Animal Classification Using CNN

1. Project Structure

```
Animal_Classification_Project/
 train/                    # Training images
 validation/               # Validation images
 animal_classifier_model.h5 # Saved trained model
 predict.py                # Prediction script
 train_model.py            # Training script
 requirements.txt          # List of libraries
```

2. Features

- Image classification using Convolutional Neural Networks (CNN).

- Augmented image preprocessing for better generalization.

- Real-time prediction from file path input.

- Graphical visualization using Matplotlib.

- Evaluation through accuracy, confusion matrix, and classification report.

3. Model Architecture

Sequential CNN Architecture:

- Conv2D(32) -> MaxPooling2D

- Conv2D(64) -> MaxPooling2D

- Conv2D(128) -> MaxPooling2D

- Flatten -> Dropout(0.5)

- Dense(512) -> Dense(1, activation='sigmoid')

- Loss Function: binary_crossentropy

- Optimizer: Adam (lr=0.001)

4. How to Run

1. Install dependencies:

```
   pip install tensorflow matplotlib scikit-learn
```

2. Train model:

```
   python train_model.py
```

3. Predict an image:

```
   python predict.py
```

5. Sample Prediction

```
img_path = "path_to_image.jpg"

print(f"Predicted animal: {predict_animal(img_path)}")

Output: Predicted animal: dolphin
```

6. Results

Training Accuracy: 0.00 (due to incorrect setup)

Validation Accuracy: 1.00 (likely due to configuration issues)

Confusion Matrix:

```
 [[2500]]
```

Classification Report:

```
  precision  recall  f1-score  support

  cats       1.00     1.00     1.00     2500
```

7. Libraries Used

- TensorFlow
- NumPy
- Matplotlib
- scikit-learn

8. Author

Shweta Pattewar