Stanford Dogs Dataset

By: Ryan King

Background and Motivation

- As it grows in digital importance, I wanted to delve deeper into image classification
- Identifying dogs breeds is difficult for most humans due to tiny physical similarities and differences
- Accurate identification typically requires extensive knowledge or experience, which is uncommon
- The Stanford Dogs dataset is specifically tailored to focus on dog breed classifications

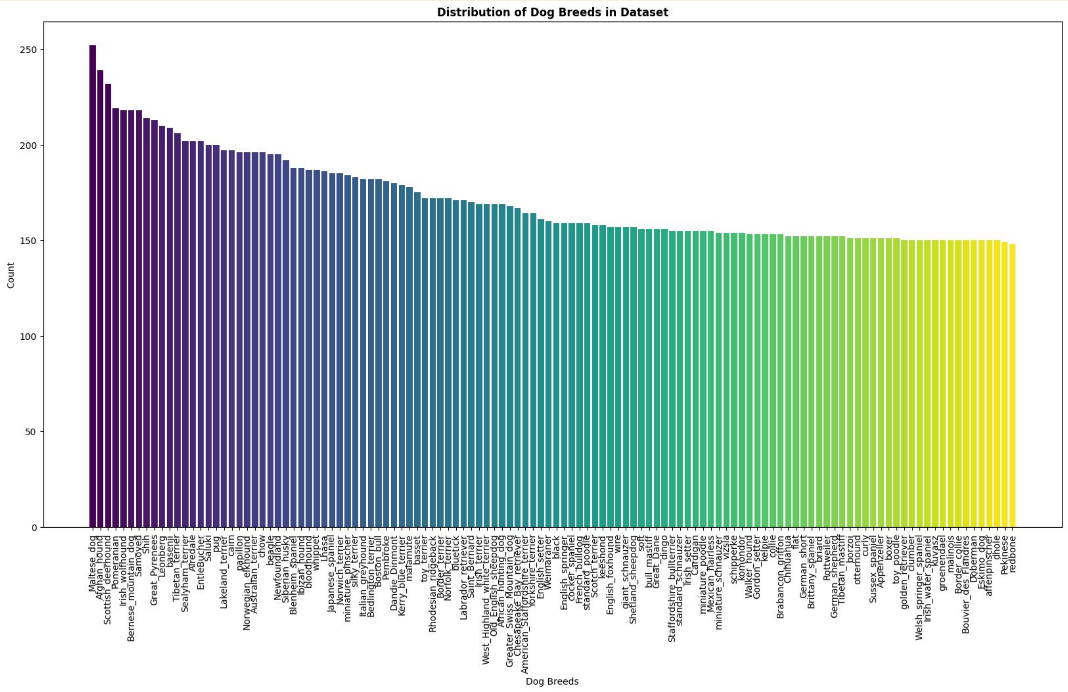
Project Objective

- The main objective of the model is to correctly identify a dog's breed from an image
    - A successful version of the model would produce a high level of accuracy for all different types of dogs
- The model should:
    - Have the ability to differentiate between fine-grained classes (120 dog breeds)
    - Perform well on unseen images, ensuring model robustness
    - Classify breeds quickly and effectively
    - Showcase a strong performance in recognizing subtle differences among dog breeds

Data Overview

- Dataset derived from ImageNet for fine-grained image classification
- Dataset contains 20,580 JPEG images
- Images of 120 unique dog breeds from across the world
    - Approximately 150-200 images per dog breed class
- Along with the images, the dataset provides annotations categorizing the different breeds of dogs

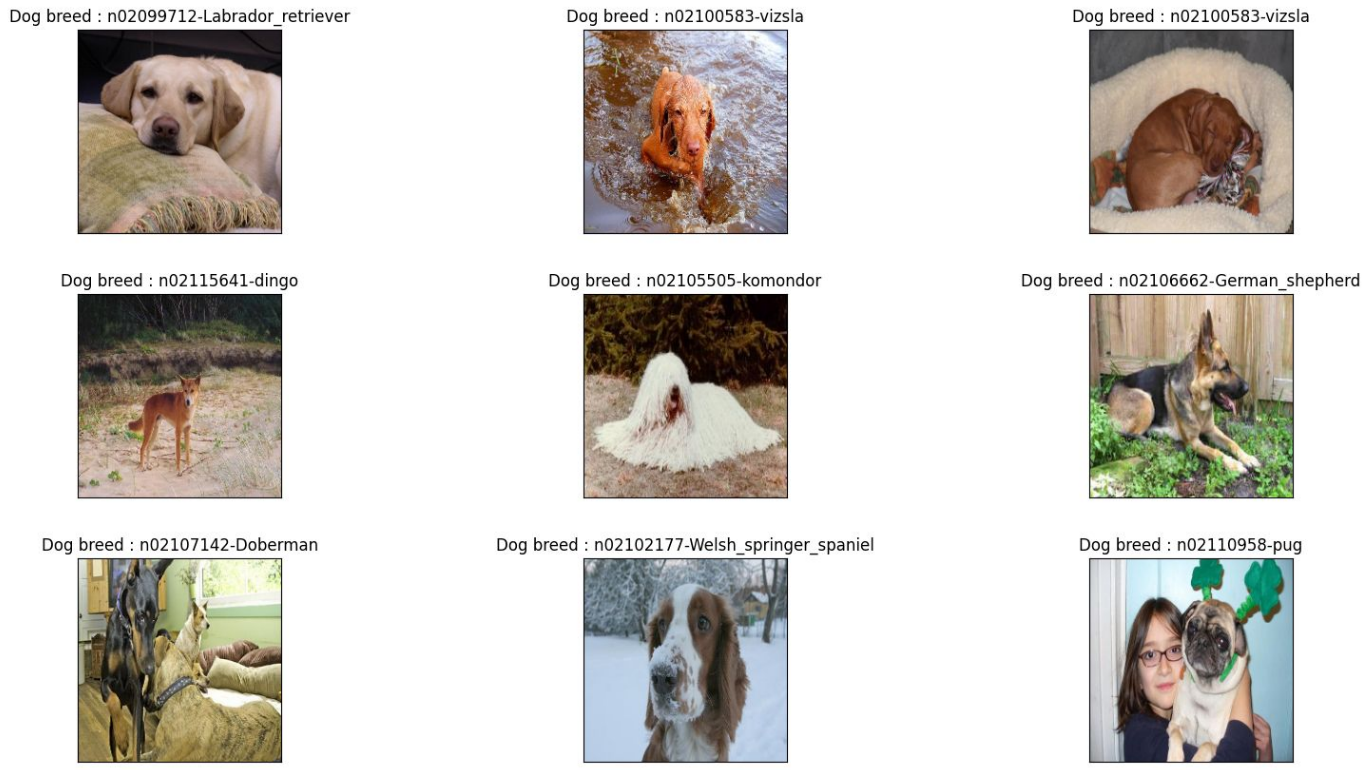
Distribution of Dog Breeds



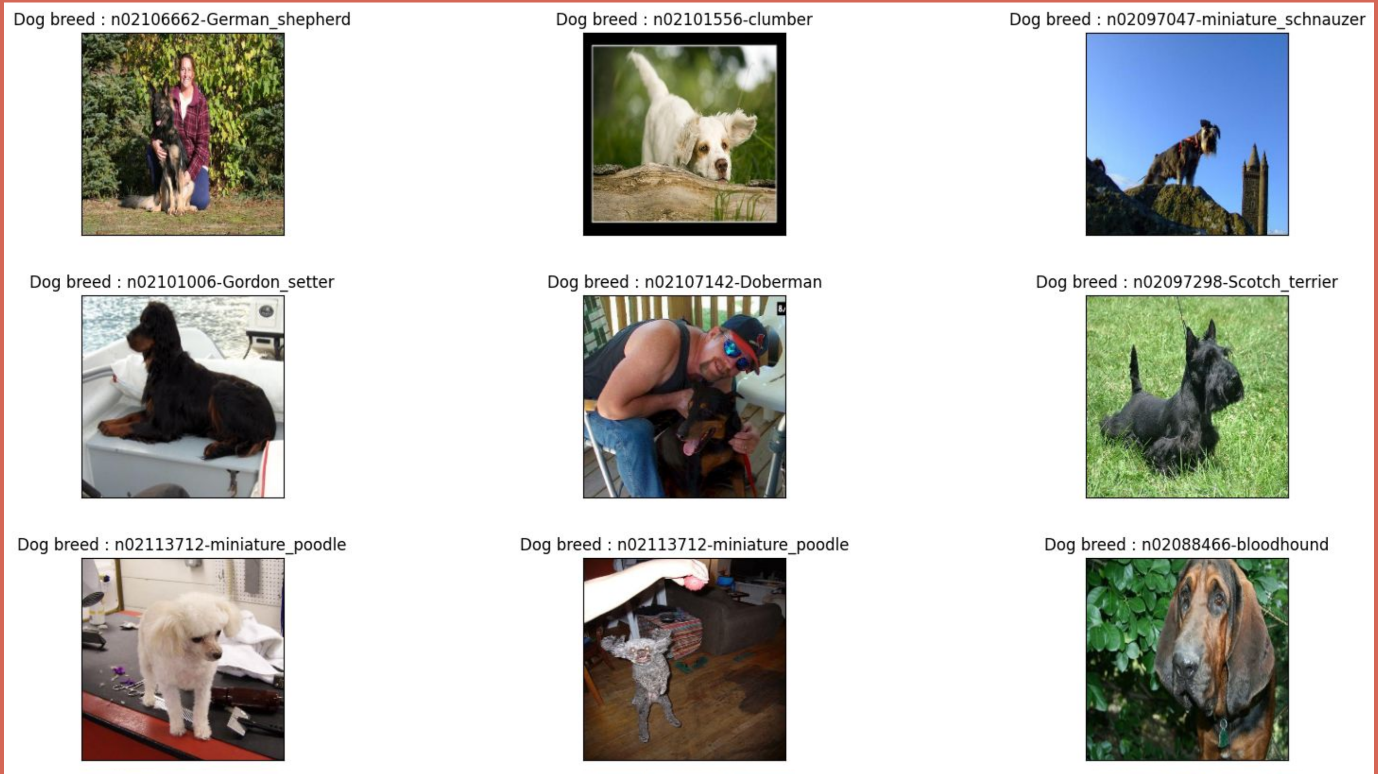
Top 10 Dog Breeds in Dataset

	Breed	Count
0	Maltese_dog	252
1	Afghan_hound	239
2	Scottish_deerhound	232
3	Pomeranian	219
4	Irish_wolfhound	218
5	Bernese_mountain_dog	218
6	Samoyed	218
7	Shih	214
8	Great_Pyrenees	213
9	Leonberg	210

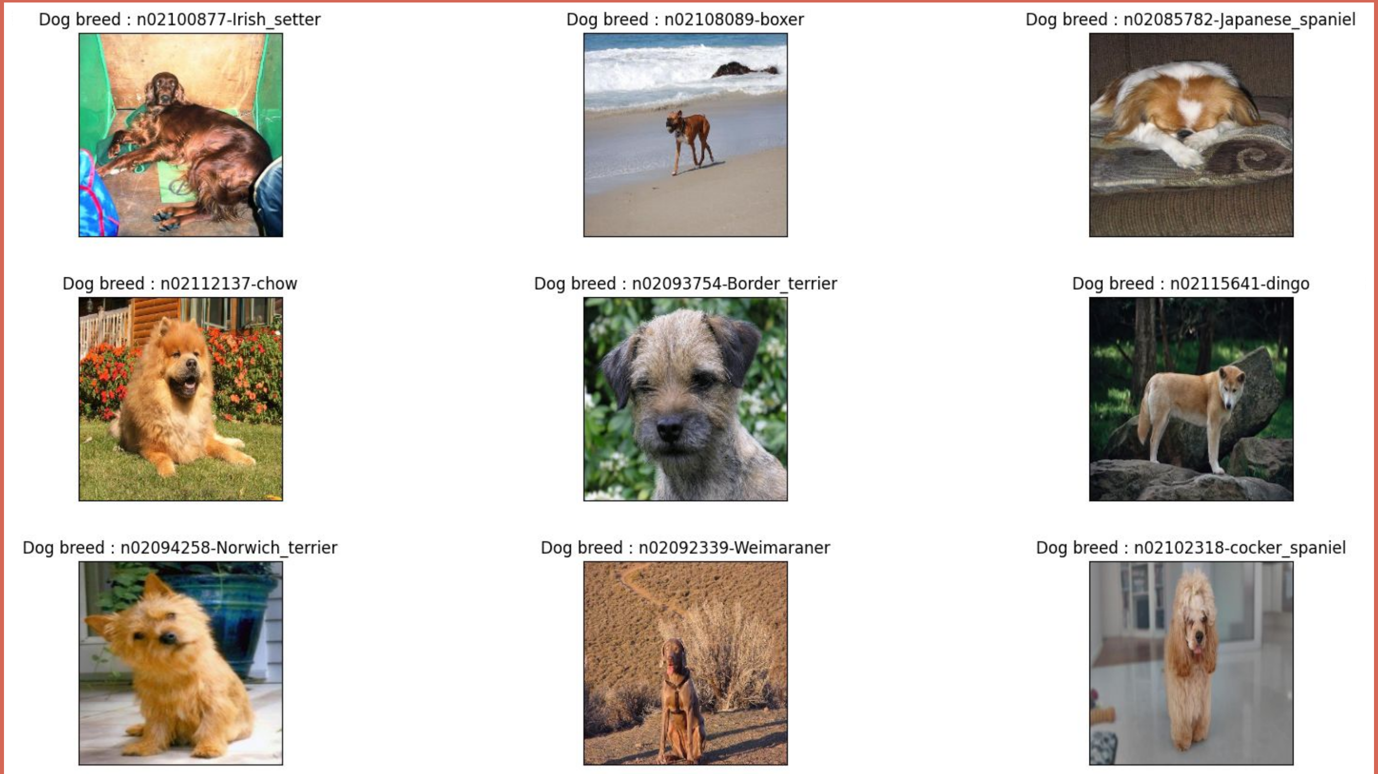
Images in the Dataset



Images in the Dataset



Images in the Dataset



Data Preprocessing and Augmentation

- Used Keras ImageDataGenerator to handle the image scaling and augmentations
- Data Augmentations: Applied random horizontal flips, width and height changes, shear transformations, and rotations to training images
- Defined a consistent image size of 224x224 pixels

Data Preprocessing and Augmentation

- Set up the model to train with batches of 24 images to ensure efficient use of memory during the training process
- Split the data, reserving 20% for the validation set to prevent overfitting
- Implemented one-hot encoding in data generators
  - Essential for training the model on multi-class classification using categorical cross-entropy

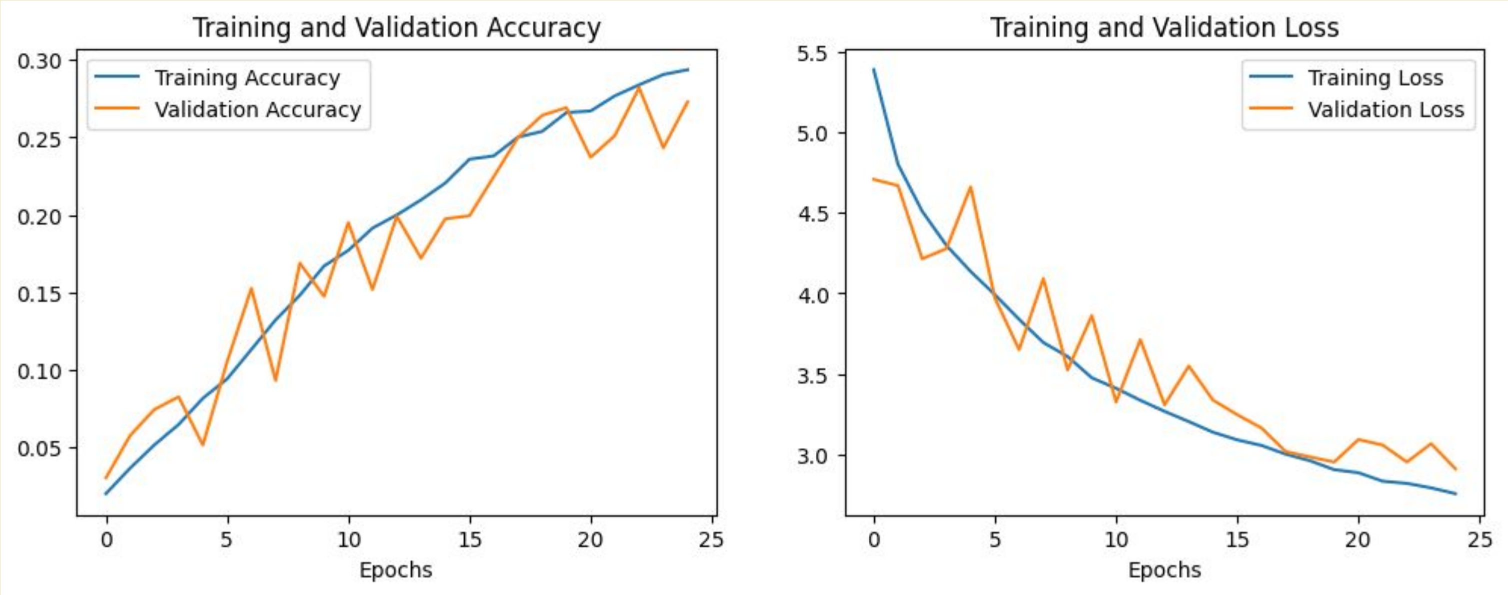
CNN Model Architecture

- Explored various general CNN models, experimenting with different architectures and parameters
  - Most of these models yielded poor loss and accuracy
- Architecture: Sequential Conv2D (16, 32, 64, 128 filters) with BatchNormalization and MaxPooling2D
- Followed by Flattening, a 512 neuron Dense layer with ReLU activation, BatchNormalization, and Dropout
- Output Layer: Softmax-activated Dense layer for classification
- Compiling: Utilized Adam optimizer, categorical crossentropy for loss and accuracy as performance metric

CNN Model Results

- Training Progress:
  - Loss reduced from 5.39 to 2.76 (Epoch 25)
  - Accuracy improved from 2.01% to 29.37%
- Validation Performance:
  - Loss reduced from 4.71 to 2.91 (Epoch 25)
  - Accuracy increased from 3.03% to 27.29%
- Significant improvement in loss and accuracy over epochs, indicating effective learning
- Model could definitely be improved possibly by changing some of the hyperparameters or using other regularization methods (L1 or L2)

CNN Model Metric Plots



Pre-Trained Models

- Unsuccessful general CNN models led me to using pre-trained models like VGG16, and InceptionV3
- VGG16 model pre-trained on ImageNet:
  - Experimented by implementing L2 regularization and dropout
  - Train accuracy increases from 69.66% to 77.56% but the validation accuracy does not consistently improve, peaking at 53.72%, indicating that the model is overfit
- After, moved on InceptionV3 based models

Pre-Trained Model Architecture

- InceptionV3 Architecture: Uses the InceptionV3 base, pretrained on ImageNet
- Middle Layers: Contains a Global Average Pooling layer followed by a Dense layer with 1024 neurons and a ReLU activation
- Output Layer: Consists of a Dense layer with a softmax activation for multi-class classification
- Compiling: Used Adam optimizer (LR: 0.0001) and the model focused on minimizing categorical cross entropy

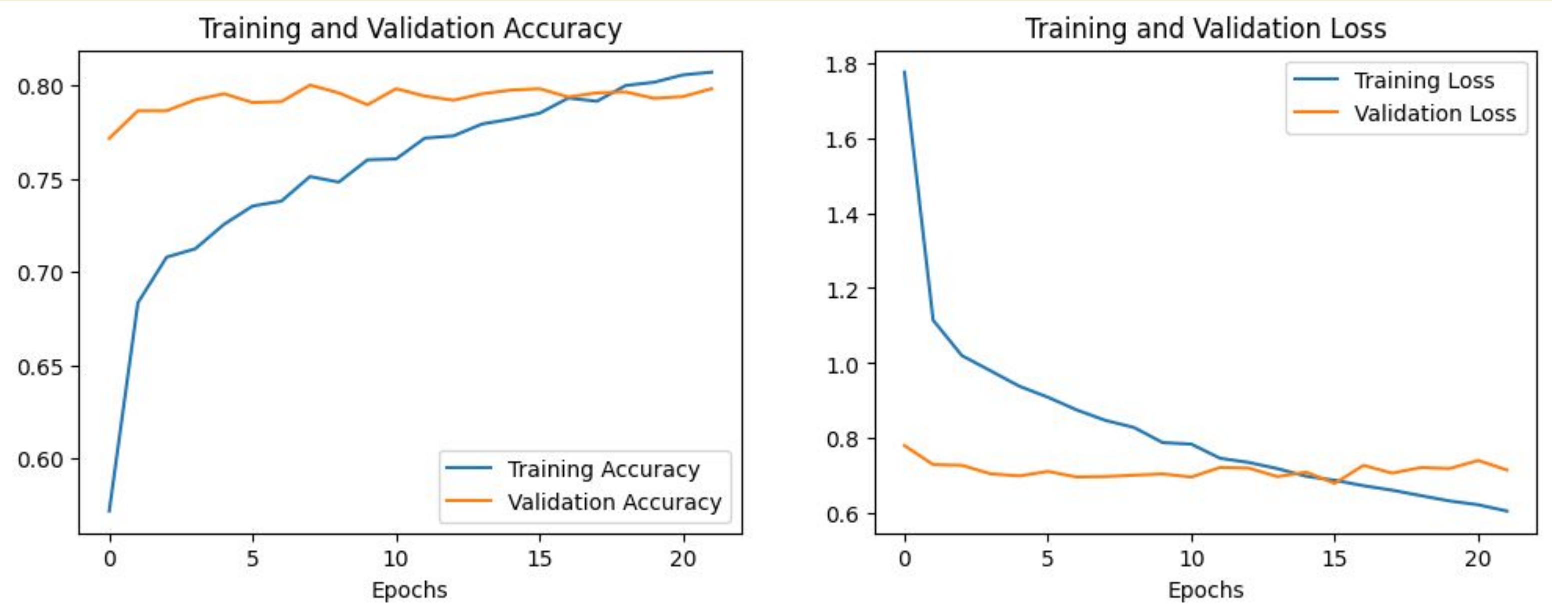
Pre-Trained Model Results

- Training Performance:
  - Started with a loss of 1.77 and an accuracy of 57.19%
  - Gradual decrease in loss and increase in accuracy over 22 epochs
  - Ended with a loss of 0.61 and accuracy of 80.72%
- Validation Performance:
  - Initial loss was 0.78 and accuracy of 77.17%
  - Small fluctuations in loss and accuracy across epochs
  - Maintained a stable accuracy throughout ending at 79.91% with the loss being 0.72

Pre-Trained Model Results Cont.

- These metrics reflect how well the model is generalizing to new data and its ability to correctly classify that data
- Insights:
    - Model displays a good generalization to unseen data with a consistent validation accuracy
    - Close alignment between training and validation accuracy suggests the model has balance between overfitting and underfitting

Pre-Trained Metric Plots



Grad CAM

GradCAM - Boston Bull

```
Top 5 Predictions:
n02096585-Boston_bull: 0.9666
n02087046-toy_terrier: 0.0266
n02091032-Italian_greyhound: 0.0018
n02110806-basenji: 0.0014
n02085620-Chihuahua: 0.0010
```

GradCAM - Scottish Deerhound

```
Top 5 Predictions:
n02092002-Scottish_deerhound: 0.9643
n02090721-Irish_wolfhound: 0.0357
n02105505-komondor: 0.0000
n02104029-kuvasz: 0.0000
n02090622-borzoi: 0.0000
```

GradCAM - Collie

```
Top 5 Predictions:
n02106030-collie: 0.8254
n02106166-Border_collie: 0.1736
n02107908-Appenzeller: 0.0004
n02104029-kuvasz: 0.0001
n02109525-Saint_Bernard: 0.0001
```

Mixed Breeds

GradCAM - Bernedoodle

```
Top 5 Predictions:
n02107683-Bernese_mountain_dog: 0.6377
n02088094-Afghan_hound: 0.1108
n02108000-EntleBucher: 0.0733
n02097474-Tibetan_terrier: 0.0622
n02101006-Gordon_setter: 0.0354
```

GradCAM - Labradoodle

```
Top 5 Predictions:
n02113712-miniature_poodle: 0.5416
n02113799-standard_poodle: 0.3692
n02113624-toy_poodle: 0.0524
n02102973-Irish_water_spaniel: 0.0188
n02099429-curly-coated_retriever: 0.0138
```

GradCAM - German Shepherd Lab

```
Top 5 Predictions:
n02106662-German_shepherd: 0.9340
n02106550-Rottweiler: 0.0271
n02093754-Border_terrier: 0.0164
n02112706-Brabancon_griffon: 0.0068
n02105412-kelpie: 0.0062
```

Thank you!