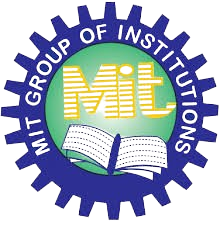
**MORADABAD INSTITUTE OF TECHNOLOGY, MORADABAD**  
  
Department of Computer Science & Engineering

  
  
PROJECT REPORT ON

**IMAGE CLASSIFICATION OF CATS AND DOGS USING CNN**

Submitted by:

Tanishka Ruhela

B.Tech – Computer Science and Engineering

Semester: 7th

Submitted to:

Samarth Sir

IBM Project-Based Experiential Learning Program

# Acknowledgment

I would like to express my heartfelt gratitude to **Mr. Samarth Amruthe** for his exceptional guidance and support throughout my project on **Image Classification of Cats and Dogs**.

I am deeply thankful to **Kaggle** and **TensorFlow** for providing the essential datasets and tools that greatly contributed to the successful execution of this project.

I also appreciate the role of the **Image Classification model**, which offered advanced functionalities and technical insights for accurately distinguishing between images of cats and dogs. Its contribution ensured consistent and reliable results.

Lastly, I am truly grateful to my **family and friends** for their constant encouragement and support throughout this journey.

Tanishka Ruhela  
B.Tech – Computer Science and Engineering  
MIT, Moradabad

# Problem Statement

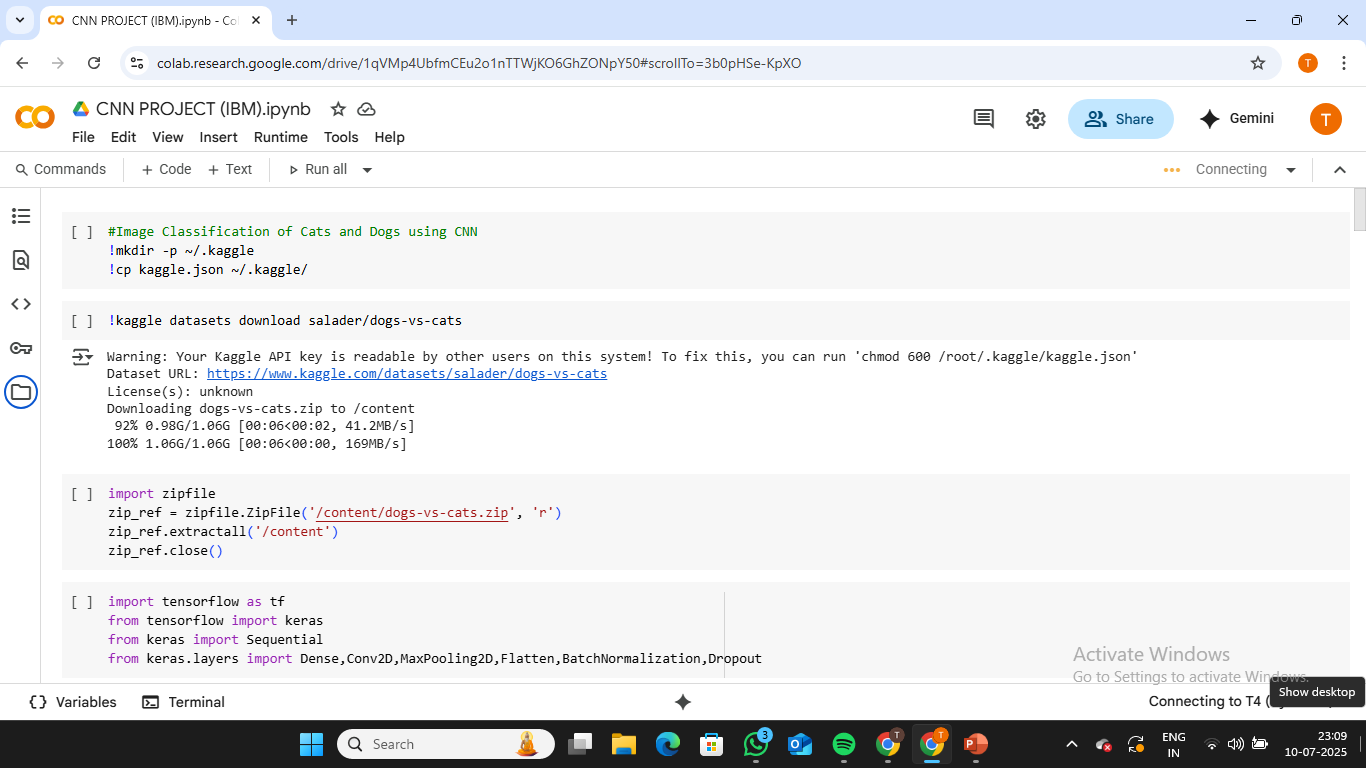
The objective is to develop a Convolutional Neural Network (CNN) model that accurately classifies images as either a cat or a dog. Although this may seem easy for humans, machines face challenges due to varying image conditions, such as lighting, background, and animal posture. The aim is to build a robust binary image classifier using deep learning techniques.

# Methodology

- The Dogs vs Cats dataset was downloaded from Kaggle.  
- Images were resized to 256x256 and normalized.  
- Built a CNN with 4 Conv blocks followed by Dense layers.  
- Used Adam optimizer and binary crossentropy loss.  
- Trained for 10 epochs on Google Colab.  
- Evaluated with accuracy/loss graphs and sample predictions.

# Dataset Details

• Source: Kaggle (Dogs vs Cats)  
• Format: Labeled images in train/test folders  
• Classes: Cat, Dog  
• Image Size: 256 x 256  
• Batch Size: 32



# Algorithms/Models Used

The CNN consists of:  
• Conv2D → BatchNorm → MaxPooling (repeated 4 times)  
• Flatten layer  
• Dense(128) → Dropout(0.25)  
• Dense(64) → Dropout(0.25)  
• Dense(1, sigmoid)  
  
• Activation Functions: ReLU (hidden), Sigmoid (output)  
• Loss Function: Binary Crossentropy  
• Optimizer: Adam (lr=0.0003)  
• Regularization: Dropout to reduce overfittingResults

The model trained over 10 epochs with the following metrics:  
• Training Accuracy: ~83%  
• Validation Accuracy: ~84%  
• Loss curves and accuracy curves indicate consistent convergence.  
• Test predictions:  
 - dogimg.jpeg → Classified as Dog  
 - catimg.jpeg → Classified as Cat



# Conclusion

The CNN model demonstrated reliable performance in binary image classification. Future improvements may include data augmentation, transfer learning, or increasing training duration for even better generalization.

# Future Scope

This project used a CNN model to find out if an image is of a cat or a dog. The model worked well, but we can make it better in the future by doing the following things:

1. **Change Images While Training (Data Augmentation):**
2. We can slightly change the training pictures (like turning them, zooming in, or flipping them). This helps the model learn better and not just remember the same images.
3. **Use Already-Trained Models (Transfer Learning):**

We can use models that are already trained on many images, like **VGG16** or **ResNet**. This will save time and make our model more accurate.

1. **Try Different Settings:**

We can test different values for settings like how fast the model learns, how many pictures we use at once, and how many times we train. This can give better results.

1. **Train for More Time or Use More Images:**

If we train the model longer or use more images, it might work even better.

1. **Make a Website or App:**

In the future, we can build a simple app or website where people can upload a picture, and the model will say if it's a cat or a dog.

1. **Add More Animals:**

Right now, the model only checks for cats and dogs. Later, we can add more animals or even different types of cats and dogs.

# References

- Kaggle Dataset: https://www.kaggle.com/datasets/salader/dogs-vs-cats  
- TensorFlow: https://www.tensorflow.org/api\_docs  
- Keras API: https://keras.io/  
- Google Colab: https://colab.research.google.com/