A.2 Second Research/Programming Assignment

Computer Vision

MSDS 458 – Artificial Intelligence and Deep Learning

Grayson Matera

October 23, 2022

Abstract

Utilizing the CIFAR 10 dataset, Deep Neural Network (DNN) and Convolutional Neural Network (CNN) architecture was explored in a series of 12 experiments. The purpose of this research is to examine the effectiveness of an assortment of neural network structures on computer vision tasks. Throughout the experimentation phase, small adjustments are made to the network structures and CIFAR 10 dataset. The results of the experiments are then compared to determine the impact of each structural change in the network. The methods of this research are non-comprehensive and open-ended. However, the conclusions drawn from these experiments provide a controlled look at the performance integrity of various neural network components on computer vision tasks. The experiments are broken up into three phases. Phase 1 of experimentation, Experiments 1-4, consists of base model designs for DNN and CNN structures. Phase 2 of experimentation, Experiments 5-8, build on the base models of Phase 1 by introducing regularization to the network designs. The results of Phase 1 are compared with Phase 2 to assess the impact of regularization on network performance outcomes. Phase 3 of experimentation, Experiments 9-12, consist of a series of incremental adjustments to the best performing model of Phase 2. These adjustments are designed to boost the networks performance towards an implementation ready design. The adjustments made in Phase 3 reveal the subtle influence of hyperparameter tweaking, batch normalization, layer stacking, and image augmentation on network performance outcomes.

Introduction

In many ways, the digital era has vastly changed the way scientists and researchers collect and utilize data. However, data does not always initially come in easily processable formats. Methods of analyzing visual mediums such as images and video remain on the cutting edge of artificial intelligence

(AI) research. Computer vision, a growing field in AI, is what experts use to tackle these types of visual problems. Computer vision is currently implemented in a wide array of industries, including manufacturing, energy, and automotive. The adoption of computer vision technology continues to grow and is expected to reach USD 48.6 billion by the end of 2022 (IBM, 2022). The CIFAR 10 dataset utilized in the following experiments is an experimental dataset often used to train and test network designs on image recognition. The dataset is composed of 60,000 images related to 10 different classes. Due to its prelabeled nature and accessibility, CIFAR 10 presents a simple entry into the nuances of AI based image recognition. The purpose of the following experiments is to explore these nuances in actual network designs and evaluate their impact on network performance. The goal is to identify several key design components that broadly contribute to improving network performance.

Literature Review

The history of computer vision is that of a long and complex timeline with the development of CNNs significantly aiding the advancement of computer vision technology. In 2012, a team of researchers from the University of Toronto designed a groundbreaking model called AlexNet which vastly improved upon previous rudimentary designs (Demush, 2019). Notably, AlexNet boasted a significant reduction I the error rate for image recognition tasks. Ultimately, the breakthrough design spurred a new era of computer vision advancements.

Methods

The following experiments were run using python via the Google CoLab environment with a standard GPU runtime setup. This was strategically chosen to help reduce runtimes and GPU strain.

Additionally, the environment comes preinstalled with all the packages necessary for the experiments including the CIFAR 10 dataset, numpy and pandas for data manipulation, matplotlib and seaborn for visualization, an array of module from sklearn and keras for modelling. Upon loading the CIFAR 10 dataset consisting of 60,000 images, it was split into 45,000 training images and 10,000 test images. 5,000 images were held back for validation. One-hot encoding was then applied to the dataset to numerically categorize the image labels.

The experimental portion of our research consists of 12 individual experiments separated into three (3) phases. In Phase 1, consisting of Experiments 1-4, we constructed, trained, and tested our baseline models on the preprocessed dataset. Experiments 1 & 2 included DNN architectures comprised of 2 layers and 3 layers respectively. Experiments 3 & 4 included CNN architectures comprised of 2 convolution/max pooling layers and 3 convolution/max pooling layers respectively. All for models in Phase 1 were specifically designed to not include a regularization method. This was purposely omitted from the baseline models to assess the utility of regularization in later phases. Additionally, the hyperparameters were standardized across Phase 1. Epochs were set to 50, ReLu activation, Adam optimizer, and standard batch size was implemented, and early stopping was disabled. The purpose of this was to assess the full training process and get accurate visualize understanding of if/when the models began to experience overfitting.

In Phase 2, consisting of Experiments 5-8, the previous experiments were reconstructed to include a method of regularization. The regularization method was standardized across all experiments in Phase 2 and included low penalty L2 for regularization. All other standards and designs from Phase 1 were carried over into Phase 2 experiments.

In Phase 3, consisting of Experiments 9-12, the best model from Phase 2 is further developed and analyzed for improvements. This phase of experimentation is less structured than the previous

phases. However, many standards from Phase 2 remain in effect. This includes batch size, L2 regularization, ReLu activation, Adam optimizer, and disable early stopping. In Experiment 9, our best Phase 2 model was adapted to include stacked convolution layers, batch normalization, and dropout methods. Experiment 10 further developed this model by adjusting the incremental dropout level in the network structure during training.

In Experiment 11, our dataset was further processed using image augmentation to increase the size of the train data. The new dataset was then tested on our previous model from Experiment 10 with the same standard methods. Experiment 12 is effectively an extension of Experiment 11 with only one adjustment made to the design. The standard 50 epoch design was increased to 200 epochs in an attempt to assess the full spectrum of descent during training. Throughout Phase 3, comparisons are made based on each iteration and their relative adjustments to the models.

Results

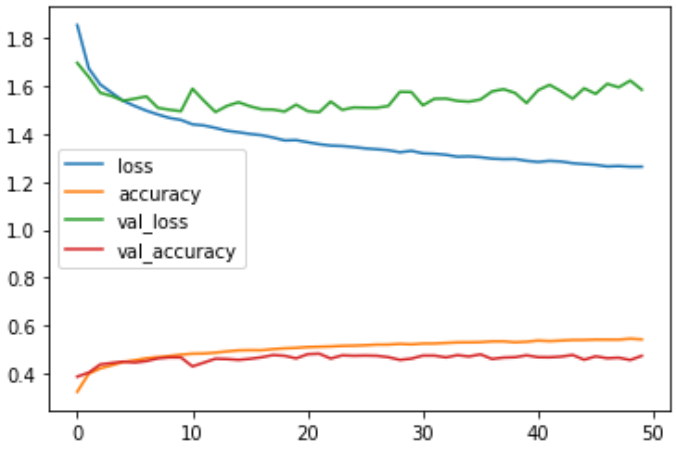
This section will cover the results of each experimental phase in order. Comparisons will then be made when relevant to our methodology.

Phase 1:

Phase 1 consisted of our baseline DNN and CNN model designs with no regularization method. The table below shows the performance metrics for each of the four Phase 1 experiments.

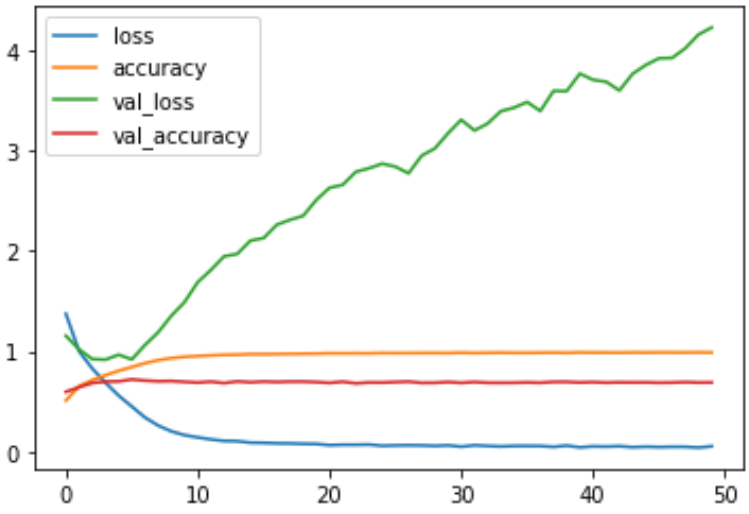
Model	Type	Training time	Training Loss	Training Acc	Val Loss	Val Acc	Test Loss	Test Acc
Model1	DNN	0:05:23	1.38	0.51	1.54	0.47	1.54	0.47
Model2	DNN	0:05:23	1.26	0.54	1.59	0.47	1.59	0.47
Model3	CNN	0:09:23	0.05	0.99	4.23	0.69	4.23	0.69
Model4	CNN	0:11:23	0.06	0.98	2.95	0.73	2.95	0.73

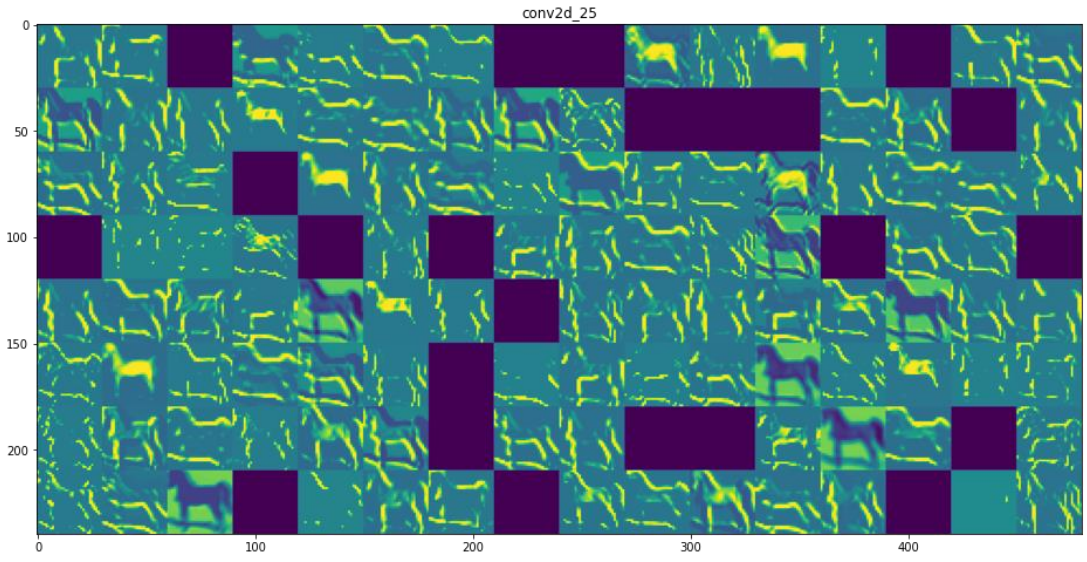
From the table, it is apparent that the DNN models performed poorly on this computer vision task. The increase in layers provided a negligible benefit in increasing its performance against the data. The 50 epoch standard allows us to see the overfitting that occurs rather quickly in these models. Below is a visualization of the models training data. We can see that the models begin to overfit around the seventh epoch where the accuracy stagnates and the loss increases drastically. This becomes even more apparent against the test data.

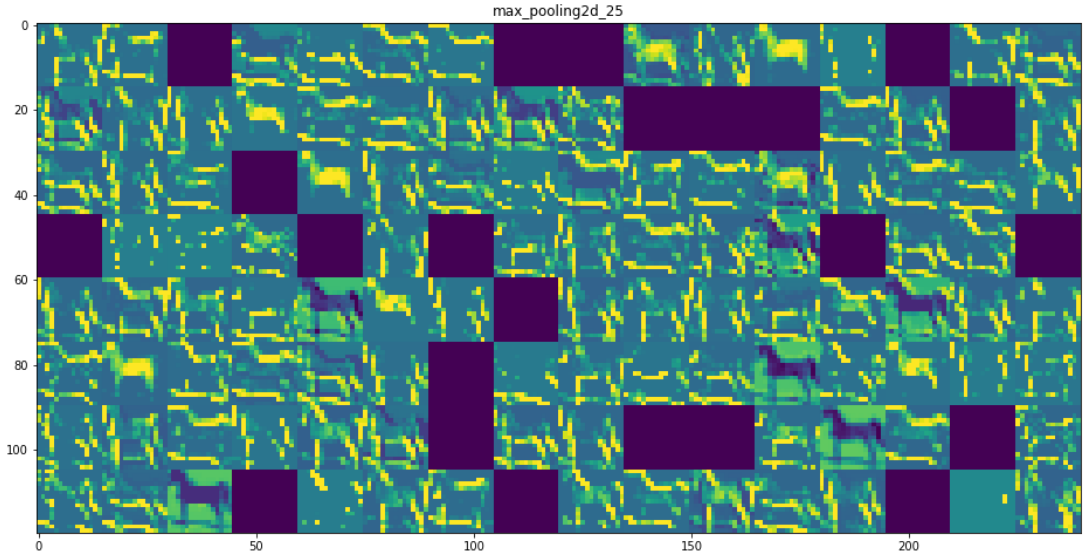


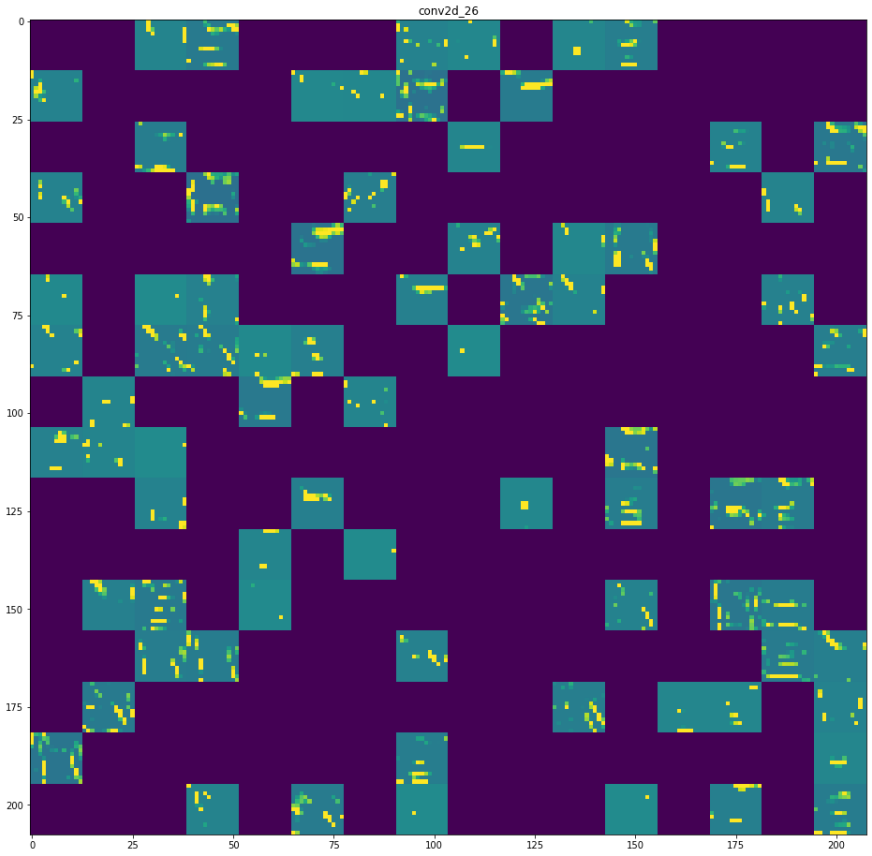
Both CNN models perform significantly better than their DNN counterparts. The 2 layers of convolution and max pooling in Model3 provide a 20% increase in accuracy. Additionally, the added

layers in Model4 increases the model's performance further. However, the 50 epoch standard still presents a problem of overfitting for the CNN models with their training peaking around 7 epochs as well. We can see from the extracted images of Model3 that the max pooling layers are corresponding to the features of the images. This likely means that these layers contributing to identifying the appropriate features of the images.






max_pooling2d_25
0
20
40
60
80
100
0
50
100
150
200

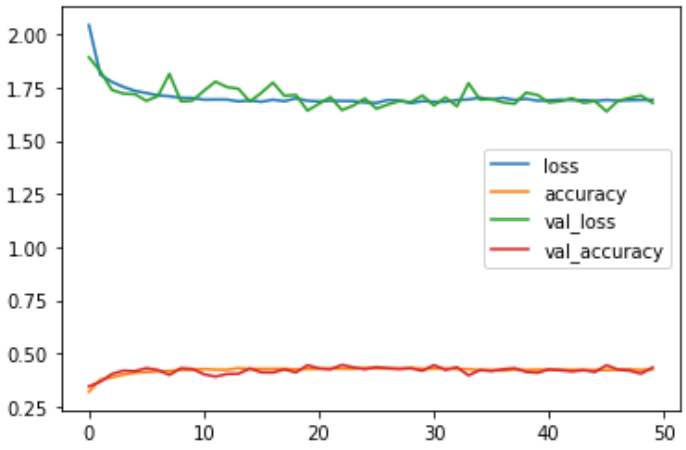

conv2d_26
0
25
50
75
100
125
150
175
200
0
25
50
75
100
125
150
175
200

Phase 2:

Phase 2 consists of our Phase 1 models with L2 regularization implementation. The table below shows the performance metrics for each of the four Phase 2 experiments.

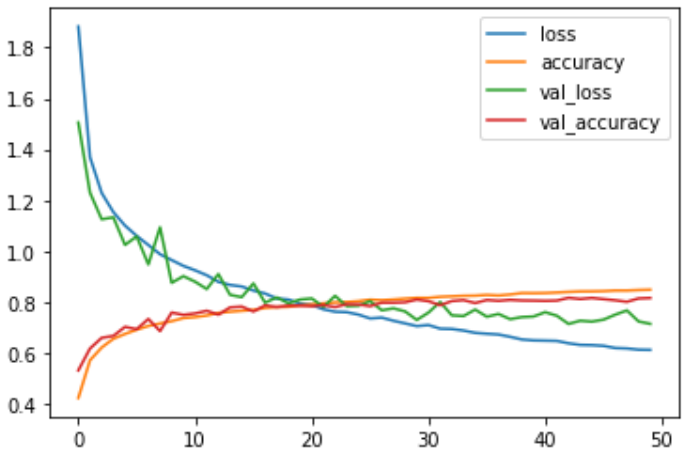
Model	Type	Training time	Training Loss	Training Acc	Val Loss	Val Acc	Test Loss	Test Acc
Model5	DNN	0:05:23	1.69	0.42	1.68	0.43	1.68	0.43
Model6	DNN	0:05:23	1.53	0.48	1.65	0.44	1.65	0.44
Model7	CNN	0:11:24	1.07	0.76	1.09	0.76	1.09	0.76
Model8	CNN	0:11:55	0.61	0.85	0.72	0.82	0.72	0.82

The performance metrics from Phase 2 are a bit perplexing. In the DNN models, we can see a reduction in overall performance with the inclusion of L2 regularization. Similar overfitting is present in the Phase 2 DNN models as in Phase 1. However, the peak training level at around 7 epochs is several percentages below the Phase 1 baseline metrics.



It seems the L2 regularization was beneficial to the CNN models performance. There is a notable increase to the accuracy scores of both CNN models. It would also seem that the models tendency

towards early overfitting has declined. Based on the metric visualizations below we can see that the descent of the CNNs training remains rather continuous across the 50 epoch standard. Model8 performs quite well and may even benefit from additional epochs.

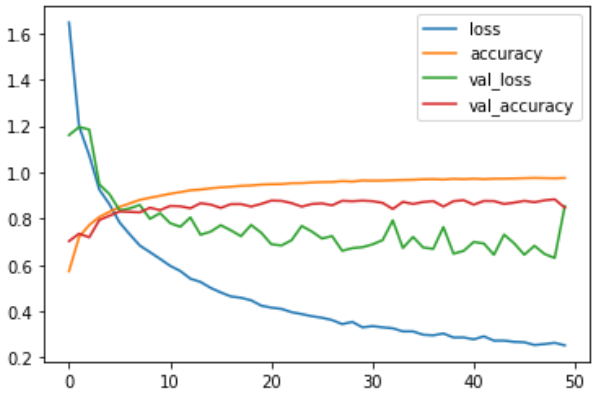


Phase 3:

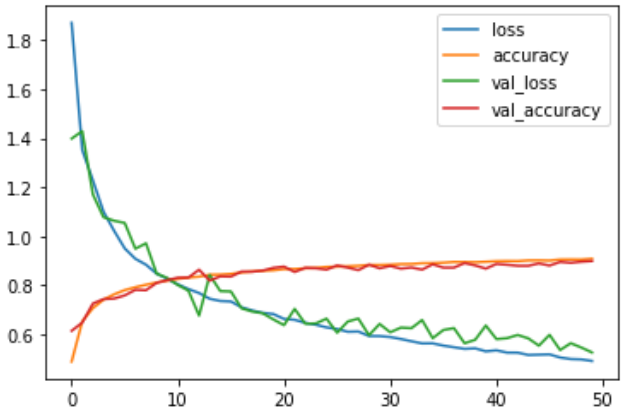
Phase 3 consists of incremental adjustments to our best model (Model8) from Phase 2 of experimentation. The table below shows the performance metrics for each of the four Phase 3 experiments.

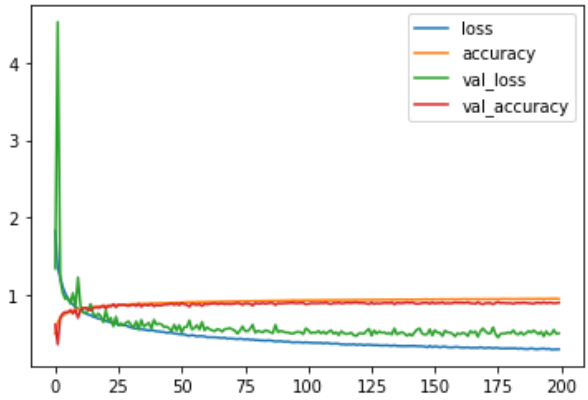
Model	Type	Training Time	Training Loss	Training Acc	Val Loss	Val Acc	Test Loss	Test Acc
Model9	CNN	0:20:24	0.17	0.96	0.71	0.82	0.71	0.82
Model10	CNN	0:27:23	0.25	0.97	0.85	0.85	0.85	0.85
Model11	CNN	0:33:23	0.49	0.91	0.53	0.90	0.53	0.90
Model12	CNN	2:09:09	0.30	0.95	0.51	0.90	0.51	0.90

The performance metrics from Phase 3 show an increase in performance from the best model in phase 2. Model9 consisted of the most adjustments to our previous best CNN model. With the addition of more convolutional and batch normalization layers, Model9 don't present much performance increase from Model8. However, the dropout rate adjustments in Model10 show a performance increase of 3%. The overall loss of Model10 also sees some improvement and it doesn't appear the model is in danger of overfitting within the 50 epoch standard.



Model11 and Model12 are virtually the same model run to different lengths. Both utilize the new dataset available through image augmentation. The new training data appears to have benefited the model's performance. Model11, running for 50 epochs, reaches 90% accuracy and does not appear to be overfitting. Model12, running for 200 epochs, also reaches 90% accuracy and does not show signs of overfitting despite its longer training time. The decrease in loss of Model12 is the only apparent difference between the two models performance.





Conclusions

Across all three experimental phases, we were able to incrementally increase the performance of our computer vision networks. However, after the completion of Phase 2, it was apparent that DNN architecture would likely not suffice for the task at hand. The CNN models in Phase 2 showed promising results from the implementation of L2 regularization with Model8 reaching an accuracy score of 82%. L2 implementation also appears to have benefited the model's descent and reduced the risk of overfitting during training. The increase in convolutional layers between Model7 and Model8 also shows promise in performance outcomes. This same relationship is apparent between Model8 and Model9 with the increase in layers. We see a reduction in the overall loss and a more stable model.

Stabilization of the model can also be attributed to the proper adjustments to dropout in the Model. This is apparent in the adjustments to the dropout rate made between Model9 and Model10. The loss in Model10 in notably decreased and the accuracy is improved. However, the most notable benefit of our experiments is the use of image augmentation in Model11 and Model12. Through image augmentation, the amount training data became much larger and allowed the model to train on more images. This benefited our model's performance significantly with a baseline increase of 5% accuracy. However, it appears that the increase in epochs between Model11 and Model12 yielded little benefit relative to the increase in training time. Overall, it is apparent that incremental adjustments to a CNN models structure and the relative dataset can greatly benefit performance metrics against a computer vision task.

References

Kumar M., & Chakrapani, A. (2022). Classification of ECG signal using FFT based improved Alexnet classifier. *PloS One*, *17*(9), e0274225–e0274225. https://doi.org/10.1371/journal.pone.0274225

IBM (2022). What is Computer Vision? IBM. https://www.ibm.com/topics/computer-vision#citation5

Rostyslav Demush (2019). A Brief History of Computer Vision and Convolutional Neural Networks. Hacker Noon. https://hackernoon.com/a-brief-history-of-computer-vision-and-convolutional neural-networks-8fe8aacc79f3