Data Mining Seminar

Benchmark Studies of Various Deep Learning Architecture

Irfan Nur Afif (1035476)

Background

"How does the performance comparison of deep learning architecture model looks like for a given image classification dataset?"

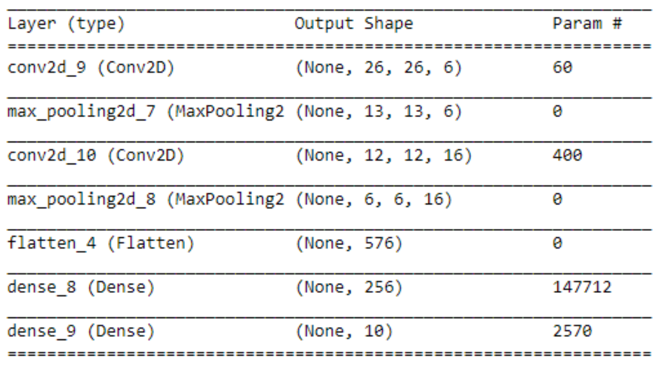
Sub-question:

1. Which architecture that works bests for a given datasets?
2. What kind of datasets characteristics that makes a deep learning architecture works well?
3. Is there any architecture that generally works well for image classification?

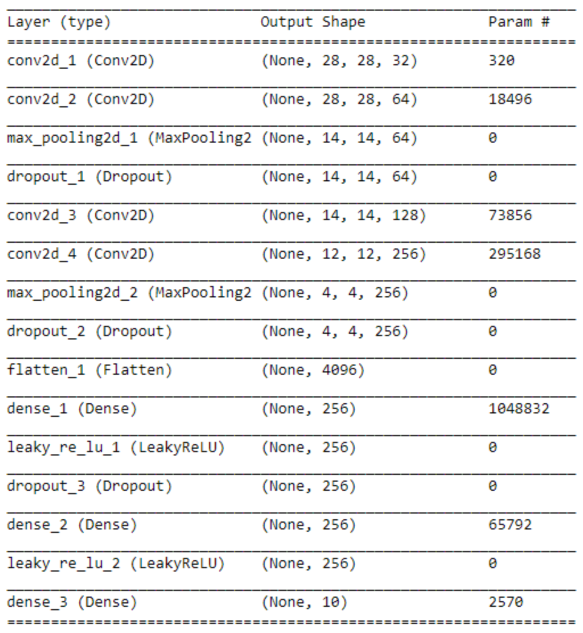
Implementation

Architecture

LeNet, VGG-Net, ResNet20V1, ResNet20V2, SqueezeNet/Alex-Net



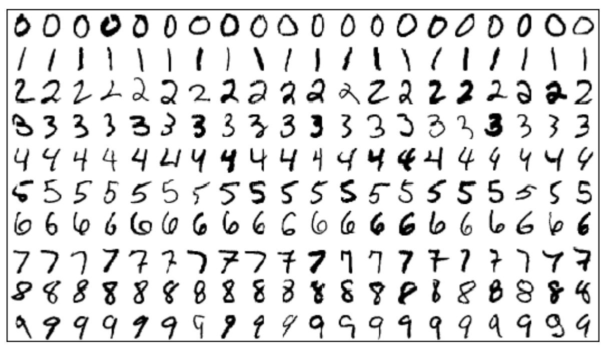
```
_________________________________________________________________
Layer (type)                 Output Shape              Param #
=================================================================
conv2d_9 (Conv2D)            (None, 26, 26, 6)         60
_________________________________________________________________
max_pooling2d_7 (MaxPooling2 (None, 13, 13, 6)         0
_________________________________________________________________
conv2d_10 (Conv2D)           (None, 12, 12, 16)        400
_________________________________________________________________
max_pooling2d_8 (MaxPooling2 (None, 6, 6, 16)          0
_________________________________________________________________
flatten_4 (Flatten)          (None, 576)               0
_________________________________________________________________
dense_8 (Dense)              (None, 256)               147712
_________________________________________________________________
dense_9 (Dense)              (None, 10)                2570
=================================================================
```

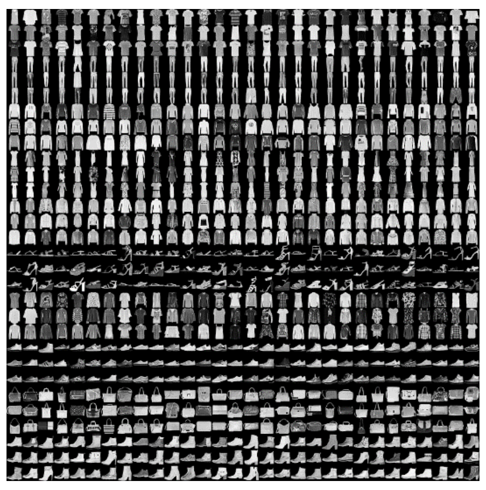


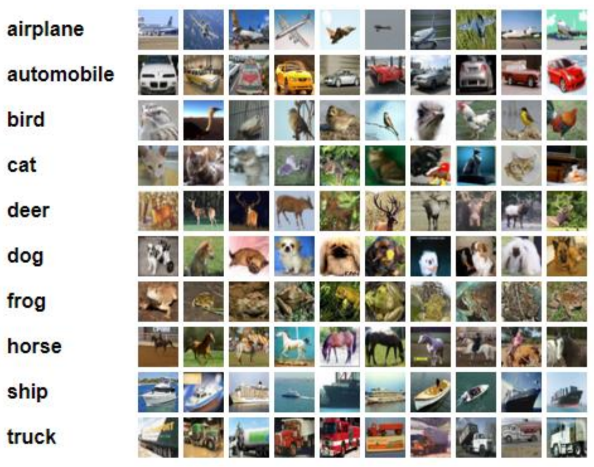
```
_________________________________________________________________
Layer (type)                 Output Shape              Param #
=================================================================
conv2d_1 (Conv2D)            (None, 28, 28, 32)        320
_________________________________________________________________
conv2d_2 (Conv2D)            (None, 28, 28, 64)        18496
_________________________________________________________________
max_pooling2d_1 (MaxPooling2 (None, 14, 14, 64)        0
_________________________________________________________________
dropout_1 (Dropout)          (None, 14, 14, 64)        0
_________________________________________________________________
conv2d_3 (Conv2D)            (None, 14, 14, 128)       73856
_________________________________________________________________
conv2d_4 (Conv2D)            (None, 12, 12, 256)       295168
_________________________________________________________________
max_pooling2d_2 (MaxPooling2 (None, 4, 4, 256)         0
_________________________________________________________________
dropout_2 (Dropout)          (None, 4, 4, 256)         0
_________________________________________________________________
flatten_1 (Flatten)          (None, 4096)              0
_________________________________________________________________
dense_1 (Dense)              (None, 256)               1048832
_________________________________________________________________
leaky_re_lu_1 (LeakyReLU)    (None, 256)               0
_________________________________________________________________
dropout_3 (Dropout)          (None, 256)               0
_________________________________________________________________
dense_2 (Dense)              (None, 256)               65792
_________________________________________________________________
leaky_re_lu_2 (LeakyReLU)    (None, 256)               0
_________________________________________________________________
dense_3 (Dense)              (None, 10)                2570
=================================================================
```

Dataset:

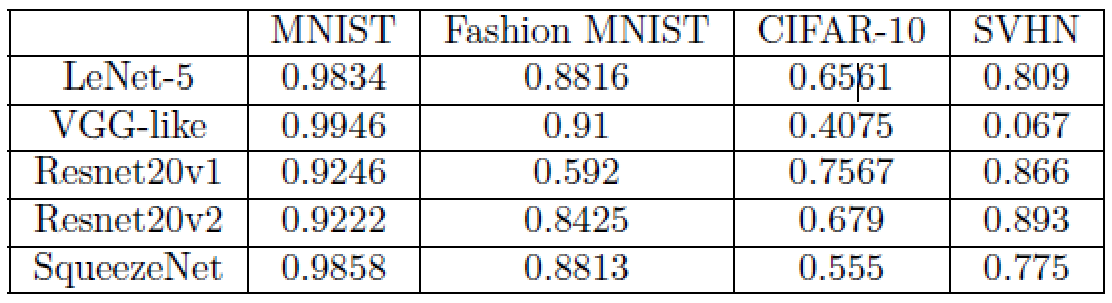
MNIST, Fashion-MNIST, CIFAR-10, SVHN





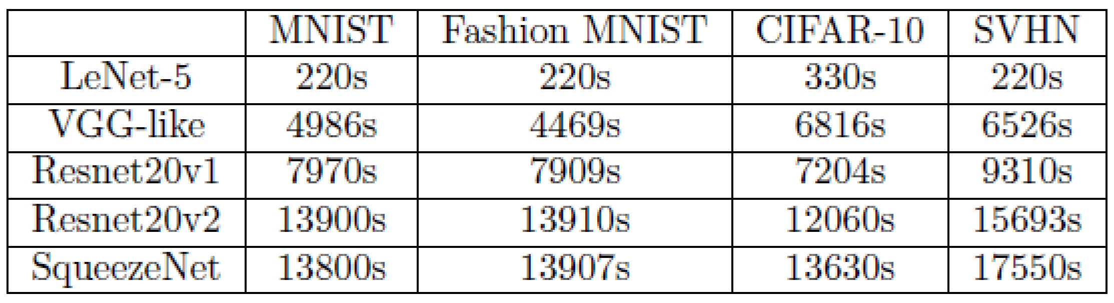


Result



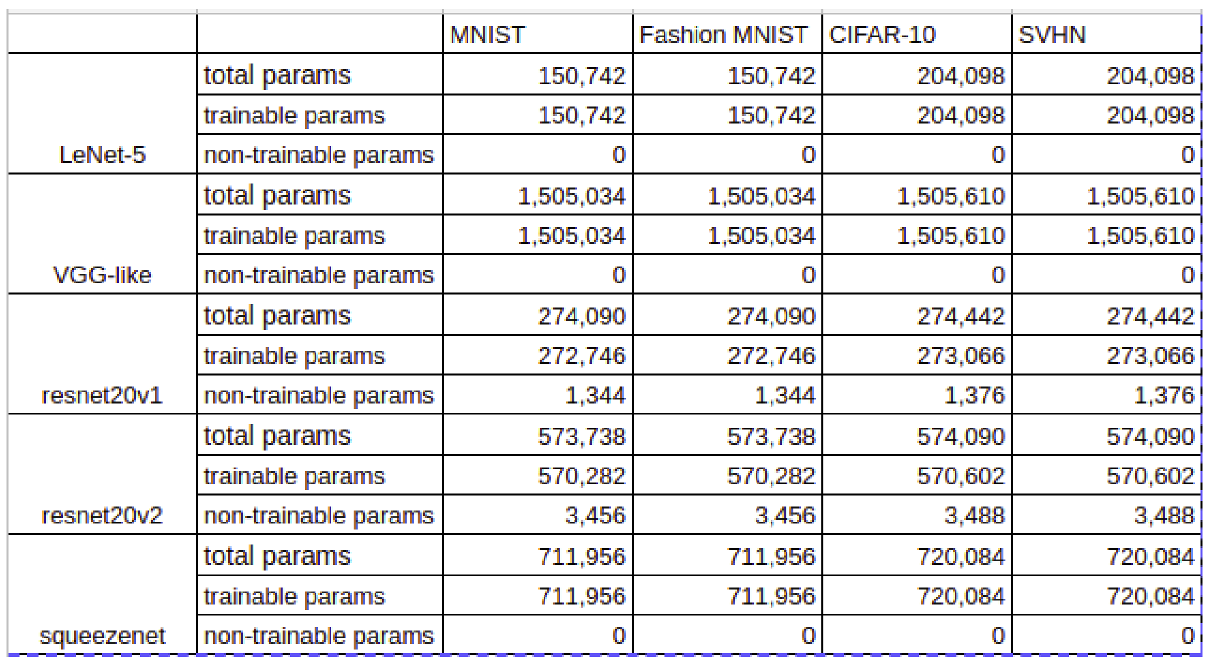
	MNIST	Fashion MNIST	CIFAR-10	SVHN
LeNet-5	0.9834	0.8816	0.6561	0.809
VGG-like	0.9946	0.91	0.4075	0.067
Resnet20v1	0.9246	0.592	0.7567	0.866
Resnet20v2	0.9222	0.8425	0.679	0.893
SqueezeNet	0.9858	0.8813	0.555	0.775

Table 4.1: Accuracy table



	MNIST	Fashion MNIST	CIFAR-10	SVHN
LeNet-5	220s	220s	330s	220s
VGG-like	4986s	4469s	6816s	6526s
Resnet20v1	7970s	7909s	7204s	9310s
Resnet20v2	13900s	13910s	12060s	15693s
SqueezeNet	13800s	13907s	13630s	17550s

Table 4.2: Execution time table



		MNIST	Fashion MNIST	CIFAR-10	SVHN
LeNet-5	total params	150,742	150,742	204,098	204,098
	trainable params	150,742	150,742	204,098	204,098
	non-trainable params	0	0	0	0
VGG-like	total params	1,505,034	1,505,034	1,505,610	1,505,610
	trainable params	1,505,034	1,505,034	1,505,610	1,505,610
	non-trainable params	0	0	0	0
resnet20v1	total params	274,090	274,090	274,442	274,442
	trainable params	272,746	272,746	273,066	273,066
	non-trainable params	1,344	1,344	1,376	1,376
resnet20v2	total params	573,738	573,738	574,090	574,090
	trainable params	570,282	570,282	570,602	570,602
	non-trainable params	3,456	3,456	3,488	3,488
squeezenet	total params	711,956	711,956	720,084	720,084
	trainable params	711,956	711,956	720,084	720,084
	non-trainable params	0	0	0	0

Table 4.3 Parameters usage

Discussion

1. There are not a single architecture that gives the best performance across all tested dataset.
2. LeNet is the most simple yet powerful network. This networks can become the tester network for ian image dataset because of its low time to train and memory to use.
3. Simplied VGGNet performs well on grayscale dataset compared to colored-dataset. Also it performs better on smaller image.
4. ResNetV1 works better in digit recognition compared to object recognition.
5. ResNetV2 does not always increase the performance of ResNetv1. However, on Fashion MNIST dataset, the improvemance is quite huge, around 25% accuracy.
6. ResNetV2 gives quite good and stable performance compared to other network.
7. SqueezeNet consumes the most memory and time to train but does not yields the best classication performance on any dataset.

Faculty of Mathematics and Computer Science