Midterm Project Report

Prithvi Bhardwaj(24b0406)

Contents

1 Introduction

The objective of this midterm report is to document all work completed in the SOC 2025 project to date, including foundational Python skills, key modules and frameworks, and the design and preliminary implementation of a Siamese Neural Network. Additionally, this report integrates insights from the accelerated “10-Day Intensive” Code Challenge by CodeWithHarry to demonstrate rapid upskilling.

2 Python Fundamentals Review

2.1 Data Types and Structures

```
# Numeric types
a = 10
b = 3.1415
# Sequence types
s = "Hello,␣SOC"
lst = [1,2,3,4]
tup = (5,6,7)
# Mapping type
d = {"name": "Alice", "age": 24}
# Logical type
flag = True
```

Insight: Mastery of built-in types lays the groundwork for efficient algorithms and data handling.

2.2 Control Flow and Functions

```
# Conditional logic
if a % 2 == 0:
    print("Even")
else:
    print("Odd")

# Loop constructs
for i in lst:
    print(i)

# List comprehensions
squares = [x**2 for x in range(10) if x % 2 == 0]

# Functions with variable arguments
def greet(*names):
    for name in names:
        print(f"Hello,␣{name}!")
```

Insight: List comprehensions and argument unpacking increase code readability and conciseness.

2.3 Modules, Packages, and Virtual Environments

- Created and activated virtual environment via Anaconda Navigator.
- Installed packages: `numpy`, `pandas`, `matplotlib`, `tensorflow`, `opencv-python`.
- Learned differences between `pip` and `conda` installations.

3 10-Day Intensive Code Challenge by CodeWithHarry

3.1 Overview

- Duration: Complete the full playlist in 10 days.
- Platform: YouTube playlist by CodeWithHarry.
- URL: `https://youtube.com/playlist?list=PLu0W_9lII9agwh1XjRt242xIpHhPT2llg`
- I have completed the first 76 lectures which gave me a strong command on phython syntax and structure.

3.2 Sample Code Examples

Exception Handling and JSON Parsing

```
import json
try:
    with open(’config.json’,’r’) as f:
        config = json.load(f)
except FileNotFoundError:
    print("Configuration␣file␣not␣found.")
    config = {}
```

Working with CSV Files

```
import csv
with open(’data.csv’,’w’,newline=’’) as f:
    writer = csv.DictWriter(f, fieldnames=[’name’,’score’])
    writer.writeheader()
    writer.writerow({’name’:’Alice’,’score’:95})
```

Regular Expressions for Validation

```
import re
def validate_email(email):
    pattern = r"^[\w\.-]+@[\w\.-]+\.\w{2,4}$"
    return re.match(pattern,email) is not None

print(validate_email(’test@example.com’))  # True
```

Web Scraping with BeautifulSoup

```
import requests
from bs4 import BeautifulSoup

resp = requests.get(’https://example.com’)
soup = BeautifulSoup(resp.text,’html.parser’)
headings = [h.text for h in soup.find_all(’h2’)]
print(headings)
```

CLI Mini-Project Skeleton

```
import argparse

def main():
    parser = argparse.ArgumentParser(description=’Simple␣CLI’)
    parser.add_argument(’--name’, type=str, help=’Your␣name’)
    args = parser.parse_args()
    print(f"Hello,␣{args.name}!")

if __name__ == ’__main__’:
    main()
```

4 Core Python Modules for Data Science

4.1 NumPy (2 days)

```
import numpy as np
A = np.arange(16).reshape((4,4))
b = np.array([1,2,3,4])
C = A + b  # Broadcasting example
```

Deep Dive: Universal functions, memory views, advanced indexing. graphicx

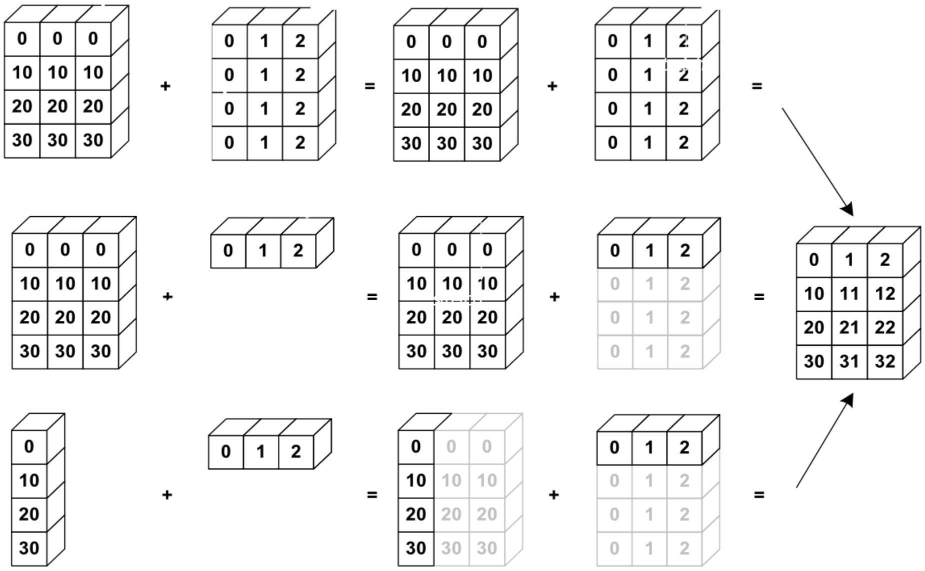

Figure 1: Numpy Broadcasting

4.2 Pandas (3 days)

```
import pandas as pd
df = pd.read_csv(’data.csv’)
df.fillna(method=’ffill’,inplace=True)
stats = df.groupby(’category’).agg({’score’:[’mean’,’std’]})
```

Deep Dive: Multi-index DataFrames, pivot tables.

4.3 Matplotlib (1 day)

```
import matplotlib.pyplot as plt

fig, ax = plt.subplots(1,2,figsize=(10,4))
ax[0].scatter(df[’age’],df[’score’])
ax[0].set_title(’Age␣vs␣Score’)
ax[1].hist(df[’score’],bins=10)
ax[1].set_title(’Score␣Distribution’)
plt.tight_layout()
plt.show()
```

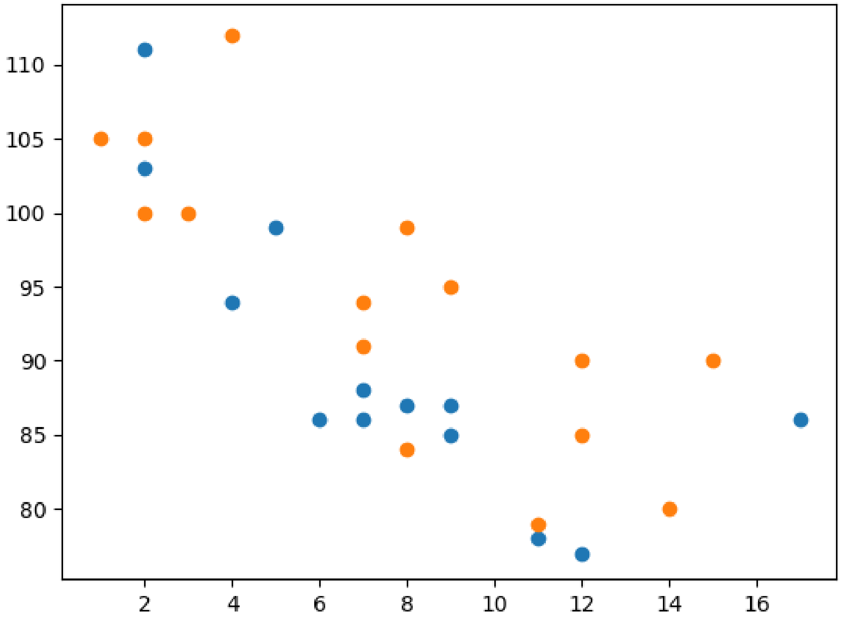

Figure 2: Matplotlib Plot

5 Deep Learning Frameworks

5.1 TensorFlow & Keras

```
import tensorflow as tf
from tensorflow.keras import layers, models

model = models.Sequential([
    layers.Flatten(input_shape=(28,28)),
    layers.Dense(256, activation=’relu’),
    layers.Dropout(0.3),
    layers.Dense(10, activation=’softmax’)
])
model.compile(
    optimizer=’adam’,
    loss=’sparse_categorical_crossentropy’,
    metrics=[’accuracy’]
)
history = model.fit(x_train,y_train,epochs=10,validation_split=0.1)
```

5.2 OpenCV

```
import cv2
face_cascade = cv2.CascadeClassifier(’
   haarcascade_frontalface_default.xml’)
gray = cv2.cvtColor(img,cv2.COLOR_BGR2GRAY)
faces = face_cascade.detectMultiScale(gray,1.1,4)
for (x,y,w,h) in faces:
    cv2.rectangle(img,(x,y),(x+w,y+h),(255,0,0),2)
cv2.imshow(’Faces’,img)
cv2.waitKey()
```

6 Siamese Neural Network for One-Shot Recognition

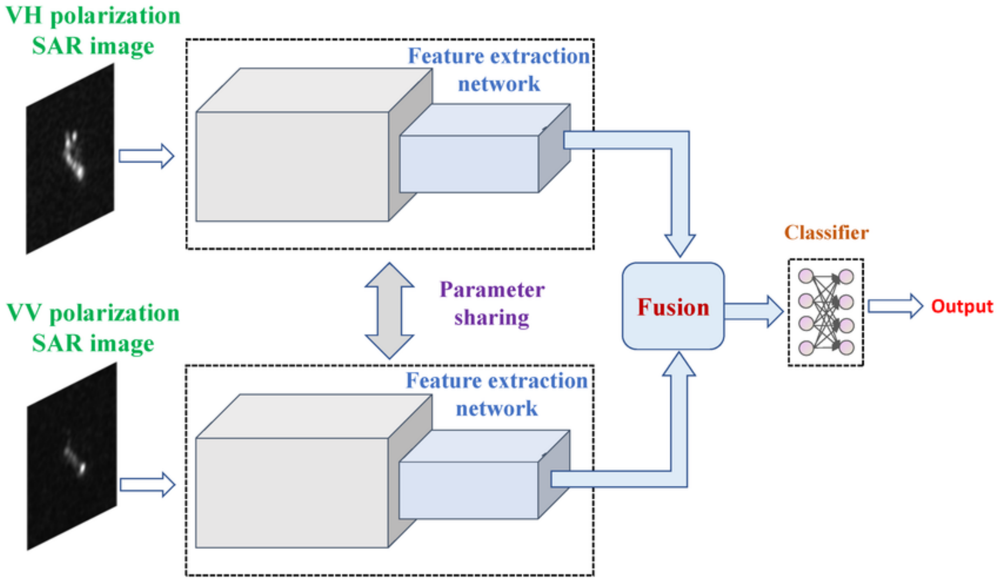

Figure 3: Siamese Network Schematic

6.1 Architecture and Loss Function

Two identical CNN branches (shared weights) generate 128-dim embeddings. Contrastive loss:

$$L = y \cdot D^2 \; + \; (1 - y) \cdot \max(0, m - D)^2$$

where D is Euclidean distance and m is the margin.

6.2 Dataset Preparation

Collected 1,000 image pairs (positive/negative) from the Omniglot subset. Preprocessing: resize to 105×105, normalize pixel values to $[0, 1]$.

6.3 Training Outline

```
siamese_model.compile(
    optimizer='adam',
    loss=’binary_crossentropy’,
    metrics=[’accuracy’]
)
siamese_model.fit(
    [pairs_a,pairs_b],
    labels,
    batch_size=32,
    epochs=20,
    validation_split=0.2
)
```

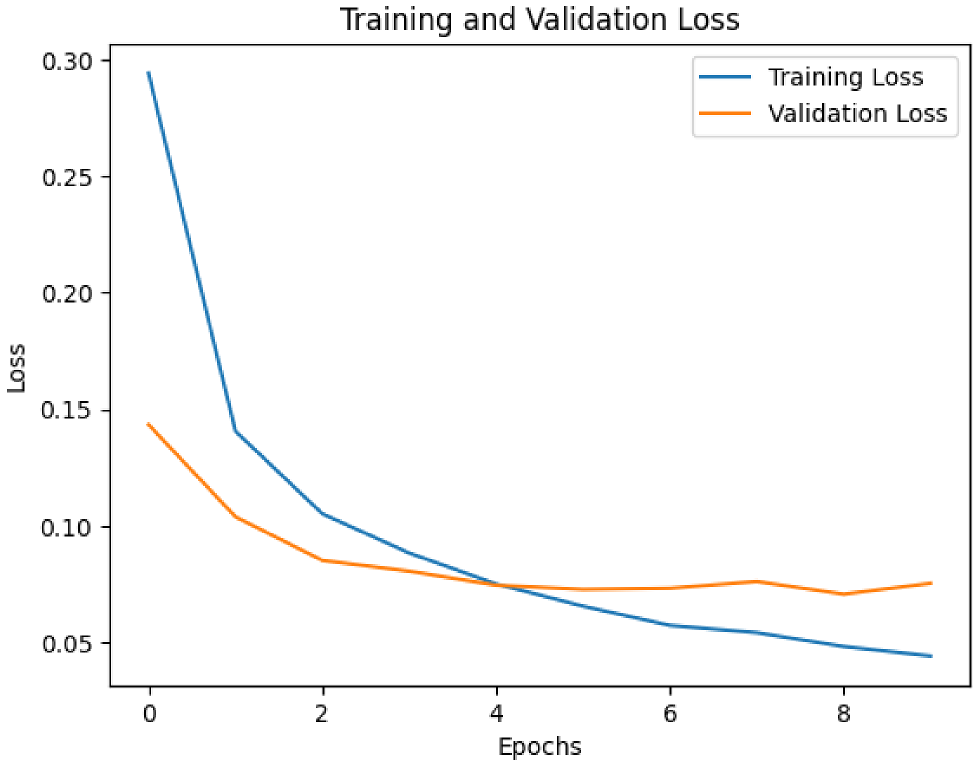

Figure 4: Training vs Validation Metrics Chart

7 Challenges & Reflections

- Balancing rapid 10-day challenge with in-depth exploration.
- Debugging cross-version conflicts between TensorFlow and OpenCV.
- Establishing Git workflows and collaborative practices.

8 Next Steps

1. Finalize Siamese model training and evaluation.
2. Optimize with advanced loss functions (e.g., triplet loss).
3. Integrate the trained model into the application pipeline.
4. Prepare the final report and presentation.

9 References

1. NumPy Documentation: `https://numpy.org/doc/stable/user/absolute_beginners.html`
2. Pandas Tutorial: `https://www.w3schools.com/python/pandas/default.asp`
3. CodeWithHarry 10-Day Intensive Playlist: `https://youtube.com/playlist?list=PLu0W_9lII9agwh1XjRt242xIpHhPT2llg`
4. Siamese Neural Networks for One-shot Recognition (pdf)
5. TensorFlow Keras Guide: `https://www.tensorflow.org/guide/keras/api`