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

This project involves a selection of 14 models, comprising 9 trained from scratch and 5 pretrained models. Scratch models are initialized randomly, without leveraging any prior training, while pretrained models have been trained on extensive datasets like ImageNet. This diversification allows for a comprehensive exploration of both approaches, offering insights into the effectiveness of starting from scratch versus building upon pre-existing knowledge.

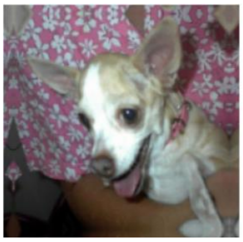
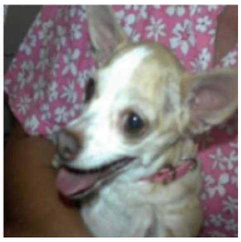
We've modified hyperparameters, nodes, and layers of the models to investigate the correlation between input sample size, accuracy, and model performance. Additionally, we've assessed the impact of using pretrained networks on evaluating model performance. This analysis aims to discern how these factors influence the effectiveness of the models.

Model	I/O Layers	Filter Size	Filters	Optimizer	Training/ Validation/Test	Drop out	Max pooling	Strides	Padding	Loss/ Accuracy
M-1	5	3	32 - 256	Adam	1000/500/500	0	Yes(Size -2)	-	-	(0.643, 0.650)
M-2	5	3	32 - 256	Adam	1000/500/500	0.5	Yes(Size -2)	-	-	(0.601, 0.694)
M-3	6	3	32 - 512	Adam	1000/500/500	0.5	Yes(Size -2)	-	-	(0.608,0.700)
M-4	5	3	64-1024	Adam	1000/500/500	0.5	Yes(Size -2)	-	-	(0.672, 0.646)
M-5	5	3	32 - 256	Adam	2000/500/500	0.5	Yes(Size -2)	-	-	(0.420,0.824)
M-6	5	3	32 - 256	Adam	3000/500/500	0.5	-	2	-	(0.583,0.716)
M-7	5	3	32 - 256	Adam	3000/500/500	0.5	Yes (Size -2)	-	-	(0.693,0.542)
M-8	5	3	32 - 256	Adam	3000/500/500	0.5	Yes(Size -2)	2	-	(0.435,0.806)
M-9	**5**	**3**	**32 - 512**	**Adam**	**3000/500/500**	**0.5**	**Yes (Size -2)**	**2**	**yes**	**(0.438,0.834)**

Observations:

- The unregularized model trained on the Cats and Dogs dataset, with a training set of 1000 samples, a validation set of 500, and a test set of 500, demonstrated a low accuracy of 64.6%. This suggests overfitting, a prevalent issue in machine learning where a model performs well on training data but poorly on unseen data. The limited size of the training dataset hinders the model's ability to generalize to validation and test data effectively.

- Model 3 achieves the highest accuracy among the initial four models due to its training involving augmented images, employing methods such as Random Flip, rotation, and zoom, as detailed below.

   :

- Altering the input filters does not notably impact the performance of the models. Model 3 and 4 both utilized input filters ranging from 32 to 1024. Furthermore, the selection of hyperparameters is influenced by powers of 2, aligning with hardware accelerators like GPUs and TPUs, which are optimized for such operations. Employing powers of 2 for filter numbers enhances memory access and computational efficiency.

- Expanding the training dataset to 2000 samples for Models 5 and 6 resulted in noteworthy findings. Model 5 achieved an accuracy of 82.4% and a loss of 42%, marking a substantial improvement compared to its initial accuracy of 70.0% when trained with only 1000 samples. The inclusion of augmented images in the larger dataset notably enhanced Model 5's proficiency in accurately identifying images, underscoring the benefits of increased training samples and data augmentation for improving learning and recognition within the model.

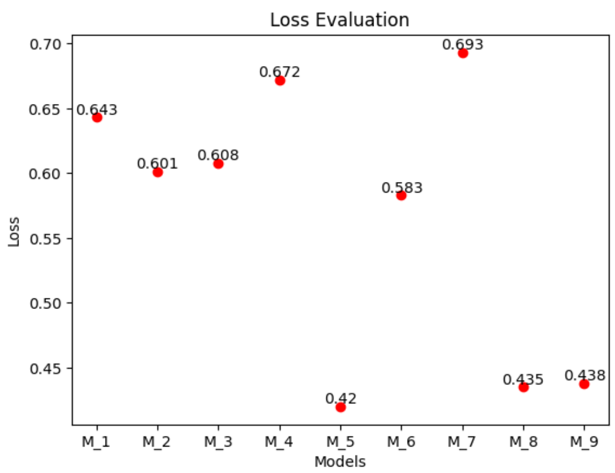
- In Model 6, the traditional pooling layer was replaced with strides to decrease spatial dimensionality. However, this alteration from pooling to strides did not result in any notable enhancement in the model's performance.

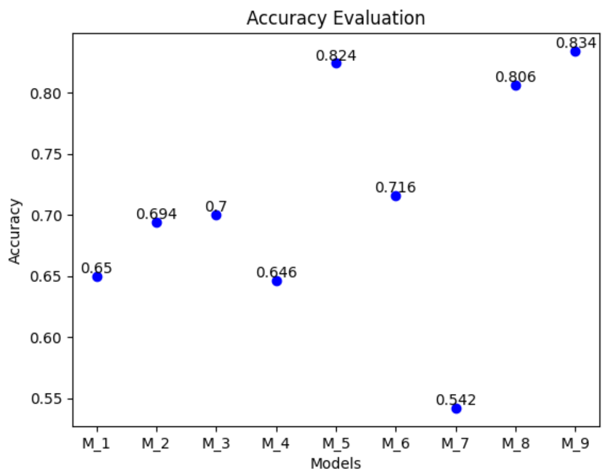
- Model 7 comprises 5 layers utilizing convolutional filters of size 3. The number of filters differs across these layers, ranging from 32 to 256. This configuration signifies a moderately deep convolutional neural network (CNN). Additionally, the training sample size was augmented from 2000 to 3000.

- Model 8 is constructed with a combination of strides and max pooling, which is considered desirable, resulting in an accuracy of approximately 80.6%. In contrast, in Model 9, padding was enabled, this adjustment impacted the performance of the model.

Conclusion:

In summary, among all the models, the ninth model achieves the highest accuracy at 83.4%, suggesting an improvement in performance with increased input data. Model 9 is characterized by a moderately deep CNN design, featuring tailored hyperparameters and dataset configurations. While its performance is commendable, it's evident that factors such as pooling and strides adjustments play pivotal roles in influencing accuracy levels.





Pre-Trained Network:

VGG-16 was utilized as the pretrained model for the image recognition task. Renowned for its robustness and extensive training, VGG-16 has been exposed to a vast and diverse dataset encompassing thousands, if not hundreds of thousands, of images across various categories. As a result, it stands as a formidable and globally recognized pretrained network widely employed for numerous image recognition tasks.

Model	Dense Layers	Optimizer	Training/ Validation/Test	Drop out	Pretrained weight update	Freeze Layers	Augmented images	Loss/ Accuracy
M-1	1(256 Nodes)	Rmsprop	1000/500/500	0.5	No	No	No	(4.531,0.960)
M-2	1(256 Nodes)	Rmsprop	1000/500/500	0.5	Yes	No	Yes	(6.376, 0.968)
M-3	1(256 Nodes)	Rmsprop	1000/500/500	0.5	Yes	Yes	Yes	(6.386, 0.964)
M-4	1(256 Nodes)	Adam	5000/500/500	0.5	Yes	No	Yes	(0.289, 0.994)
M-5	1(256 Nodes)	Adam(1e-5)	5000/500/500	0.5	Yes	Yes	Yes	(0.007,0.998)

- When employing pre-trained networks, the size of the training sample has surfaced as a crucial factor shaping the model's learning attributes and its efficacy in handling unseen data.

- Although rmsprop is recognized as a strong optimizer for constructing convolutional neural networks, Adam maintains an edge because of its distinct fusion of Momentum and rmsprop, which efficiently optimizes the neural network.

- By restraining the pre-trained network from modifying its weights, we safeguard the valuable knowledge embedded in the pre-existing model. Instead, the model's attention is directed towards training the densely connected classifier layer at the conclusion. This strategy has demonstrated its transformative impact, as seen in the enhanced performance of Models 2 and 3. Moreover, it serves as a valuable method for mitigating overfitting.

- When we freeze the first layers of pre-trained networks, we're basically keeping the parts that understand different types of images unchanged. This makes the model concentrate more on the particular task of recognizing the images we're interested in. It's been a successful strategy, especially for models like Model 3 and Model 5, which have shown the best accuracy among their groups.

Conclusion:

In the scenario of using pretrained networks, the tactic of immobilizing the initial layers has demonstrated its effectiveness in combating overfitting. By doing so, we prevent these layers from adjusting their weights during training, ensuring that the model doesn't simply memorize the training data but rather generalizes well to new, unseen data.

Expanding the sample size from 1000 to 5000 resulted in a notable improvement in the model's performance. Additionally, the model exhibited minimal loss, indicating its proficiency. Providing the model with a larger dataset, coupled with augmented versions of the images, proved to be an effective strategy for enhancing accuracy.