National College of Ireland

Configuration Manual

MSc Research Project
Data Analytics

Sahana Hombal
Student ID: 23297655

School of Computing
National College of Ireland

Supervisor: Teerath Kumar Menghwar

National College of Ireland

MSc Project Submission Sheet

School of Computing

Student Name: ……. Sahana Hombal……

Student ID: ……x23297655……..……

Programme: ……Data Analytics…… **Year:** ……2024……..

Module: …… MSc Research Project
……..……

Lecturer: …… Teerath Kumar Menghwar ……..……

Submission Due Date: ……29/01/2025……..……

Project Title: Enhancing Deepfake Detection with Multi-Modal Transformers

Word Count: …… **Page Count:** ……..……..……

I hereby certify that the information contained in this (my submission) is information pertaining to research I conducted for this project. All information other than my own contribution will be fully referenced and listed in the relevant bibliography section at the rear of the project.
ALL internet material must be referenced in the bibliography section. Students are required to use the Referencing Standard specified in the report template. To use other author's written or electronic work is illegal (plagiarism) and may result in disciplinary action.

Signature: ……Sahana Hombal……..……

Date: ……29/01/2025……

PLEASE READ THE FOLLOWING INSTRUCTIONS AND CHECKLIST

Attach a completed copy of this sheet to each project (including multiple copies)	□
Attach a Moodle submission receipt of the online project submission, to each project (including multiple copies).	□
You must ensure that you retain a HARD COPY of the project, both for your own reference and in case a project is lost or mislaid. It is not sufficient to keep a copy on computer.	□

Assignments that are submitted to the Programme Coordinator Office must be placed into the assignment box located outside the office.

Office Use Only	
Signature:	
Date:	
Penalty Applied (if applicable):	

Configuration Manual

Sahana Hombal
x23207655

1 Introduction

Deepfakes, created using **Generative Adversarial Networks (GANs)**, have become a significant challenge in ensuring media authenticity. These manipulated images, videos, and audio are increasingly realistic, posing risks to **cybersecurity**, **misinformation campaigns**, and **digital fraud**. Detecting such falsified media is crucial to maintaining trust and integrity in information systems.

While advancements in machine learning have improved deepfake detection, existing methods often fail when tested against **high-quality manipulations** or real-world distortions. The primary challenge lies in the adaptability and robustness of these detection systems in varying conditions. This research focuses on enhancing detection accuracy by leveraging **state-of-the-art machine learning models** tailored for both image- and audio-based detection tasks.

2 Hardware and Software Requirements

2.1 Hardware Requirements

The hardware configuration of the system for this research project and executed are as follow:

- Processor: Intel Core i5 or higher.
- RAM: 16 GB or more.
- GPU: NVIDIA Tesla or similar (for model training).
- Storage: At least 100 GB of free space.

2.2 Software Requirements

- Operating System: Windows 10 / Linux / macOS.
- Programming Language: Python 3.7 or above.
- IDE: Google Colab or Jupyter Notebook.
- Libraries:
  TensorFlow, Keras, OpenCV, Librosa, NumPy, Matplotlib, and Seaborn.

3 Environment Setup

3.1 Google Colab Setup

- Open Google Colab.
- Enable GPU: **Runtime > Change runtime type > Hardware Accelerator > GPU**.

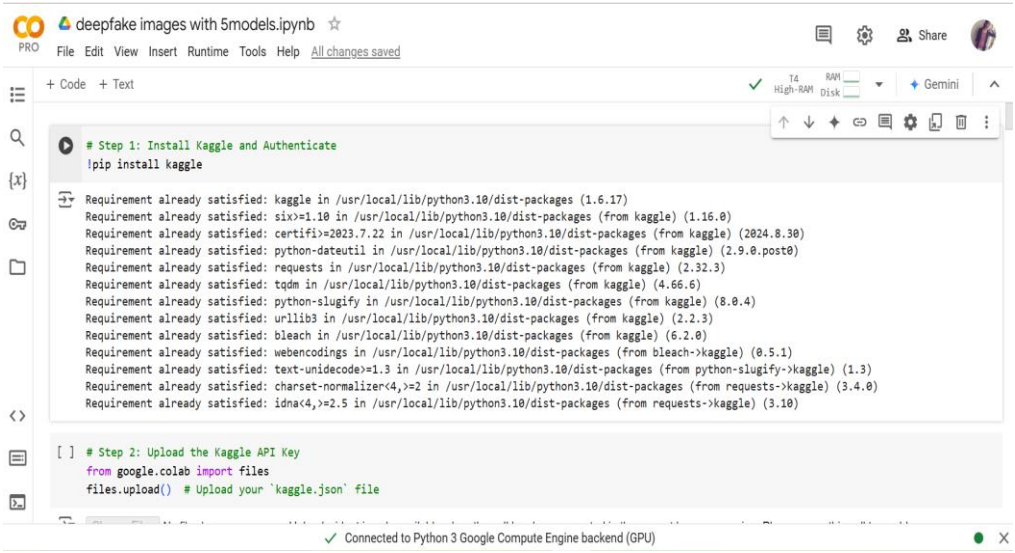

Figure 1: Screenshot of Google Colab with GPU enabled

3.2 Installing Libraries

Run the following command to install required libraries:

```
import tensorflow as tf
from tensorflow.keras.applications import ResNet50
from tensorflow.keras.models import Model
from tensorflow.keras.layers import GlobalAveragePooling2D, Dense, Dropout
from tensorflow.keras.preprocessing.image import ImageDataGenerator
from tensorflow.keras.optimizers import Adam
from tensorflow.keras.callbacks import EarlyStopping, ModelCheckpoint
from sklearn.metrics import classification_report, confusion_matrix
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
```

Figure 2: Importing libraries and packages

4. Data Preparation

4.1 Accessing Datasets

- Download datasets from Kaggle and upload them to Google Drive.
- Path setup in Colab:
  This section provides instructions on accessing Kaggle datasets using the Kaggle API in Google Colab.

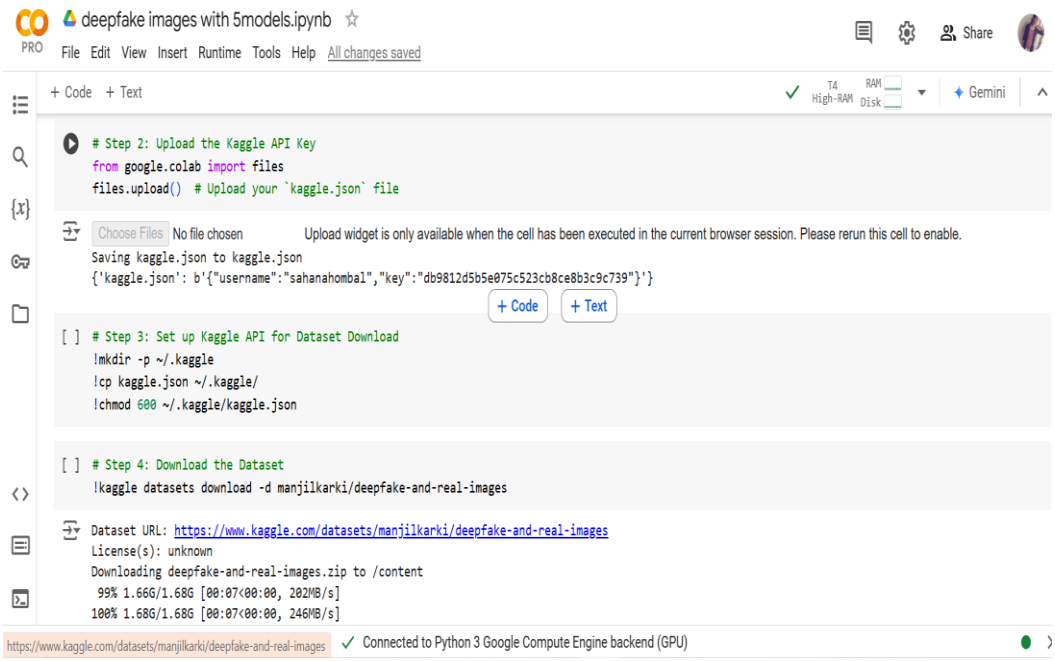

Figure 3: the dataset being downloaded using the API

4.2 Preprocessing

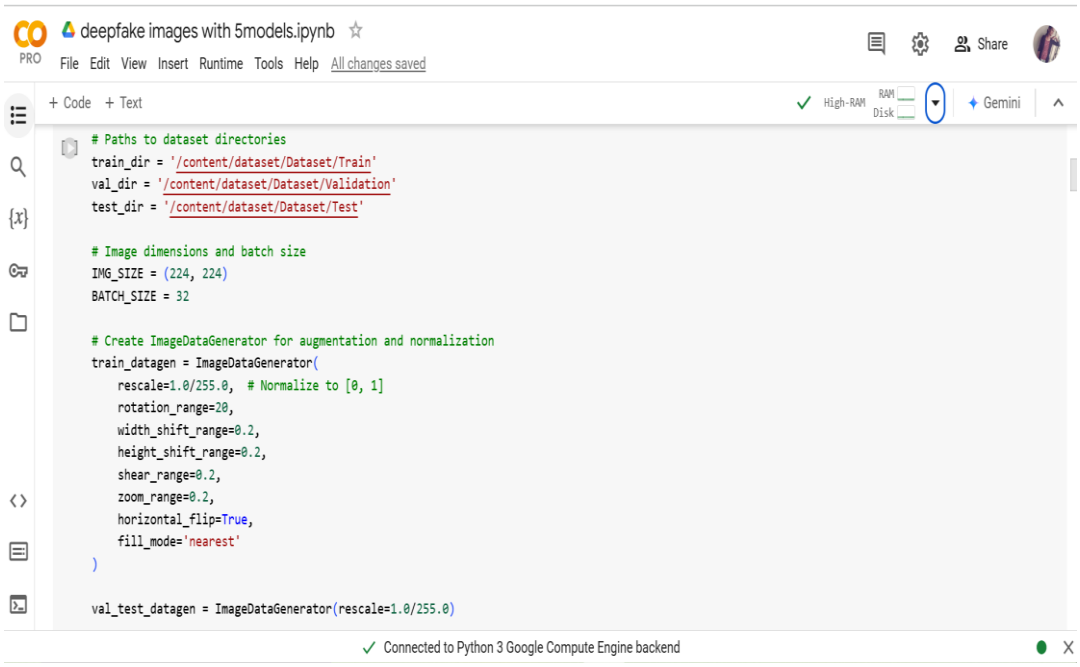

Figure 4: The preprocessing pipeline

5. Model Implementation

- Import the ResNet50 model from tensorflow.keras.applications with the pre-trained weights from ImageNet.
- Set the input shape to (224, 224, 3) to match image dimensions and exclude the top layers for fine-tuning.

- Freeze the first layers of the ResNet50 model (except the last 10 layers) to retain pre-trained weights and speed up training.

Figure 5: ResNet50 Model Building

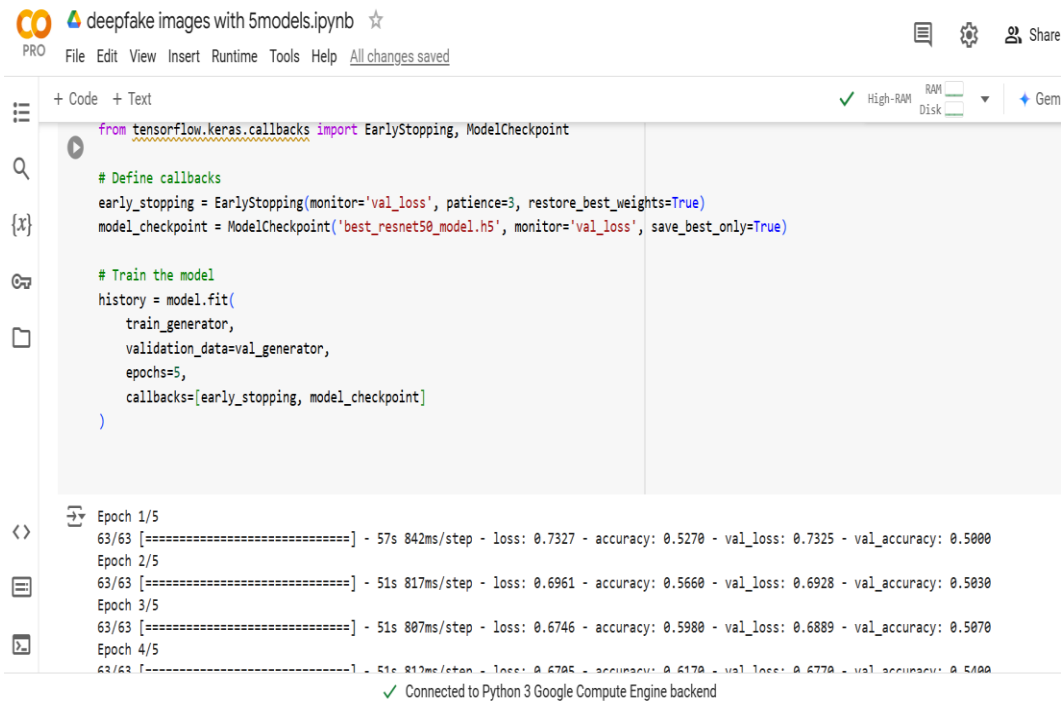

Fig 6: Training model ResNet50

6. Model Evaluation

6.1 Evaluate on Test Data:

- Use the evaluate() function to assess the model's performance on the test dataset. The function computes the loss and accuracy metrics for the model based on the data from the test generator.
- After evaluation, print the test accuracy to see how well the model is performing on unseen data

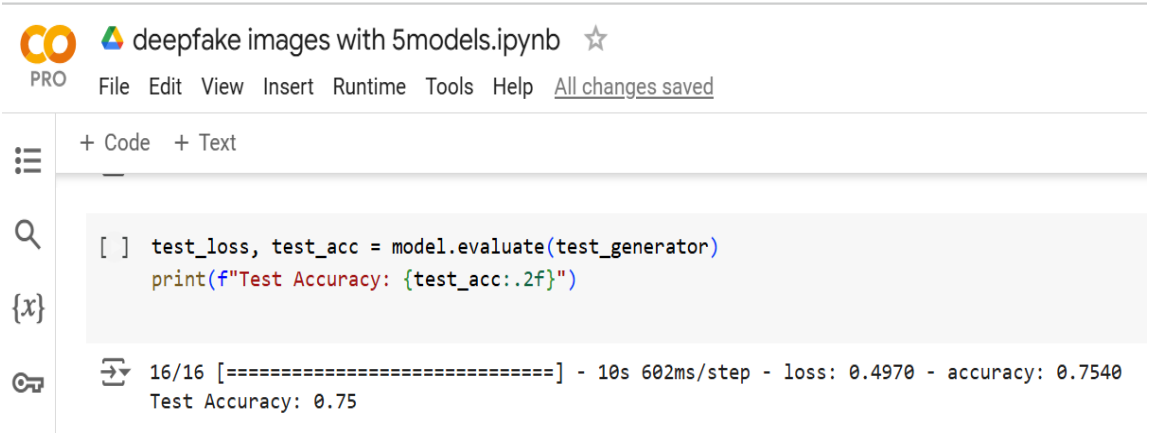

Figure 7: Evaluating model

7. References

Chen, H. (. X. W. Y. &., n.d. Assessing Performance in Natural Language Generation.. *Neural Networks and Learning Systems.*.

Chen, L. W. Y. &. Z. X. (., 2021. Hyperparameter Optimization in NLP Models: A Survey..

Guo, Y. G. Y. &. Z. X. (., 2018. Dimensionality Reduction Techniques for Text Mining. *Neural Networks and Learning Systems.* .

Xu, L. &. L. Y. Z. Y. (., n.d. Evaluation Metrics for Question Generation: A Comparative Study.. *Knowledge and Data Engineering.*.

Yu, J. L. Y. &. Z. X. (., 2019. Attention-Based Models for Question Generation.. *Pattern Analysis and Machine Intelligence.*.

Zhang, X. W. J. &. L. Y. (., 2020. Efficient Hyperparameter Optimization for Deep Learning Models.. *Neural Networks and Learning Systems.*.