**Assignment 2: Convolution (Cats vs Dogs)**

To apply convnets to image classification problem and achieve high accuracy. Explain the relation between between sample sizes and whether training convnets from scratch or utilizing a pretrained network.

**Model Building:**

**Scratch Model**

For this project we have considered a total of 14 models and 9 – Scratch and 5 - Pretrained Models.

Scratch models are models that are trained from scratch, meaning that all of their parameters are initialized randomly. Pre-trained models are models that have been trained on a large dataset, such as the ImageNet dataset.

We have altered the hyper tuning parameters, nodes and layers of the models to identity a relation between input sample size with accuracy and model performance and also the usage of pretrained networks in evaluating the model performance

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Model** | **I/O Layers** | **Filter Size** | **Filters** | **Optimizer** | **Training/ Validation/Test** | **Dropout** | **Max pooling** | **Strides** | **Padding** | **Loss/ Accuracy** |
| M-1 | 5 | 3 | 32 - 256 | Adam | 1000/500/500 | 0 | Yes(Size -2) | - | - | (0.626, 0.644) |
| M-2 | 5 | 3 | 32 - 256 | Adam | 1000/500/500 | 0.5 | Yes(Size -2) | - | - | (0.552, 0.748) |
| M-3 | 6 | 3 | 32 - 512 | Adam | 1000/500/500 | 0.5 | Yes(Size -2) | - | - | (0.634,0.708) |
| M-4 | 5 | 3 | 64-1024 | Adam | 1000/500/500 | 0.5 | Yes(Size -2) | - | - | (0.670, 0.626) |
| M-5 | 5 | 3 | 32 - 256 | Adam | 2000/500/500 | 0.5 | Yes(Size -2) | - | - | (0.447,0.812) |
| M-6 | 5 | 3 | 32 - 256 | Adam | 3000/500/500 | 0.5 | - | 2 | - | (0.691,0.582) |
| **M-7** | **5** | **3** | **32 - 256** | **Adam** | **3000/500/500** | **0.5** | **Yes(Size -2)** | **-** | **-** | **(0.410,0.850)** |
| M-8 | 5 | 3 | 32 - 256 | Adam | 3000/500/500 | 0.5 | Yes(Size -2) | 2 | - | (0.488,0.814) |
| M-9 | 5 | 3 | 32 - 512 | Adam | 3000/500/500 | 0.5 | Yes(Size -2) | 2 | yes | (0.555,0.736) |

**Observations:**

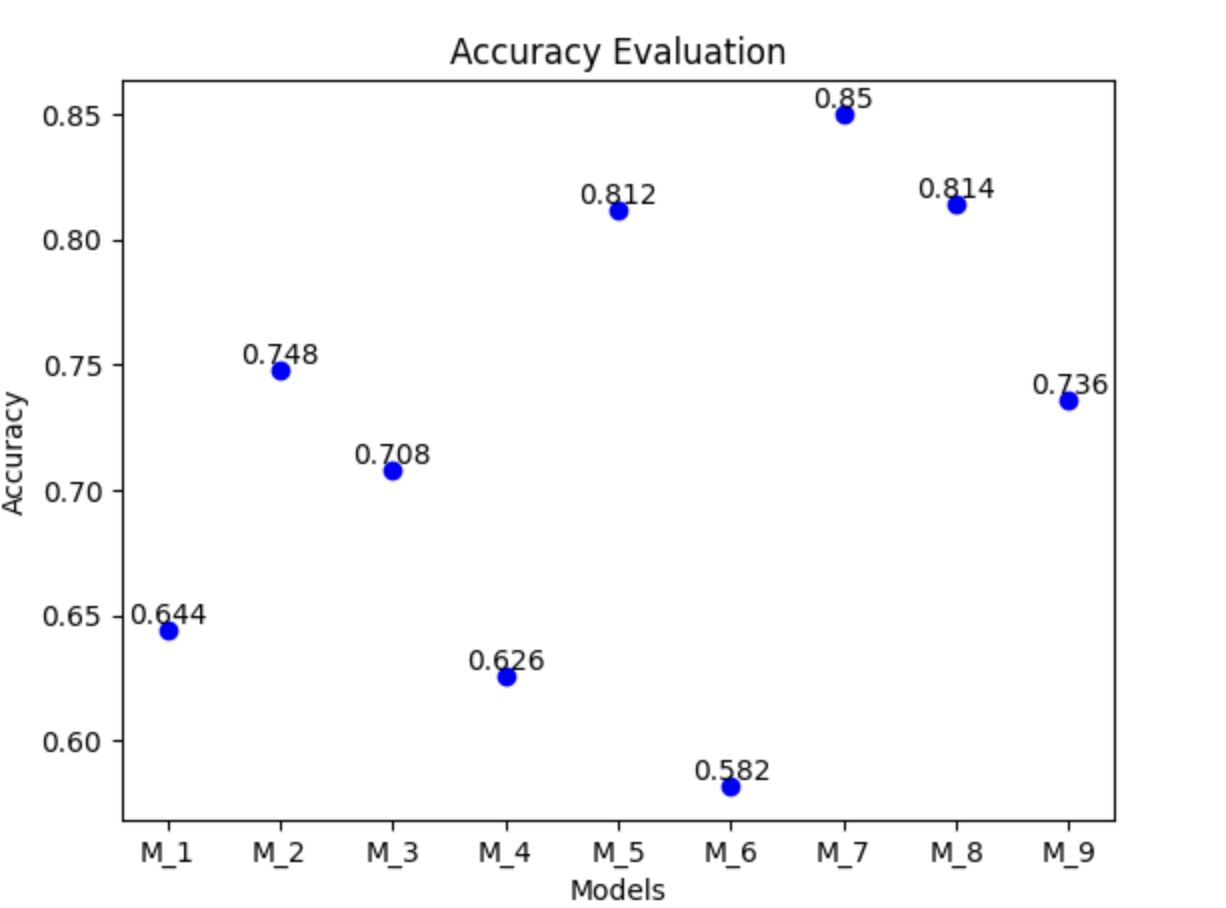
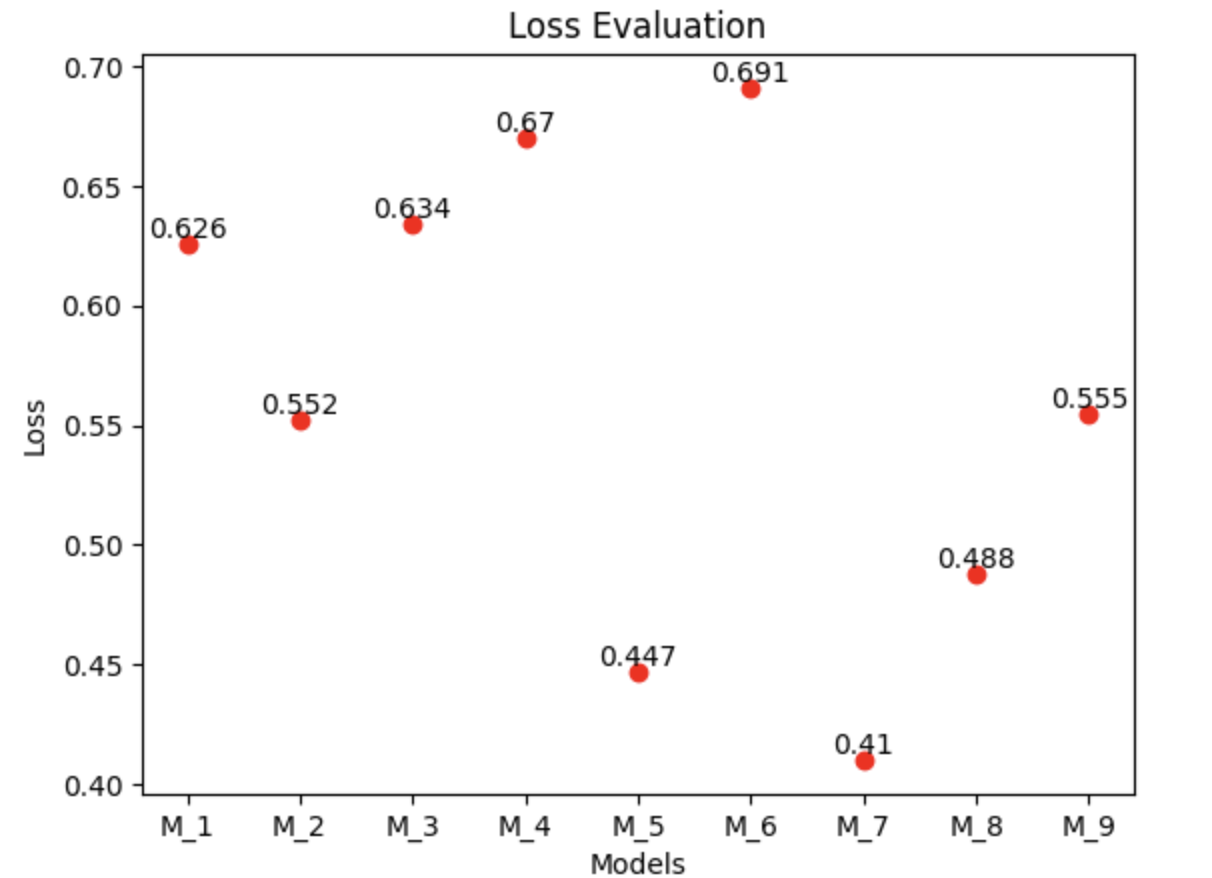
* The unregularized model trained on the Cats and Dogs dataset, utilizing a training sample of 1000, a validation sample of 500, and a test sample of 500, yielded a notably low accuracy of 64%. This outcome strongly indicates overfitting, a common problem in machine learning. Overfitting occurs when a model performs well on the training data but poorly on unseen data. In this case, the small training dataset size of 1000 instances is insufficient for the model to generalize effectively to the validation and test data.
* Model 2 has the highest accuracy among the first 4 models as it was trained with augmented image using the techniques Random Flip, rotate and Zoom as mentioned below

:

* Changing in the input filters does not have any significant differences in the model performance Model 3&4 had input filters from 32, 64, 128, 256, 512, 1024. Also, Hyperparameters selection is related to powers of 2 as many hardware accelerators, such as GPUs and TPUs, are optimized for operations involving powers of 2. Using a power of 2 for the number of filters can lead to more efficient memory access and computation.
* The training dataset was expanded to 2000 samples for Models 5 and 6, and the results showed that Model 5 achieved an accuracy of 81% and a loss of 44.7%. It's noteworthy that this was the same model that initially attained an accuracy of 74.8% when trained with only 1000 samples. However, when provided with a larger dataset, which included augmented versions of the images, Model 5 showed significant improvement in its ability to correctly recognize the images. This demonstrates how increasing the number of training samples, along with data augmentation, enhanced the model's learning and recognition capabilities.
* In Model 6, a pooling layer was omitted, and instead, strides were used to reduce the spatial dimensionality. However, this change in the mechanism from pooling to strides did not yield any significant improvement in the model's performance.
* Model 7 consists of 5 layers with convolutional filters of dimension 3. The number of filters varies across these layers, ranging from 32 to 256. This indicates a relatively deep convolutional neural network (CNN). The training sample was increased from 2000 to 3000.
* Model 8 is built with strides and max pooling, well this is desirable combination and the accuracy is around 81.4, In model 9 the padding was on but this does not have a significant effect in the models performance

**Conclusion:**

In conclusion, out of all the models 7th model has the highest accuracy at 85%, indication that there is increase in the model performance with the increase in the Input data. Model 7 represents a moderately deep CNN with specific hyperparameters and dataset settings. Its performance was good, but it's clear that adjustments like pooling and strides can significantly impact accuracy.



**Pre-Trained Network:**

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Model** | **Dense Layers** | **Optimizer** | **Training/ Validation/Test** | **Dropout** | **Pretrained weight update** | **Freeze Layers** | **Augmented images** | **Loss/ Accuracy** |
| M-1 | 1(256 Nodes) | Rmsprop | 1000/500/500 | 0.5 | No | No | No | (4.74,0.962) |
| M-2 | 1(256 Nodes) | Rmsprop | 1000/500/500 | 0.5 | Yes | No | Yes | (7.98, 0.966) |
| M-3 | 1(256 Nodes) | Rmsprop | 1000/500/500 | 0.5 | Yes | Yes | Yes | (5.7, 0.97) |
| M-4 | 1(256 Nodes) | Adam | 5000/500/500 | 0.5 | Yes | No | Yes | (0.151, 0.996) |
| M-5 | 1(256 Nodes) | Adam(1e-5) | 5000/500/500 | 0.5 | Yes | Yes | Yes | (0.008,0.998) |

VGG-16 served as the pretrained model for the image recognition task. VGG-16 is a robust and extensively trained neural network that has learned from a vast and diverse dataset comprising thousands to hundreds of thousands of images from various categories. It stands as a formidable and globally utilized pretrained network for a wide range of image recognition tasks.

* In the context of using pre-trained networks, the size of the training sample emerged as a significant factor influencing the model's learning characteristics and its performance on unseen data.
* While rmsprop is indeed an excellent optimizer for building convolutional neural networks, Adam maintains an advantage due to its unique combination of Momentum and rmsprop, which effectively optimizes the neural network.
* By preventing the pre-trained network from updating its weights, we preserve the valuable knowledge encoded in the pre-trained model. Instead, the model's focus shifts towards training the densely connected classifier layer at the end. This approach has proven to be a game-changer, as evidenced by the performance improvements in Models 2 and 3. Additionally, it serves as a valuable technique for preventing overfitting.
* Since pre-trained networks are typically trained on a broad range of image categories, freezing the initial layers of the model preserves the general categorization layers. This forces the model to concentrate specifically on the image recognition task at hand. This strategy has yielded positive results, particularly in the fine-tuned models, such as Model 3 and Model 5, which have achieved the highest accuracy within their respective sample sizes.

**Conclusion:**

In the case of pretrained networks the strategy of freezing the initial layers of the pretrained network, thus preventing it from updating its weights during training, has proven effective in mitigating overfitting. This approach ensures that the model doesn't memorize the training data but instead generalizes well when faced with unseen data.

When we increased the sample size from 1000 to 5000, a significant enhancement in the model's performance became evident. Moreover, the model exhibited minimal loss, signifying its proficiency. Offering the model a larger dataset, alongside augmented versions of the images, emerged as an efficient approach to boost accuracy.

