

Transcript:

Greetings to everybody.

Slide 1 (Introductory)

Team 1 consists of three members: Samuel, Rachel and Konstantinos, who seek to implement convolutional neural networks using the CIFAR 10 dataset and to subsequently evaluate the performance of the model.

Slide 2 (Info for Convolutional Neural Network + Info for the dataset)

The Convolutional Neural Network is a modelling technique with a wide range of applications (Vaidya, N.D.; Flatworld Solutions, N.D.). However, in most cases the technique is applied in image classification or object detection (Hijazi et al, 2015).

The overall aim of this project is to build and evaluate the performance of a convolutional neural network that utilizes the CIFAR 10 dataset (Canadian Institute for Advanced Research, 2009). The dataset consists of 60 000 images, each belonging to one of the following ten categories: 'airplane', 'automobile', 'bird', 'cat', 'deer', 'dog', 'frog', 'horse', 'ship' and 'truck'. There are 6 000 images per category.

The aim of the project is to build a model with sufficiently high performance to enable each image to be classified correctly, according to their respective category.

Slide 3 (Validation set)

In order to train and test the model, the dataset was first divided into two parts: namely, a training set and a testing set:

The training set is used, as the name suggests, to train the model in order to obtain the best fit and thus improve the prediction rate. The testing set is the portion of the data that is used in the final evaluation of the model, to discern whether it is accurate enough to be applied in similar cases.

However, there is an intermediate stage, which is referred to as 'validation'. To undertake this stage, it is necessary to further divide the training set into a training sample and a validation sample in order to carry out the validation procedure. Accordingly, the initial 10000 records of the training set were selected to be utilized for our validation step as indicated.

(Include a screenshot of the relevant commands in the slide).

Slide 4 (Give details of the metadata after validation)

Two separate line charts were constructed in order to compare the training loss along with the validating loss and the training accuracy along with the validating accuracy, as a function of the number of epochs.

The training loss demonstrates how well the model fits the training data and the validation loss demonstrates how well the model fits new data (Yathish, 2022). Similarly, the training accuracy demonstrates the measure of accuracy of the model in respect of the training data and the validation accuracy demonstrates the measure of accuracy of fitting new data. In this manner, it is possible to identify underfitting or overfitting issues and to resolve these issues.

In this case it was observed that, until the second iteration, both training accuracy and validation accuracy remained low, implying underfitting of the model. However, following the subsequent

iteration, the training accuracy significantly exceeded the validation accuracy and consequently the model did not fit new data, and was therefore overfitting.

Slide 5 (Discuss the rationale of having a validation set)

The main objective of splitting the training dataset to obtain a validation set is to prevent the model from overfitting, as previously mentioned. In this case the model performed well in classifying the training data but, on the other hand, was not able to make accurate predictions with data that was as yet unseen by the model.

Therefore, steps were taken in order to eliminate this issue. However, a detailed explanation of these steps is beyond the scope of this presentation and thus will not be discussed further. Validation is a procedure that is used to review the performance of the model based on different combinations of hyperparameters, to allow the best selection for modelling the test data.

Slide 6 (Discuss the structure of the ANN)

Proceeding to the structure of the ANN, the flatten layer converts the final three-dimensional layer into a one-dimensional vector and therefore the structure is transformed to fit the input of an artificial neural network later on. After the value of each pixel gets flattened, two sequential dense layers consisting of 256 and 10 neurons finalize the process of CNN. Despite this, the number of neurons within the dense layer is variable, depending on the complexity of the classification, however it is very important to highlight that since we wish to classify the images across 10 categories, the last dense layer should consist of 10 neurons. To further explain this, imagine that the dense layer is a simple layer of neurons in which each unit receives input from every neuron of the previous layer, sums the product of each input with the value of the neuron within the dense layer and adds a unique bias. Thus, they generate an equal quantity of values to the quantity of neurons contained in the dense layer. For this reason, it is necessary that the last dense layer contains 10 neurons.

This structure of ANN gives the ability of the Artificial Neural Network to extract the feature of the input more efficiently and thus significantly improves the performance of the model. In addition to this, parallel processing allows for more efficient computation: It is adjustable and has great learning potential. However, it requires excessive computation and thus the potential for overfitting is greater in comparison to other modelling structures.

Slide 7 (Discuss the activation function used: ReLU and SoftMax)

A topic which has not yet been discussed relates to the activation functions utilized in between the layers, which are the Rectified Linear Unit (ReLU) function and SoftMax.

To begin with, ReLU is the most popular activation function (DanB, 2018). Despite its name, the output of the function is non-linear, which increases the accuracy of the Convolutional Neural Network model. Otherwise, if a non-linear function was not utilized, multilayers of the Convolutional Neural Network models could be compressed into a single layer with equivalent performance. In addition, Convolutional Neural Network models are observed to train faster in comparison with other non-linear activation functions.

It has been verified that ReLU is less computationally expensive than other activation functions. On the other hand, for activations in the region where the input is negative, the gradient will be 0 and thus the relevant weights will no longer adapt from this point onwards. Consequently, the corresponding neurons will stop responding to variations in error or input during the modelling

process. According to the formula of the function for negative inputs the outcome of the function is zero and for positive inputs the outcome is equal to the input.

In respect of the SoftMax activation function (Wood, N.D.), the outputs of the Convolutional Neural network model are scaled so that they sum to 1. Each of these outputs represent the probability that the input image represents the corresponding class category. Thus, it is a useful function that contributes to determining the final decision of the model in terms of classification. The same transformation can occur utilizing the process of standard normalization. However, with the SoftMax activation function the max logit is weighted significantly greater than the rest logits.

Slide 8 (Discuss the loss function used)

We now present the reasons for selecting the categorical cross-entropy loss function. In general, the loss function represents the value which should be minimized when training the model. To further expand on this topic, the cross-entropy loss function measures the difference between the predicted probabilities and the true labels of a classification model (Koech, 2020; Kumar, 2020). More specifically, the categorical cross entropy loss function is utilized for multi class classification model where there are two or more output labels which are converted into categories after utilizing the one-hot category encoding procedure.

In general, the cross-entropy loss function is easy to implement and optimize, since appropriate built-in functions are provided in most of neural network environments. In addition to that, the graphical representation of the function consists of a smooth and convex shape: a fact which facilitates the global minimum identification for gradient-based optimization methods. Lastly, cross-entropy is not affected by scaling or shifting of the predicted probabilities if the values range between 0 and 1. This can be useful when regularizing or calibrating the output of the model (LinkedIn, 2023).

However, the cross-entropy loss function can be sensitive to outliers or imbalanced data and thus can result in overfitting. In this case, despite the fact that the probability classification is too high for the true category, the cross-entropy loss function outcome is located at a high value level. In addition to that, the accuracy of the model performance is not indicated directly and thus cross-entropy should always be used in combination with more meaningful metrics as such precision, accuracy, F1-score or recall (Brownlee, 2019).

Slide 9 (Number of epochs utilized)

A total of 25 epochs were selected to train our model and thus improve its performance with every subsequent iteration. This value consists of a high quantity which significantly increases the computational cost. However, we have set the constraint that when the model performance reaches a point where the outcome of the cross-entropy loss function for the validation data remains the same for the two subsequent cycles then we conclude that our model has reached a satisfactory level of performance and thus the training should stop at that point. Consequently, the model training process performs only the initial 7 iterations until it reaches that point. The remaining cycles are omitted.

Slide 10 (Neural Network design)

In regards to the design of the CNN, during the initial phase of modelling double sets of layers are constructed, each one consisting of a convolutional layer and a pooling layer: The convolutional layer is utilized to extract several features from the input image. Following that, the output becomes the

input of the following pooling layer whose primary aim is to reduce the size of the convolved feature map and thus reduce the computation costs (Kumar, 2021).

To begin with, the first convolutional layer receives inputs of dimensions equal to the length and width of the input image, both of which are equal to 32. However, each of these images contain three color channels: the red channel, the green channel and the blue channel. Therefore, the input shape consists of three dimensions which are equal to 32 multiplied by 32 multiplied by 3.

The kernel size refers to the dimensions of the sliding window all over the input. The selection of these dimensions contributes a significant role as regards the overall performance of the model: small kernel sizes are able to extract more details of the characteristics of the input image. Additionally, selecting a small kernel size leads to a smaller reduction in layer dimensions, thus allowing for a deeper architecture. However, the overall computational cost of the model increases largely.

As regards the model constructed by our team, the length and width of the kernel size were both chosen to be equal to 3, since the input images contain small details which should be extracted during the process of modelling in order to improve the accuracy of the model. Subsequently, the stride hyperparameter was selected to be 1, indicating that 1 pixel should be shifted over at a time, with every subsequent operation. The impact of stride size on the performance of the model is similar to the kernel size: the smaller the value, the better the performance and thus the accuracy of the model. Lastly, we have chosen 64 filters which is a high value for the filtering argument to enhance the performance of the model. Essentially, the greater number of filters chosen, the better the performance (Pradeep, 2017).

Next, a pooling layer was constructed. This process involved selecting a kernel with a user determined size which produces only one value for each kernel slice from the input (Brownlee, 2019). In the model produced by our team the max pooling layer is utilized which, as the name suggests, extracts only the maximum value for each slice, thus only the most prominent features proceed to the modelling phase. This procedure is repeated twice, before reaching a point at which the final classification part of the CNN can occur.

Slide 11 (Concluding section)

To conclude, the project has given us the opportunity to develop and extend our knowledge in respect of the implementation of Convolutional Neural Networks for image classification purposes. However, this modelling technique has a wider range of applications, for example, in facial recognition, analyzing documents or modelling climate change (Kim et al, 2022). There is a lot more to learn about this modelling technique and hence this project represents just the beginning.

Slide 12 (Thank the audience)

We would like to thank you for your attentiveness in listening to our presentation and to express our gratitude in being able to convey our knowledge of Convolutional Neural Network modelling.

References:

Vaidya, D. (publication date unknown). What Is Artificial Neural Network? Available from: <https://www.wallstreetmojo.com/artificial-neural-network/#Advantages-And-Disadvantages> [Accessed 16 July 2023].

Unknown (publication date unknown). APPLICATIONS OF CONVOLUTIONAL NEURAL NETWORKS. Available from: <https://www.flatworldsolutions.com/data-science/articles/7-applications-of-convolutional-neural-networks.php> [Accessed 16 July 2023].

Hijazi, S. Hijazi, S., Kumar, R. and Rowen, C. (2015). Using convolutional neural networks for image recognition. Cadence Design Systems Inc.: San Jose, CA, USA, 9(1).

Canadian Institute for Advanced Research (2009) The CIFAR-10 dataset. Available from: <https://www.cs.toronto.edu/~kriz/cifar.html> [Accessed 16 July 2023].

Yathish, V. (2022). Available from: <https://towardsdatascience.com/loss-functions-and-their-use-in-neural-networks-a470e703f1e9> [Accessed 16 July 2023].

DanB (2018) Rectified Linear Units (ReLU) in Deep Learning. Available from: <https://www.kaggle.com/code/dansbecker/rectified-linear-units-relu-in-deep-learning/notebook> [Accessed 16 July 2023].

Wood, T. (publication date unknown). What is the Softmax Function? Available from: <https://deeptai.org/machine-learning-glossary-and-terms/softmax-layer> [Accessed 16 July 2023].

Koech, K. E. (2020) Cross-Entropy Loss Function. Available from: <https://towardsdatascience.com/cross-entropy-loss-function-f38c4ec8643e> [Accessed 16 July 2023].

Kumar, A. (2020) Keras – Categorical Cross Entropy Loss Function. Available from: https://vitalflux.com/keras-categorical-cross-entropy-loss-function/#:~:text=categorical_crossentropy%3A%20Used%20as%20a%20loss,into%20categorical%20encoding%20using%20keras [Accessed 16 July 2023].

LinkedIn (2023) What are the advantages and disadvantages of using cross-entropy loss for classification tasks? Available from: <https://www.linkedin.com/advice/0/what-advantages-disadvantages-using-cross-entropy> [Accessed 16 July 2023].

Brownlee, J. (2019) A Gentle Introduction to Cross-Entropy for Machine Learning. Available from: <https://machinelearningmastery.com/cross-entropy-for-machine-learning/> [Accessed 16 July 2023].

Kumar, P. (2021) Max Pooling, Why use it and its advantages. Available from: <https://medium.com/geekculture/max-pooling-why-use-it-and-its-advantages-5807a0190459> [Accessed 16 July 2023].

Pradeep, S. (2017) Why would one use larger strides in convolutional NNs, as opposed to smaller strides? Available from: <https://www.quora.com/Why-would-one-use-larger-strides-in-convolutional-NNs-as-opposed-to-smaller-strides> [Accessed 16 July 2023].

Brownlee, J. (2019) A Gentle Introduction to Pooling Layers for Convolutional Neural Networks. Available from: <https://machinelearningmastery.com/pooling-layers-for-convolutional-neural-networks/> [Accessed 16 July 2023].

Kim, J., Lee, M., Han, H., Kim, D., Bae, Y. and Kim, H.S. (2022). Case study: Development of the CNN model considering teleconnection for spatial downscaling of precipitation in a climate change scenario. *Sustainability*, 14(8), p.4719.