CS 554 - Introduction to Machine Learning and Artificial Neural Networks

Project Report

Adaptive Cost-Sensitive Trade-off Analysis for Deep Neural Networks

Said Bolluk

E. A.

E. T.

7 June 2022

ABSTRACT

Imbalanced datasets complicate the accuracy of classification in machine learning. Image classification models, which include many crucial tasks such as cancer detection, and object detection in the securities industry, are prone to suffer from such datasets. To that end, this study aims to discover different Artificial Neural Network (ANN) models applied to three image datasets and obtain the following results: (1) By applying hyperparameter tuning to the ANN models and pre-processing the datasets, competitive baseline results will be obtained for each model. (2) Using the available literature to comprehend the existing-cost sensitive lost functions, this study will introduce a new loss function to deal with the misclassification of the imbalanced dataset and test its feasibility over the basic and advanced functions that exist in the literature. After completing the analysis, the results suggested that the introduced cost functions did not improve classification accuracy. However, the authors plan to advance the analysis with different cost-sensitive loss functions in future work, given the minor improvements.

1. INTRODUCTION

Computer vision is a trendy working area for providing digital solutions to real-world problems with its many subfields, such as text extraction, object/event detection, and spatial analysis. Among these, image recognition is an exciting application of Artificial Intelligence (AI). Image recognition, a classification task in machine learning, helps us understand the visual world and represent it in numbers that can distinguish the characteristic features of visuals. Following the patterns in visuals, image recognition can manage to classify images and automate/elaborate many tasks in the social and business life, including face recognition to encode systems [1], decision making for autopilots [2], and object detection in the defense industry [3].

Classification models in machine learning suffer from imbalanced datasets. While training such models, the imbalanced distribution of the class labels might induce misclassification. This is due to the update of the model parameters inevitably according to specific classes with numerical superiority [4]. As classification models, image recognition models also have difficulty functioning with such datasets. Despite the rapid advancements in the deep learning domain, the class imbalance problem, which is frequent in real-world data collection scenarios, remains challenging for image detection problems. In the literature, the research on the class imbalance problem is investigated from two central angles: Data Level Approach and Algorithm Level Approach [5]. In data level approaches, balanced datasets are established via either oversampling of the minority classes or undersampling of the majority classes. On the other hand, algorithm-level approaches operate in the training phase and try to amplify the effect of minority class representation in the overall optimization step.

Loss functions are critical components of every optimization problem, as they define an objective on which the model's performance is evaluated. Loss functions measure the overall distance, or error, between the estimated and actual value of a specific input set. Based on this evaluation, internal updates of the model weights are affected when performing backpropagation. A mathematical optimization approach aims to minimize this loss function for stable convergence. Depending on the scenario, different loss functions are used to develop a well-defined objective. Therefore, there is no such silver bullet to map loss functions to algorithms in the machine learning domain. The most important factors to be considered when deriving a loss function are the machine learning algorithm selected, the optimization approach and characteristics of the derivatives, and the nature of the dataset.

One of the most popular loss functions used in machine learning studies is the cross-entropy loss (logistic loss), a metric used to evaluate classifier performance in a given task. The error is bounded between zero and one as a probabilistic metric. However, as said above, the cross-entropy loss might be paralyzed when handling imbalanced datasets. As an algorithm-level approach to dealing with an imbalance dataset, cost-sensitive loss functions might be an option. The cost-sensitive approach minds the distribution of the class labels in a dataset and calculates the error of the model's prediction regarding those distributions [6]. With this approach, cost-sensitive models can

classify the datasets through a balanced training process. For example, the Focal Loss, a modified version of cross-entropy loss, down-weights the examples that are easy to train (majority classes) and focuses training on minority classes with tunable focusing parameters [7].

With these in mind, we will perform several image classifications over the balanced and imbalanced versions of three datasets using ANN models. The models include the Multilayer Perceptron (MLP) algorithms and the Convolutional Neural Network (CNN) models, specifically designed for an image processing task. The datasets are CIFAR-10, Fashion MNIST, and Intel Image Classification. We will first analyze the models with simple hyperparameters settings to reach the achievements in the literature and save the outputs as the baseline results. By working with three different ANN algorithms, this study aims to decide on the best model for training efficiency and classification accuracy. Then, we will introduce some cost-sensitive functions to improve the baseline results and continue our analysis with the best desirable ANN model to interpret the contribution of the cost-sensitive approach. This part will include the loss functions that already existed in the literature and those tested with minor changes. In the next section, the methodology of the process will be given. In the results section, we will compare the accuracy of the baseline and cost-sensitive models. In the discussion section, we will analyze the effect of the cost-sensitive approach and question its contribution. Finally, the study's limitations and future works will be discussed.

2. METHODOLOGY

Image recognition tasks are usually performed over Artificial Neural Network (ANN) models. However, some traditional machine learning models, such as Support Vector Machines (SVM) [8], or Decision Tree [9], can provide good but not competitive results. ANN models are the algorithms that mimic the human brain in data extraction and processing [10]. ANN models utilize the input parameters separately, which refers to online learning, taking a linear combination of inputs and transforming them into non-linear forms with activation functions (Figure 1). Then, the model parameters are updated according to the error calculated by comparing the model's prediction and the input set's actual label. ANN models can be designed with random hidden layers with random hidden units. This enables the building of complex discriminant functions and effectively classifies datasets. Images are a perfect example of such data on which the ANN models yield extraordinary results. This study performed image classification over three different datasets matched with three different ANN models. The datasets were examined in their balanced and imbalanced forms to observe the model's accuracy. For the first step, we created baseline models with regular loss functions. For the next step, we reviewed the literature to find cost-sensitive loss functions to compensate for the accuracy loss induced by the imbalanced data. Here, we also suggested a new cost-sensitive loss function featuring the occurrence and non-occurrence probabilities of the classes. Such updates will be tested over a single desirable ANN model a dataset, among the tested three, considering the computational capacity (Figure 2).

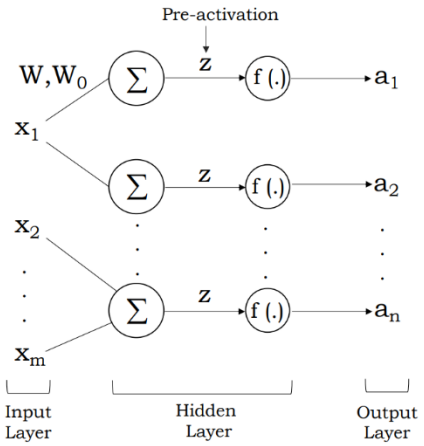

Figure 1. Neural Network Diagram

2.1 Baseline Models

We first implemented an image classification on the CIFAR-10 dataset via the MLP algorithm to establish a base model. MLP is a fully connected neural network model conducting forward-pass calculation. The CIFAR-10 dataset consists of 60,000 32x32 color images in ten classes belonging to several animals and vehicles, with 6,000 images per class. There are 50,000 training images and 10,000 test images. Since one of the project's goals is to enhance the accuracy of image classification using advanced classifiers and architectures, we wanted to show that the MLP could

achieve only a limited amount of accuracy. This is because MLP treats images as tabular data and examines more than three hundred features, where a notable amount of those features might be redundant, to classify the images. Accordingly, state of the art suggests that around 60-70% accuracy is possible with MLP using hundreds of hidden units [10].

Table 1. Hyperparameter setting for the baseline MLP on CIFAR-10

Hyperparameter	**Selection**
Hidden Layer Size	(4000, 1000, 4000)
Alpha	0.001
Batch Size	4
Activation	ReLU
Optimizer	Adam
Initialization	Kaiming Initialization

After reporting the results of the MLP model, which is a conventional ANN model, we passed to more complex models: Deep Learning Models. The process of finding the best deep learning architecture is often a challenging task, which involves the following tasks: creating a subset of architecture candidates based on data complexity, selecting and tuning hyperparameters, model training, evaluating candidate performance, and re-iterating the process.

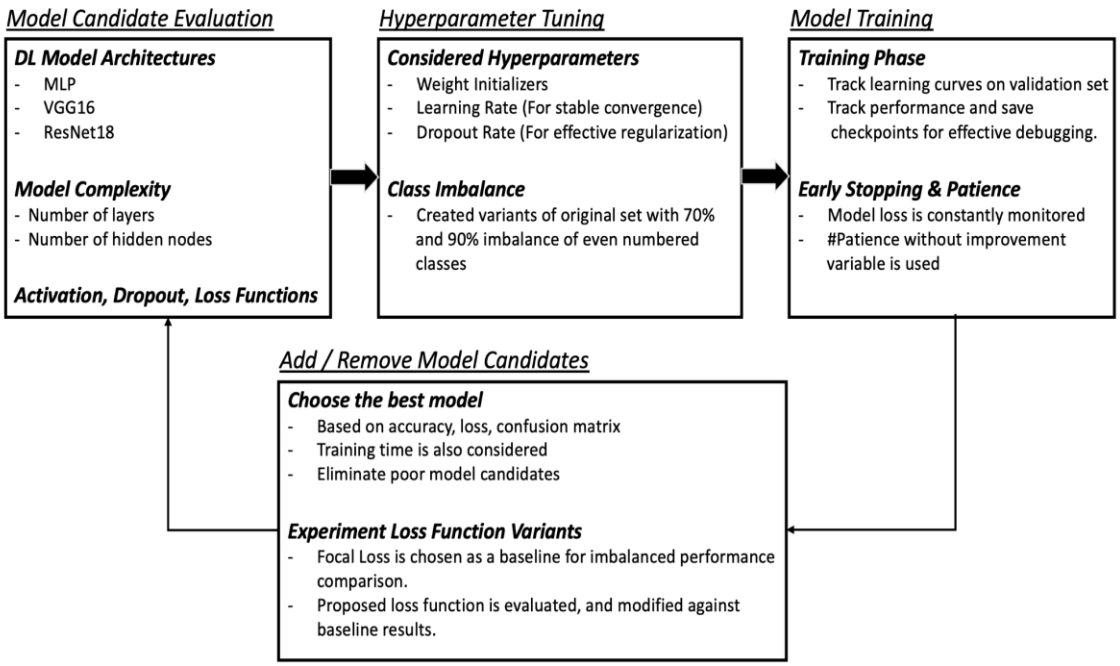

Figure 2. Framework of the methodology

The most famous architecture used in the literature is adapting CNN architectures for image classification tasks. The input of CNN models can be a 2D matrix (e.g., grayscale image) or a 3D tensor (e.g., color image). CNN has three main types of layers: convolutional layers where most calculations occur to extract high-level features, pooling layers for dimensionality reduction, and fully connected layers that perform classification based on the weights and features propagated through previous layers. In a CNN architecture, the ability to learn complex patterns increases with the layer size of the network. However, deep learning architectures may suffer from the Vanishing Gradients Problem after a certain amount of depth. As more layers are added to the network, the gradients of the loss function approach zero in the optimization step, making the network nearly impossible to be trained. This behavior can lead to overall inaccuracy of the whole network [11,12]. Following the literature review on deep learning algorithms, we moved with two different CNN models: ResNet and VGG16.

Sometimes, simple CNN models may not be enough to solve complex state-of-the-art problems. To solve this issue, deep CNN models can be preferred. However, using more complex models can cause saturation problems, which may negatively affect the prediction performance. To solve this issue, residual networks came to the stage. ResNet is one of the deep learning architectures used by combining CNN blocks multiple times.

The basic idea of the ResNet architecture [13] is that there is a direct linkage by skipping some layers of the model. This connection is called skip connection. During the skip connection procedure, the dimension of the inputs may change. To handle this problem and protect the overall

structure, dimensions can be increased with the help of a zero-padding operation. Besides, 1x1 convolutional layers may be added to match the dimensions with the inputs. The vanishing gradient problem in deep CNN models can be solved thanks to this smart skip connection technique. Besides, with the help of the skip connection technique, the model gets a chance to protect itself against some layers, which may hurt the performance of the architecture.

The Fashion-MNIST dataset [14] is used in the model exploration phase because the original MNIST dataset is relatively straightforward and overused in the community. Fashion-MNIST is a dataset of Zalando's article images—consisting of a training set of 60,000 examples and a test set of 10,000 examples. Each example is a 28x28 grayscale image associated with a label from 10 classes. Each training and test example is assigned to one of the following labels: T-shirt/Top, Trouser, Pullover, Dress, Coat, Sandal, Shirt, Sneaker, Bag, and Ankle Boot.

As displayed in Table 2, a Residual Network with 18 hidden layers is used with the ReLU activation function. When building the model, no transfer learning procedure is applied to examine the interior workings of the model. Instead, the He initialization with normal distribution is applied. Stochastic Gradient Descent with Nesterov Momentum is used as an optimizer with a learning rate of 0.001. Nesterov momentum is an additional feature to the momentum concept related to calculating the moving average of the gradients of projected positions in the search space. The state of the results for the Fashion-MNIST dataset with Residual Networks is around 90%, with excessive effort spent on data preprocessing.

To achieve state-of-the-art results, besides normalization of pixels, random cropping, random erasing, and converting to RGB are applied for better generalization of the model.

Table 2. Hyperparameter setting for the baseline ResNet on Fashion MNIST

Hyperparameter	**Selection**
Hidden Layer Size	18 Hidden Layers
Alpha	0.001
Batch Size	256
Activation	ReLU
Optimizer	StochasticGradient Descent
Initialization	He initialization

Intel Image Classification Data is a multiclass dataset used in image scene classification. The context of the dataset is natural scenes around the world. The data format is 150x150 resolution

RGB images. The data contains 14,000 training images and 3,000 test images under six class labels: Buildings, Forest, Glacier, Mountain, Sea, and Street (Figure 3).

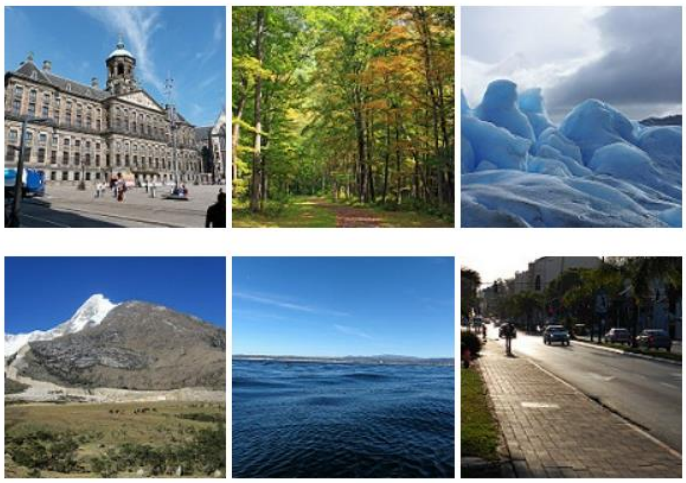

Figure 3. Intel Image Classification Dataset

We used a pre-trained VGG16 model and applied transfer learning to classify images in Intel Image Classification dataset. VGG16 architecture is shown in Figure 4. The model takes a 224x224 pixel RGB image as input. To match 150x150 pixel data with the model, resizing is applied on the data in data pre-processing. The model has five convolutional layers with ReLU activation and max pooling, and three fully connected layers with ReLU activation and SoftMax at the end. The number of outputs at the last layer depends on the number of classes in the classification tasks. To analyze the model on Intel Image Classification dataset, we applied transfer learning by removing the last layer and adding a new layer with six outputs. After adding the new layer, we trained the model on Intel Image Classification dataset by only updating the last layer of VGG16.

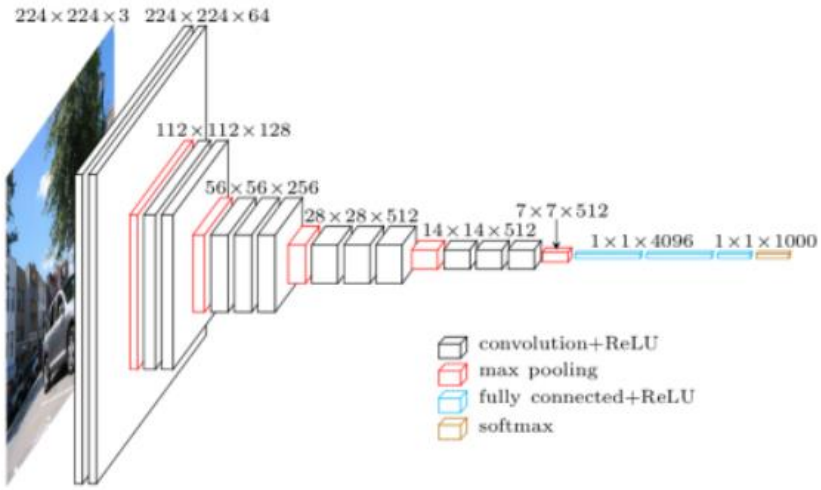

Figure 4. VGG16 Architecture

Before training and testing the model with Intel Image Classification dataset, we applied pre-processing to training and test sets. Here, different pre-processing steps are applied. The steps are listed below:

Training data:

i. Random Resized Crop
ii. Random Horizontal Flip
iii. Normalize

Test data:

i. Resize
ii. Center Crop
iii. Normalize.

The state of the results for the Intel Image Classification dataset is around 90%. To achieve state-of-the-art results, selected hyper-parameters are shown in Table 3.

Table 3. Hyperparameter setting for the baseline VGG16 on Intel Image Classification

Hyperparameter	Selection
Batch Size	4
Learning Rate	0.001
Learning Rate Decay	0.1 for every 7 epochs
Optimizer	Stochastic Gradient Descent
Momentum	0.9
Alpha	0.001

2.2) Cost-Sensitive Models

The last part of the ANN models is where the losses are calculated to update the model's parameters. Since we were dealing with a classification task, the calculated loss should be the difference between the predicted and actual labels. Therefore, we employed the Cross-Entropy Loss, which is a very popular loss function in classification, in our models. Cross-entropy loss compares the occurrence probability of multiple classes and their actual labels on the same feature space. For example, in a classification with n numbers of possible outcomes (labels), the ANN model calculates the occurrence probability of each outcome. Then loss function calculates n different errors for each class and update the weight of them accordingly. For multiclass cases, let us say, r and y are the actual and predicted labels of a given input set where y is a probability of specific label (Equation 1):

$$CE = -1 \times \sum_{i=1}^{n} r_i \times log(y_i) \quad (1)$$

However, in cases where the class distribution is not balanced, the estimate of the loss can be misleading with the cross-entropy loss. This is because, number of samples in minority classes are low and parameters are not often updated according to minority classes. Therefore, a cost-sensitive approach should be introduced that prioritizes the minority class in loss calculations. One of the methods is to give some pre-calculated weights to the classes and make weights according to the percentage of samples in each class. Hence, a loss calculated using a minority class will be multiplied by a higher number and learning will yield toward the minority class and equalize minority and majority classes' importance. Also, searching for different loss functions in the literature, we decided that focal loss can be a good example of cost-sensitive loss functions. The modified version of cross-entropy loss, named focal loss, which considers class imbalance is used in experimentations. Focal loss is an extension of the cross-entropy loss function that down-weight

easy examples and focus training on hard negatives with a tunable focusing parameter γ with being bigger than zero [15]. In the focal loss, minority class losses are increased using a factor γ, which ranges between a random number (Equation 2). This slows down updating parameters of the majority classes, and thus increases the learning capability of the model on the side of minority ones [15].

$$FL = -1 \times \sum_{i=1}^{n} r_i \times (1 - y_i)^{\gamma} \times log(y_i) \quad (2)$$

Focal loss function can also be rewritten in terms of CE (Equation 3). This makes cross-entropy function as an input for focal loss. Gaining inspiration from this, we thought that we can use a weighted cross entropy function as an input for focal loss and named that function as Weighted Focal Loss (WFL). WFL initiates the loss calculation with several parameters derived from the class distribution of the dataset. It puts more attention on the minority classes in loss calculations. Therefore, the convergence of those parameters will be faster, and the overall parameter update will be balanced. For the weight terms in the cross-entropy loss, we determined a weight matrix and assigned 1.0 and 0.1 for the minority and majority classes.

$$FL = (1 - e^{CE})^{\gamma} \times CE \quad (3)$$

To assess the contribution of the cost-sensitive methods mentioned above, we transformed the datasets into an imbalanced form with half of the classes reduced by 90%. In the next section, the results of the baseline and cost-sensitive models will be provided.

3. RESULTS

ANN models in training image data requires high computational capacity. Unfortunately, we did not have enough testing capacity in this study. Therefore, we created only an imbalanced dataset for each set with 90% of certain classes eliminated. However, to see the accuracy reduction gradually as we go forward to imbalanced forms, we examined three versions of Fashion MNIST: Original, 70% reduced, and 90% reduced [Table 4].

Table 4. Baseline Results for ResNet on Fashion MNIST

	Top-1 Accuracy	Top-5 Accuracy	CE Loss
Original Set	88.710	99.840	0.308
70 % Imbalance	85.810	99.790	0.384
90% Imbalance	79.200	99.570	0.543

3.1 Baseline Results

After conducting several scenarios with different hyperparameter settings (Table 1), we observed that MLP only achieved 40.64% accuracy in classifying the CIFAR-10 dataset. We also observed that increasing the number of epochs did not help significantly in increasing the model accuracy. Moreover, when the data was imbalanced, the accuracy was even lower with 30.75 % (Table 5). The cross-entropy loss fell short in reinforcing the model in covering minority classes (Figure 4).

Table 5. Baseline Results for MLP on CIFAR-10

	Top-1 Accuracy	Top-5 Accuracy	CE Loss
Original Set	40.640	5.527	5.527
90% Imbalance	30.750	68.620	6.632

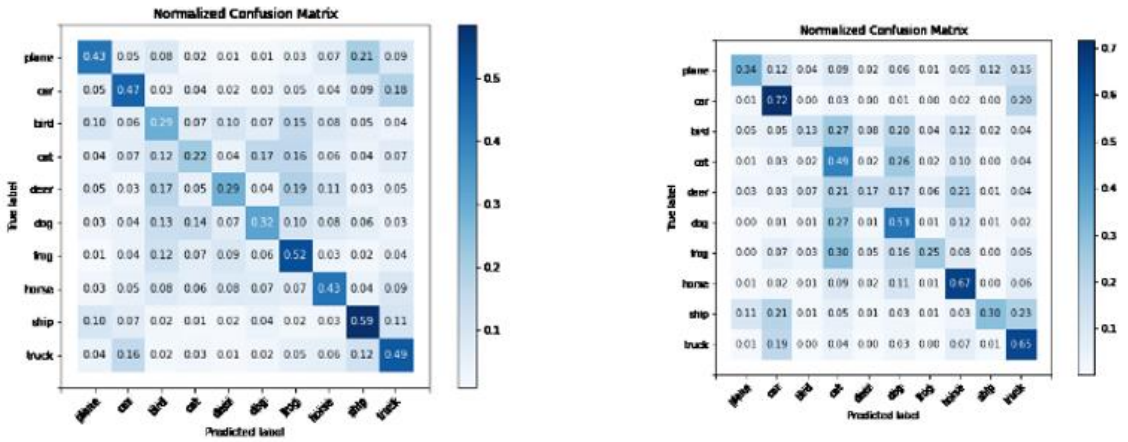

Figure 4. Confusion Matrices for MLP on CIFAR-10: Original (left) and 90% Imbalanced (right)

Classification results for the ResNet on Fashion MNIST are represented in Table 4. Even though the accuracy on the original set was satisfactory with around 89%, the accuracy was 79.2% for the 90% imbalanced set. The classes that were reduced from the original set exhibited misclassification of the model as they weighted relatively less than the majority ones (Figure 5).

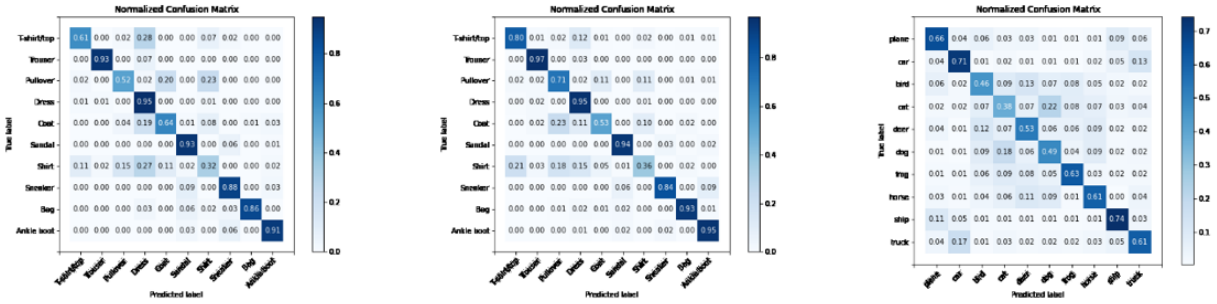

Figure 5. Confusion Matrices for ResNet on Fashion MNIST: Original (left), 70% Imbalanced (middle), and 90% Imbalanced (right)

VGG16 model was 92.53% accurate in classifying the original dataset, whereas its accuracy was reduced to 85% when predicting the classes of 90% imbalanced set (Table 6). Figure 6 represents the confusion matrices for the Intel Image Classification dataset in different forms.

Table 6. Baseline Results for VGG16 on Intel Image Classification

	Top-1 Accuracy	Top-5 Accuracy	CE Loss
Original Set	92.530	100.000	0.227
90% Imbalance	85.070	99.830	0.455

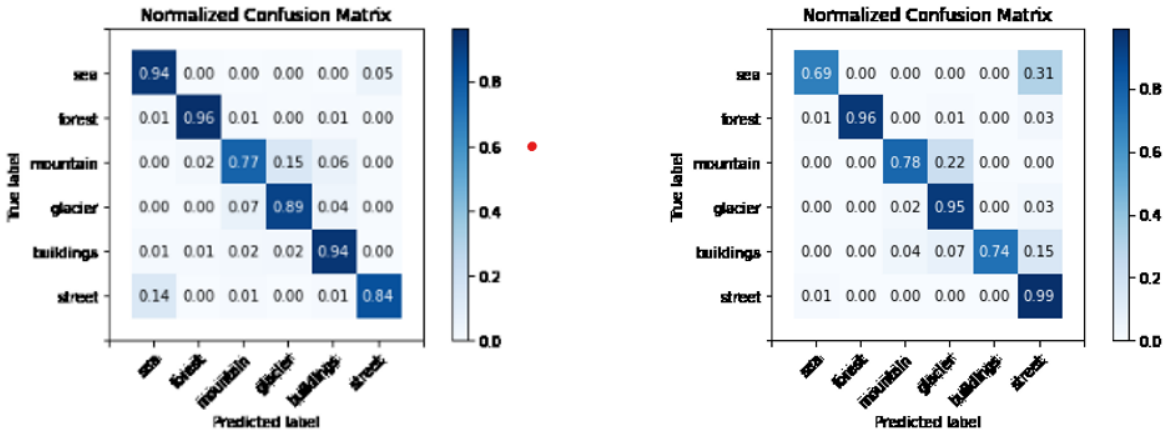

Figure 6. Confusion Matrices for VGG16 on Intel Image Classification: Original (left) and 90% Imbalanced (right)

3.2 Cost-Sensitive Results

After reporting each model with its several runs over different datasets, we decided to continue with the cost-sensitive loss functions to further the analysis in the imbalanced case. This section tested the VGG16 model on the imbalanced version of the Intel Image Classification dataset with the following cost functions: Cross-entropy loss, focal loss, and weighted loss. This is due to the technical complexities since the CNN algorithms require advanced Graphics processing units (GPU) and extensive computational time.

Introducing cost-sensitive lost functions appeared to be valuable when working with VGG16 on the Intel Image Classification dataset (Table 7). There are notable improvements both in the accuracy and the loss calculated for the imbalanced and original versions of the dataset. Plus, the confusion matrices in Figure 6 depict the improvement in the accurate classification of the minority classes.

Table 9. Cost-Sensitive Results for VGG16 on Intel Image Classification

Loss Function	Top-1 Accuracy	Top-5 Accuracy	Loss
Cross-Entropy	85.07	99.93	0.4546
Focal Loss	86.77	99.83	0.3516
Focal Loss + Weighted Loss	86.83	99.87	0.3567

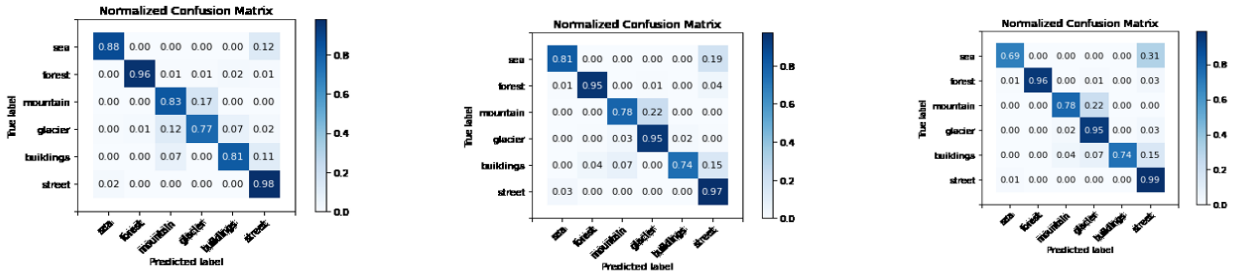

Figure 6. Cost-Sensitive Confusion Matrices for VGG16 on Intel Image Classification: Focal + Weighted Loss(left), Focal Loss (middle), Cross-Entropy Loss (right)

4. DISCUSSION

In the first part, the baseline results indicate that MLP models fail to satisfy accurate classification for image data where the features are the pixels with hundreds or even thousands. Such results indicate the necessity of complex models in image classification, for example, CNN models. Accordingly, the tested two CNN models, ResNet and VGG16, performed well in classifying images.

The next part is more of an experimental study where we aimed to evaluate the efficiency of cost functions in imbalanced datasets. Imbalanced datasets complicate the accurate classification of images in the field. This is a prevalent problem, and many practitioners are working to enhance the efficiency of the techniques aiming to propose solutions to deal with imbalanced data. Starting from this point, we aimed to analyze the dynamics of the ANN models in working with different cost-sensitive loss functions and see whether we could achieve improvement for the classification of minority classes. In this sense, the VGG16 model with elaborated cost functions achieved

improved the classification accuracy of the minority classes. As a future work, same procedure can be implemented via the other ANN models over different image datasets to ensure the accuracy achieved for the Intel Image Classification dataset with the help of const-sensitive approach in loss calculations.

Since ANN models require a significant amount of computational power, we suffered from the lack of adequate hardware during the analysis. A complete hyperparameter tuning could not be achieved due to high computational time. Similarly, the proposed loss functions could not be tested over multiple settings. This led to creating and testing simple architectures and thus moderate or even minor improvements in our results. Considering the dynamics of the dataset and understanding the ANN models' characteristics might yield advanced loss functions. Based on this fact, to obtain or even pass the baseline results in the literature, a future work of this study could be performing analysis with advanced ANN architectures explicitly developed for the desired dataset. More importantly, another future work of this study could be developing cost-sensitive loss functions particular to a specific dataset or ANN model. This could only be possible via advanced computational capacity.

5. CONCLUSION

This study analyzed different image datasets and proposed three ANN architectures to obtain the state-of-art results in the literature, followed by different cost-function that were tested for the imbalanced form of the Intel Image dataset using VGG16 model. Understanding the image data and the classification process of images through ANN and CNN models with hyperparameter tuning, this study provided a perspective on the current approaches in the literature. Moreover, the development process enables us to discover the fundamentals and the current trends in the field that might be helpful for our future studies. Even though there is no significant improvement achieved with the cost-sensitive loss functions, the minor improvements reveal the potential of promising results over the advanced loss functions regarding the imbalanced datasets and the cost for the misclassification of minority classes, which might be crucial in real life.

REFERENCES

1. Dong, S., Wang, P., & Abbas, K. (2021). A survey on deep learning and its applications. Computer Science Review, 40, 100379. https://doi.org/10.1016/j.cosrev.2021.100379

2. Zhang, J., & Li, J. (2020). Testing and verification of neural-network-based safety-critical control software: A systematic literature review. Information and Software Technology, 123, 106296. https://doi.org/10.1016/j.infsof.2020.106296

3. Yu, C., & Pei, H. (2021). Face recognition framework based on effective computing and adversarial neural network and its implementation in machine vision for social robots. Computers & Electrical Engineering, 92, 107128. https://doi.org/10.1016/j.compeleceng.2021.107128

4. V. Garcia, J. Sanchez, J. Mollineda, R. Alejo, and J. Sotoca, “The class imbalance problem in pattern classification and learning,” in Congreso Espanol de Informatica, 2007, pp. 1939– 1946.

5. Hou, Y., et al.: Adaptive learning cost-sensitive convolutional neural network. IET Comput. Vis. 15(5), 346–355 (2021)

6. Takase, T., Oyama, S., & Kurihara, M. (2018). Effective neural network training with adaptive learning rate based on training loss. Neural Networks, 101, 68–78. https://doi.org/10.1016/j.neunet.2018.01.016

7. T. -Y. Lin, P. Goyal, R. Girshick, K. He and P. Dollár, "Focal Loss for Dense Object Detection," 2017 IEEE International Conference on Computer Vision (ICCV), 2017, pp. 2999-3007, doi: 10.1109/ICCV.2017.324

8. Kaur, P., Singh, G., & Kaur, P. (2019). Intellectual detection and validation of automated mammogram breast cancer images by multi-class SVM using deep learning classification. Informatics in Medicine Unlocked, 16, 100151. https://doi.org/10.1016/j.imu.2019.01.001

9. Bostik, O., & Klecka, J. (2018). Recognition of CAPTCHA Characters by Supervised Machine Learning Algorithms. IFAC-PapersOnLine, 51(6), 208–213. https://doi.org/10.1016/j.ifacol.2018.07.155

10. Lin, Z., Memisevic, R., & Konda, K.. (2015). How far can we go without convolution: Improving fully-connected networks.

11. Khan, S.H., et al.: Cost-sensitive learning of deep feature representations from imbalanced data. IEEE Trans. Neur. Net. Lear. Syst. 29(8), 3573–3587 (2018)

12. Raj, V., et al.: Towards effective classification of imbalanced data with convolutional neural networks. In: IAPR Workshop on Artificial Neural Networks in Pattern Recognition, pp. 150–162. Springer, Magdeburg (2016)

13. He, Kaiming, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. "Deep residual learning for image recognition." In Proceedings of the IEEE conference on computer vision and pattern recognition, pp. 770-778. 2016.

14. Xiao, H., Rasul, K. & Vollgraf, R. (2017). Fashion-MNIST: a Novel Image Dataset for Benchmarking Machine Learning Algorithms (cite arxiv:1708.07747Comment: Dataset is freely available at https://github.com/zalandoresearch/fashion-mnist Benchmark is available at http://fashion-mnist.s3-website.eu-central-1.amazonaws.com/)

15. T. -Y. Lin, P. Goyal, R. Girshick, K. He and P. Dollár, "Focal Loss for Dense Object Detection," 2017 IEEE International Conference on Computer Vision (ICCV), 2017, pp. 2999-3007, doi: 10.1109/ICCV.2017.324