburg Groceries Dataset 2.Grocery Store Dataset 3.Vegetable Images Dataset)

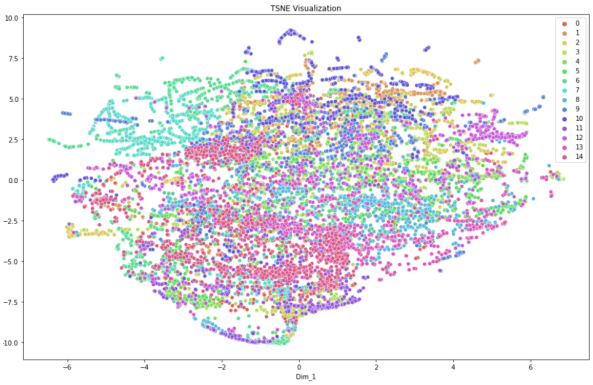

Figure 1. TSNE on Vegetable Images Dataset

Freiburg Groceries Dataset: The dataset consists of 4749 images covering 25 different classes of groceries (such as beans, cakes, candy, coffee, corn, etc), with at least 97 images per class and the image resolution is about 256 x256px with all the images having JPEG format. We have to split the dataset into train, test and validation as we don't have any predefined paths and images separately. So we splitted the data into train ,test and val with 60 percent,20 percent and 20 percent respectively. We collected this dataset from a github repository owned by PhilJd. Please refer to figure 2 for a generalized view into this dataset. [3]

Figure 2. A Generalized View into freiburg dataset and its classes

Grocery Store dataset: This dataset was collected from a github repository owned by Marcus Klasson. All the images are taken using a smartphone camera.The dataset contains 5125 natural images with 39 classes of fruits, vegetables, and carton items (e.g. juice, milk, yogurt) . As we already had images for train, test and validation separately, we didn't perform any splitting of the dataset. So there were around 2640 images under the Train set, 2485 images under the Test set, and about 672 images under the Validation set. The resolution of the image is 348x348 and the image format is JPEG. Please refer to figure 3 for a generalized view into this dataset. [5, 20]

Figure 3. A Generalized View into Grocery Store Dataset

Vegetable image dataset: In this dataset there are 21000 images from 15 classes, where each class contains a total of 1400 images. Each class has an equal proportion and image resolution of 224×224 and the image format is JPEG. The dataset was already distributed into 70 percent(approx.) around 15000 images for training and 15percent(approx.) 3000 images for testing, and the rest 15percent(approx.) 3000 images for validation.This dataset is collected from kaggle. Please refer to figure 4 for a generalized view into this dataset. [18]

Figure 4. A Generalized View into Vegetable Image Dataset

2.2. CNN Models

We chose unique architectures such as Resnet18, Googlenet and Alexnet because of their robust and unique architectures and wide range of applications which works astonishingly well for our use case since we are dealing with a little imbalanced data and high number of features. In the case of AlexNet we choose this model inorder to understand and compare how a robust model fair well with AlexNet and the results we observed are as expected. We utilized the readily available architectures from the pytorch library.

ResNet: It has the robust architecture when compared to other models in CNN. In ResNet models, all convolutional layers apply the same convolutional window of size 3×3, the number of filters increases following the depth of networks, from 64 to 512. At the end of all models, the average pooling layer is applied to replace fully connected layers. This replacement has some advantages. Firstly, there is no parameter to optimize in this layer, hence it helps to reduce the model complexity. Secondly, this layer is more native to enforce the correspondences between feature maps and categories [10]. As we have a moderate amount of data, we implemented ResNet18 implementation which contains 18 convolutional layers. The another reason for choosing this model is because of its identity shortcut connection that is skip connections which can eliminate vanishing gradient issue which we faced earlier while training with alexnet architecture. [10, 11, 15]

GoogleNet: Researchers discovered that an increase of layers within a network led to a significant performance gain. But this has also few disadvantages which came at

a cost. Large networks are prone to overfitting and suffer from either exploding or vanishing gradient problem.The GoogLeNet architecture solved most of the problems that large networks faced, mainly through the Inception module's utilisation. [7] The main reason for selecting this architecture is it performs well by reducing the size of input image by retaining most useful spatial information. This makes it a powerful model with increased computational power. It contains 22 convolutional layers and 7 pooling layers but the number of parameters are reduced from 60 million (AlexNet) to 4 million. To prevent overfitting, GoogleNet uses auxiliary classifiers which can avoid vanishing gradient problems by adding it to intermediate layers namely third and sixth layers. Therefore, this fits well with our data requirements and as previously stated in the introduction part we are solving the problem of vanishin gradient problem with help of these sturdy architectures that help our model's gradient reach global minima [4]

AlexNet: As previosuly stated, the main reason why we chose AlexNet is that it was the first model to be developed in previous decades where it had a basic network of layers of what we have today. So we wanted to compare the performance of this model with respect to more robust architectures such as Resnet18 and GoogleNet. Talking about the architecture, AlexNet has a total of eight layers in which it contains 5 convolutional layers and 3 fully connected layers which use ReLU as a non-linear activation function. [1, 2]

2.3. Optimization Algorithm

For optimization we used Adam optimizer for all of our models. The reason we chose Adam over other available optimizers is that as we learned in the course work that Adam is a combination of RMSprop and SGD Momentum as well as the results of the Adam optimizer are generally better than every other optimization algorithms, have faster computation time, and require fewer parameters for training. We used performance metrics such as confusion matrix, precision, recall and f1 score to evaluate the optimized models. Another reason for choosing this optimizer is, it performs very well with a model which has a large number of learnable parameters. As we are using complex algorithms, optimization algorithms should be powerful in order to get better performance. Because of Adam, it required less tuning of the hyperparameters to achieve expected outputs and also as it uses less memory and time, it helped us in controlling the resources used by the model. [19]

3. Results

3.1. Experimental Setup

All experiments were conducted in Windows11 with Intel CPU @ 3.50GHz; Nvidia GeForce 630M 2GB, Google Colab and Kaggle notebooks. The performance is measured in terms of training accuracy and evaluation accuracy on the considered dataset. The experimental results are conducted on the three datasets namely, Vegetable Image dataset, Freiburg groceries dataset and Grocery store image dataset. These datasets have been chosen to evaluate the performance of models for image classification.To evaluate the performance of the proposed pre-trained models with transfer learning, a fixed partitioning scheme has been chosen. The freiburg dataset is partitioned into training, test and validation sets using a ratio of 60:20:20.

3.2. Main Results

After implementing all the deep CNN models ResNet, Googlenet, and Alexnet it can be concluded that the ResNet models provided superior performance followed by Googlenet by slightest of margins, whereas in case of Alexnet as expected the performance was low where the model is getting stuck at local minimas even after changing and applying hyperparameters, balancing out data and data augmentation techniques. Overall, Vegetables dataset gave good results when compared with other two datasets because from table 1 we can observe that it has more number of learning examples as well as from tsne visualization the data is well balanced. The evaluation metrics that are used for comparing the models' performances were the precision score, the recall/sensitivity score, the F1 score, confusion matrix, and accuracy. The results are summarised in the following Table images respectively for each of the dataset with the representation of the evaluation metric scores.

Vegetable Images dataset	Resnet from scratch	Googlenet from scratch	AlexNet from scratch
Accuracy	0.99	0.98	0.94
Precision	0.99	0.98	0.94
Recall	0.99	0.98	0.94
F1- Score	0.99	0.98	0.93
Train Accuracy	99.13%	98.13%	97.23%
Test Accuracy	99.1%	97.4%	92.86%
Validation Accuracy	98.37%	97.67%	92.74%

Figure 5. Comparision between performance metrics of each model on Vegetable Images Dataset

Grocery Dataset Observations	Resnet from scratch	Googlenet from scratch	AlexNet from scratch
Accuracy	0.75	0.72	0.73
Precision	0.77	0.72	0.72
Recall	0.75	0.72	0.71
F1- Score	0.74	0.71	0.72
Train Accuracy	91.65%	91%	90.12%
Test Accuracy	75.81%	72%	71%
Validation Accuracy	71.92%	70%	70.43%

Figure 6. Comparision between performance metrics of each model on Grocery Store Dataset

Freiburg Dataset Observations	Resnet from scratch	Googlenet from scratch	AlexNet from scratch	Resnet Transfer Learning	Googlenet Transfer Learning
Accuracy	0.42	0.31	0.46	0.42	0.39
Precision	0.44	0.31	0.48	0.44	0.41
Recall	0.42	0.31	0.46	0.42	0.39
F1- Score	0.41	0.30	0.46	0.41	0.38
Train Accuracy	95.93%	88.05%	94.12%	87.62%	83%
Test Accuracy	42.82%	31.74%	45.39%	45.56%	33.28%
Validation Accuracy(approx)	58.00%	31.86%	47.94%	44.61%	38.10%

Figure 7. Comparision between performance metrics of each model on Freiburg groceries Dataset

Now let us analyse the performance metrics individually for each architecture on each dataset and understand what might be the causes of it:

Alexnet: Validation on the test set shows that pretrained Alexnet with transfer learning performed best in a short time compared to other proposed models. The proposed method achieved accuracy values of around 48percent, 73percent and 94percent for the Freiburg, Grocery Store and Vegetable Images datasets respectively. The Alexnet model is more primitive than the other two models, but gives satisfactory results based on the datasets other freiburg dataset. As we mentioned earlier a bit of imbalance in data is the reason why the combination couldn.t work as expexcted. However, we observed that as the images and datasets become more complex, the differences tend to become more pronounced, revealing the superior performance of the ResNet model.

Resnet: An advantage of using ResNet's deep CNN architecture for the grocery dataset is that the layers can be better stacked with far fewer kernels than the Alexnet model. ResNet models are less complex than other networks and can be easily optimised. The model also converges faster and gives better results than other networks. The Resnet model yielded the highest evaluation metric scores other than two models [Table1,2,3]. The accuracy results are 91.65percent, 95.93percent and 99.13percent.

Googlenet: From the tabular figures, we could infer that the Googlenet model performs notably well in comparison with the Alexnet and Resnet models. The training accuracies based on the three datasets are 91percent, 88.05percent, 98.13percent respectively. It gave better results compared to the traditional models and achieved significant improvement by tuning the learning rate and the batch size accordingly.

Transfer Learning: Using transfer learning, we were able to retrain the above results using the Freiburg dataset for the Resnet and Googlenet models. The final layer of the pretrained model was retrained to provide this classification, retaining significant weight-related knowledge from the initial training of the model and transferring it to the given dataset. The transfer learning model is being able to provide reasonable accuracy and classification similar to the model from scratch. Ideally, building a model from scratch would allow the customization of layers/weights giving better efficiency and a more accurate functionality. Using the Freiburg dataset would not have resulted in better results, we were limited in that regard. There are numerous possibilities leading to more complex models that can achieve more accurate results. [**?**, 13]

Concluding, comparing all models with its own Test and Validation accuracy results, the accuracy doesn't saturate and leads to overfitting especially in the Freiburg dataset. This is due to the imbalance in validation data. We have tried to optimize by using dropouts in the fully connected layers and it improved the accuracy values far better. Also larger augmentation of data has been performed to solve this issue but didn't get the satisfying result. Fine-tuning the learning rates or the hyper-parameters and/or adding or removing layers could also be done. The choice of activation functions such as ReLU, Sigmoid, and Softmax functions could also be more efficiently used to optimize the overfit issue in the network. Overall, the vegetable dataset has given the best results and the Freiburg dataset obtained inferior results and the source for overfitting. Now let us view the accuracy and loss representation for our best performing vegetable dataset models in order to understand how the model fitted on each Architecture. The orange line graph indicates the validation metrics whereas Blue line plot indicates train metrics in the following graphs.

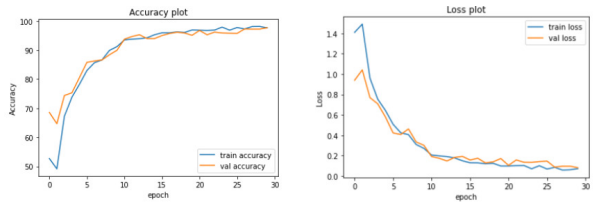

Figure 8. Accuracy and Loss plots on Vegetable Images Dataset for Googlenet

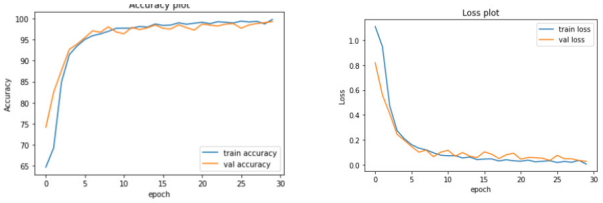

Figure 9. Accuracy and Loss plots on Vegetable Images Dataset for Resnet18

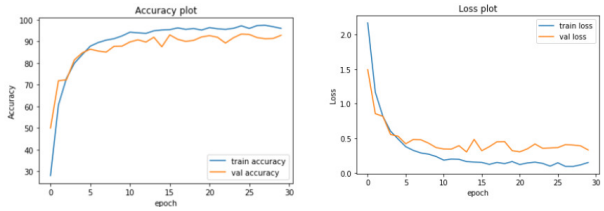

Figure 10. Accuracy and Loss plots on Vegetable Images Dataset for Alexnet

3.3. Ablative Study

We performed Hyperparameter tuning using grid search on learning rate for grocery store dataset with Resnet18 architecture because of its robust weights and bias.The reason why we choose this setup is because our 3rd dataset (i.e Vegetable Images dataset) is perfectly balanced and well trained, so we felt instead of applying it on the 3rd dataset we wanted to go with the grocery dataset as there is better scope to explore tuning since the model is underfitting and there was some generalization gap observed in the loss and accuracy plots of the trained model. Due to computational constraints, we could only perform grid search on Learning rates (0.01, 0.001, 0.0001) each trained for 50 epochs. Below are the plots of loss and accuracy over varying behaviour of learning rate. [14, 16]

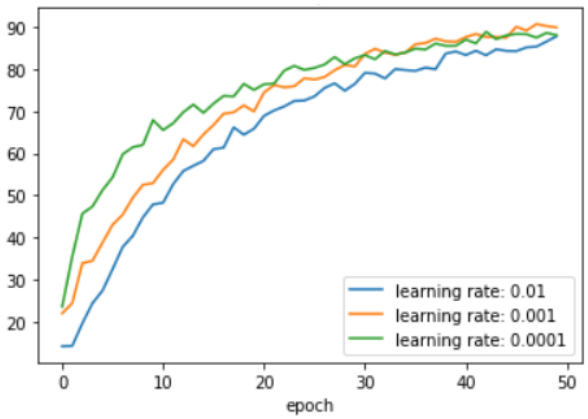

Figure 11. Comparision between different learning rates when trained using Resnet-18 model on Grocery store dataset The graph represents accuracy vs epochs for each learning rate

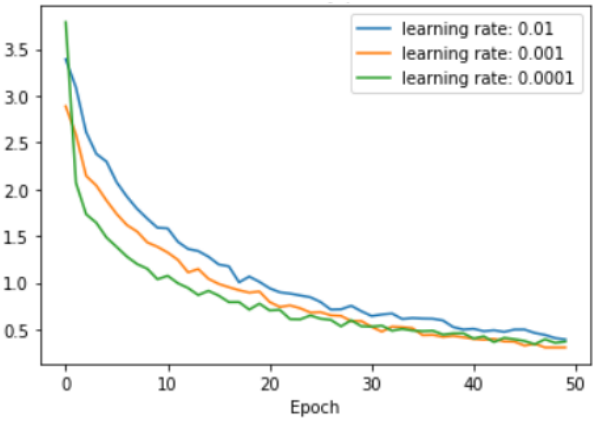

Figure 12. Comparision between different learning rates when trained using Resnet-18 model on Grocery store dataset The graph represents loss vs epochs for each learning rate

So from the above figures 11 and 12 we can infer that there is slight difference when learning rate is tuned. However, we can observe a difference when learning rate is set to 0.01,the model tends to perform a bit lower than the other two models with learning rates 0.001 and 0.0001. Therefore, from this we can understand that we can either use 0.001 or 0.0001 as learning rates to train the model on grocery store dataset using resnet18 architecture.

References

[1] Alexnet architecture and its current use. *https://iq.opengenus.org/architecture-and-use-of-alexnet/*. 4

[2] Alexnet documentation. *https://pytorch.org/hub/pytorch$_v$ision$_a$lexnet/*.4

[3] Freiburg groceries dataset link. *https://drive.google.com/drive/folders/1myukmv8kCX-CaxTNRf2aCpekUVs$_2$om2?usp = share$_l$ink*.2

[4] Googlenet documentation that we used as a template. *https://pytorch.org/hub/pytorch$_v$ision$_g$ooglenet/*.4

[5] Grocery store dataset link. *https://github.com/marcusklasson/GroceryStoreDataset*. 3

[6] Histogram of directed gradients: An overview. *https://towardsdatascience.com/hog-histogram-of-oriented-gradients-67ecd887675f*. 2

[7] Implementation of googlenet architecure on real world problems, an article we followed from towards data science in implementation. *https://towardsdatascience.com/deep-learning-googlenet-explained-de8861c82765*. 4

[8] Pytorch documentation refered while implementing tsne data visualization. *https://pypi.org/project/tsne-torch/*. 2

[9] Quick view into amazon go article that inspired us to explore this problem. *https://www.cnn.com/2018/10/03/tech/amazon-go/index.html*. 1

[10] Resnet documentation. *https://pytorch.org/hub/pytorch$_v ision_r esnet/$*.3

[11] Resnet18 implementation. *https://www.kaggle.com/datasets/pytorch/resnet18*. 3

[12] Rfid tags application in identifying products on shelves. *https://accuz.com/2021/03/rfid-tags-on-shelves-increasing-visibility-and-sales/*. 2

[13] Transfer learning documentation from pytorch that we reffered while implementing on our datasets and models. *https://pytorch.org/tutorials/beginner/transfer$_l earning_t utorial.html$*.5

[14] Understanding hyperparameter tuning. *https://www.kaggle.com/code/hatomugi/pytorch-cnn-with-hyper-parameter-tuning*. 6

[15] Understanding resnet architecture article. *https://medium.com/analytics-vidhya/understanding-resnet-architecture-869915cc2a98*. 3

[16] Understanding the dynamics of learning rate and effect on nueral network architectures. *https://machinelearningmastery.com/understand-the-dynamics-of-learning-rate-on-deep-learning-neural-networks/*. 6

[17] Understanding the literature review in context of image recognition and classification. *https://www.scribd.com/document/200341890/Literature-Review*. 2

[18] Vegetable images dataset link. *https://www.kaggle.com/datasets/misrakahmed/vegetable-image-dataset*. 3

[19] A view into adam optimizer. *https://machinelearningmastery.com/adam-optimization-algorithm-for-deep-learning/*. 4

[20] Hedvig Kjellstrom Marcus Klasson, Cheng Zhang. A hierarchical grocery store image dataset with visual and semantic labels. *arXiv preprint arXiv:1901.00711*, 2019. 3