Animal Classification

George Washington Univeristy
Machine Learning II
Shuting Cai
Jinbo Li
Kaiyuan Liang

Table Of Contents

DATA OVERVIEW

About Dataset

- Animal Image Classification Dataset

  (https://www.kaggle.com/datasets/piyushkumar18/animal-image-classification-dataset)

- DATA Description

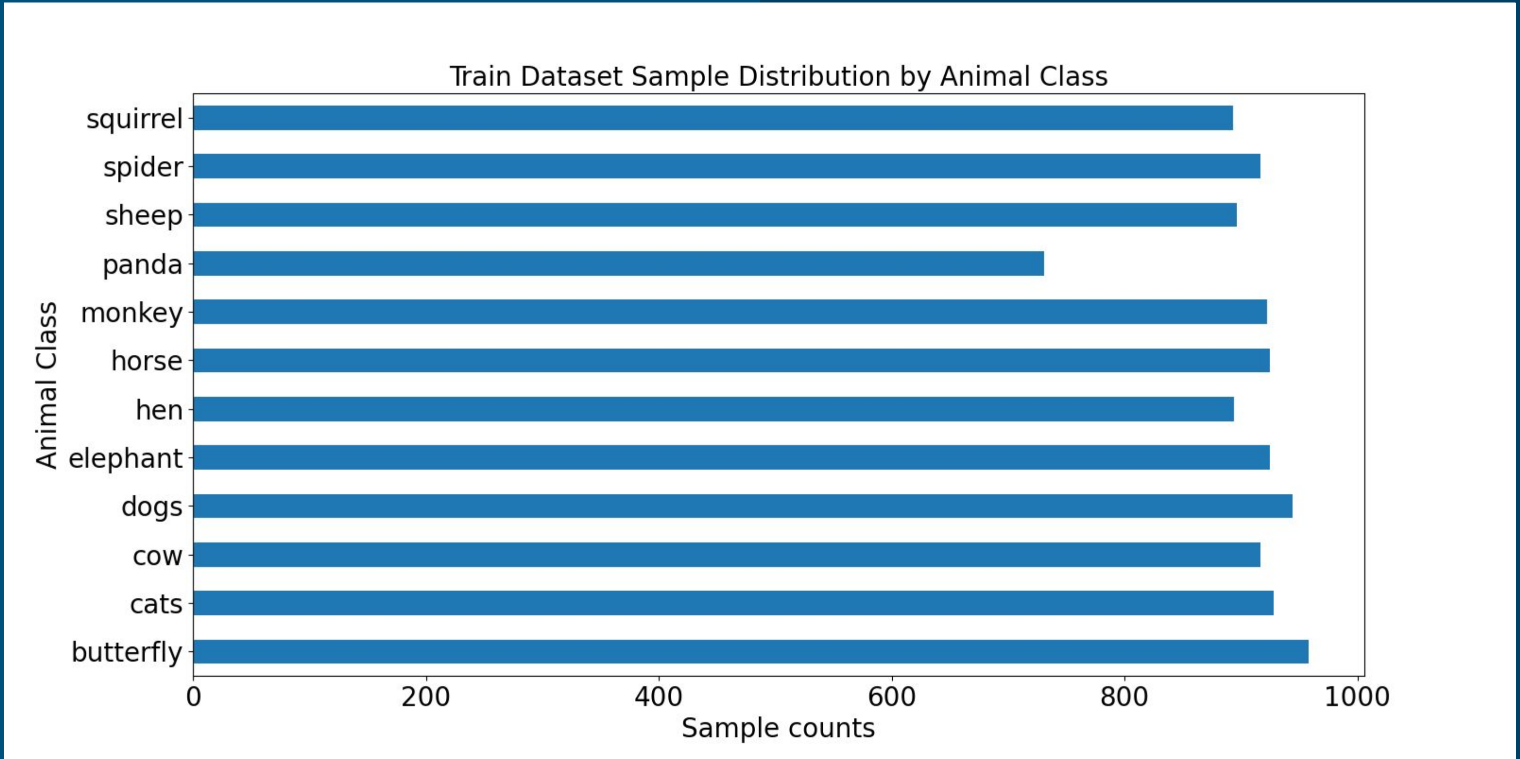
- Target Class: 12 Categories

[Class names: butterflies, cats, cows, dogs, elephants, hens, horses, monkeys, pandas, sheep, spiders, and squirrel]

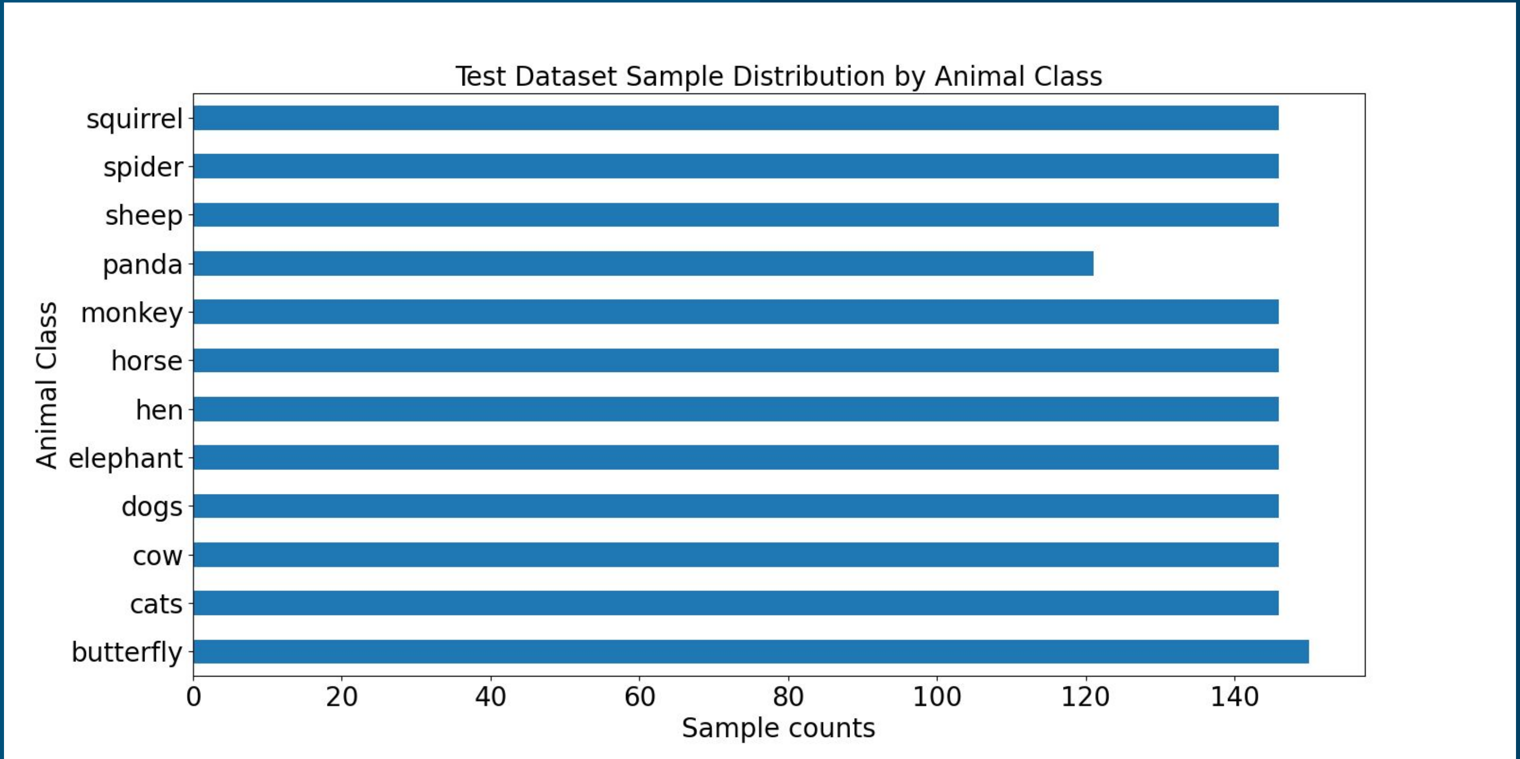
- Each class of animal has over 1200 image files

- The images have large variations in scale, pose, background and lighting.

DATA OVERVIEW



DATA OVERVIEW



DATA PREPROCESSING

DATA Generation

- Split into training dataset and testing dataset by using split-folders

  [a ratio of 9 to 1]

- Split into training and validation datasets with `tf.keras.utils.image_dataset_from_directory`

  [split the 3 over 7 from training dataset for validation]

- Data augmentations

  [image resize, horizontal flipped, random rotation, image zoom…]

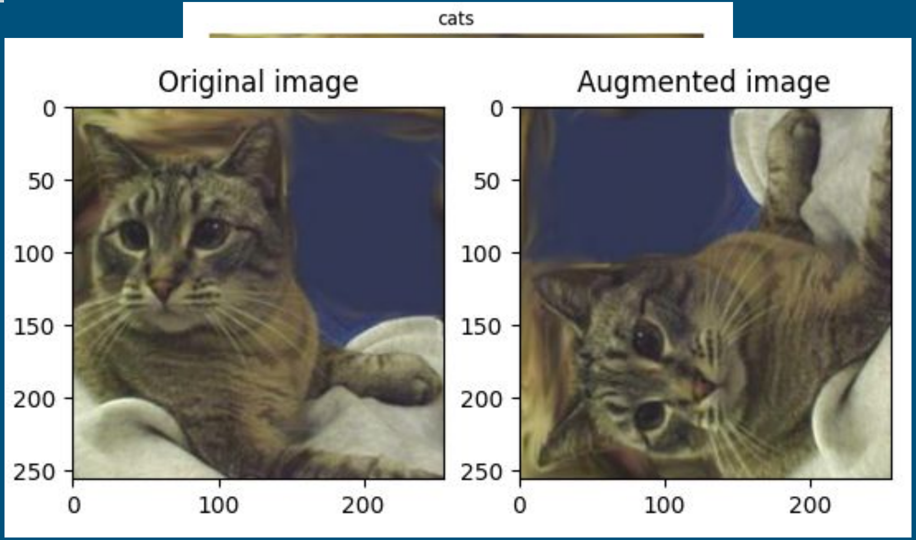
Data Augmentation

- Horizontal Flipped `tf.keras.layers.RandomFlip("horizontal")`
- Rotation 90。 `tf.keras.layers.RandomRotation()`
- Randomzoom `tf.keras.layers.RandomZoom()`

Is it really worth the effort?

Strength Training

Data Bias

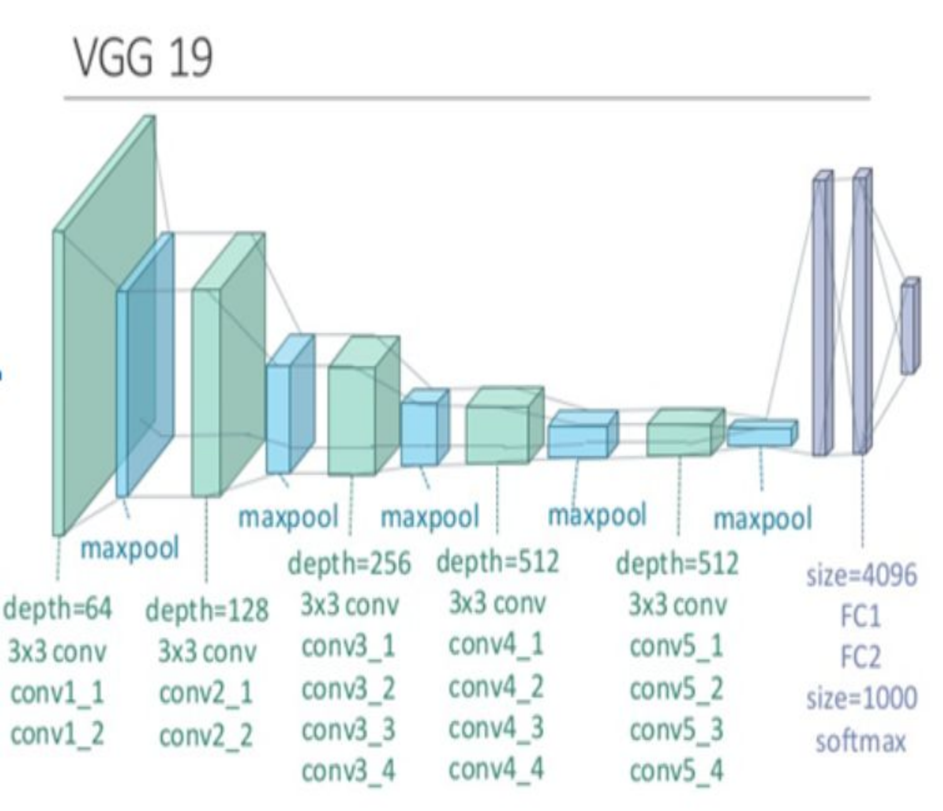


MODELS

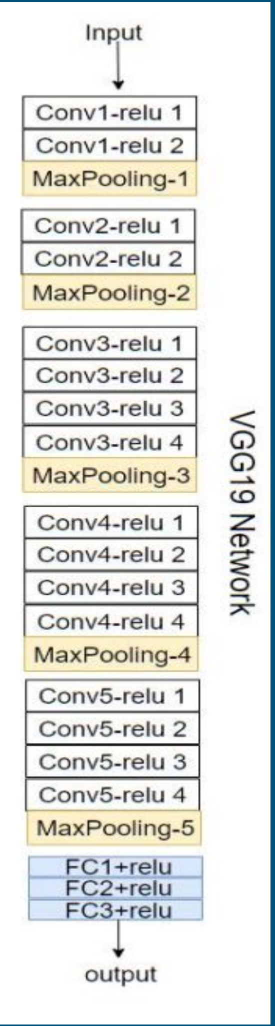
CNNs - ARCHITECTURE

- Convolutional layers
- Pooling Layers
  - Reduce data dimension and amount of computation
  - Avoid overfitting
- Fully Connected Layers

Pretrained Model VGG19

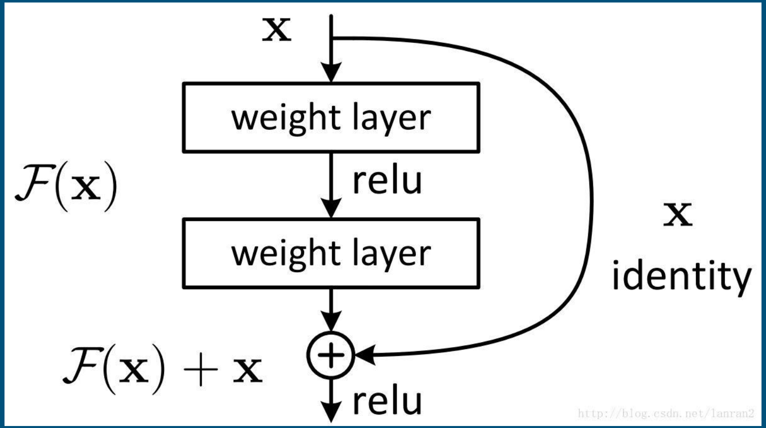


- Takes the input size as 224 by 224 with 3 RGB channels
- used kernels of 3 * 3 size with a stride size of 1 pixel
- max pooling was performed over 2 * 2 pixel windows with stride 2



Pretrained Model ResNet50

- 50 neural network layers
- Deep Residual Networks
  - residual blocks - Skip Connections
  - Residual mapping $\rightarrow$ 0
  - Identity mapping
- Higher accuarcy than ResNet34



output: y = F(x) + x

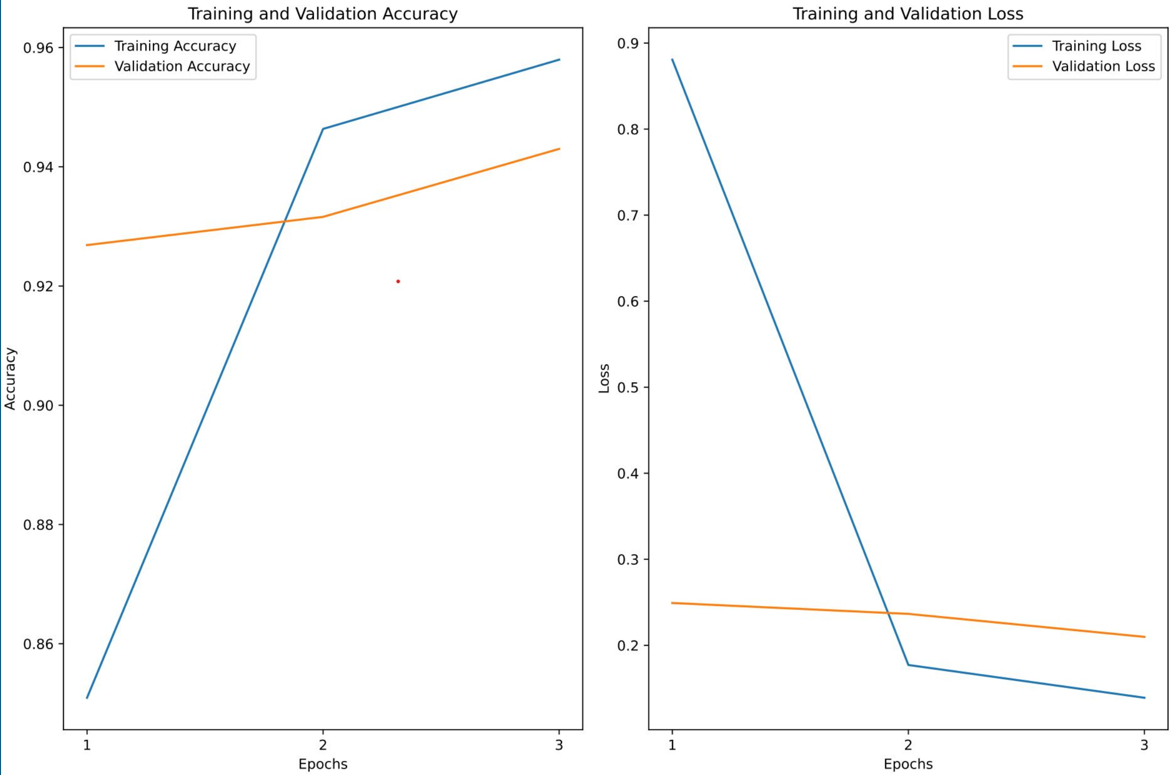
residual: F(x) = y - x

Customized Model Networks

- 15 neural network layers
- Data-Augugmentation
- Con2D layer
- BatchNormalizaition and Rescaling
- Dropout
- Output layer(Dense) with softmax activation

Overfitting & Underfitting

- Underfitting:
  - Dataset not large
  - Network not powerful
- Overfitting
  - Fails to generalize the testing



MODEL EVALUATIONS

MODEL EVALUATION

- List of metrics

- Accuracy score
- Cohen kappa score(>0.8)
- Micro F1 score
- Hamming loss

Results

Model	Accuracy score	Cohen kappa score	Micro F1 score	Avg of Metrics
VGG	0.91	0.90	0.91	0.91
Resnet	0.93	0.92	0.93	0.93
Customized	0.26	0.19	0.26	0.23

CONCLUSION

- BEST MODEL: ResNet50
- SETTING & Hyper-PARAMETER:
    - random_seed = 42
    - batch_size = 64
    - epochs = 3
    - lr = 0.01
    - img_height = 256
    - img_width = 256
    - channel = 3

FURTURE WORKS/Improvement

- Buid the model on the larger memories machine without worring ‘out of memmory issue’
- Test various hyper parameters settings(eg:learning rate, train/val size) to optimize results
- Try different pretrained models(eg:googlenet,efficientet)

REFERENCE

Gandhi, A. (2021, May 20). Data augmentation: How to use deep learning when you have limited data. AI & Machine Learning Blog. Retrieved April 26, 2022, from https://nanonets.com/blog/data-augmentation-how-to-use-deep-learning-when-you-have-limited-data-part-2/

Kaiming H., Xiangyu Z. (2016). Deep Residual Learning for Image Recognition. Retrieved April 26, 2022, from

https://www.cv-foundation.org/openaccess/content_cvpr_2016/papers/He_Deep_Residual_Learning_CVPR_2016_paper.pdf

Thank you!