

Contents

 Project Overview	2
 Development Process	2
1. Model Training in Streamlit	2
2. User Upload and Prediction	2
 Test Results	3
 Challenges & Solutions	3
 Ubuntu, Ethics, and Bias Reflection	4
 Ubuntu and Humanity in AI	4
 Bias and Fairness	4
 Data Responsibility	4
 Final Output & Deployment	4
 Conclusion	5
 Future improvements:	5
Annexure: Test Results	6

Project Report: MNIST Handwritten Digit Classifier (Trained in Streamlit)

⌚ Project Overview

This project involved building and deploying a **web-based handwritten digit classifier** using the **MNIST dataset**, trained in real-time with **TensorFlow** and served with **Streamlit**. The user uploads a 28x28 white-on-black PNG image of a digit (0–9), and the trained model predicts the digit and shows the confidence level.

🏗 Development Process

1. Model Training in Streamlit

- Used the built-in MNIST dataset via `tf.keras.datasets.mnist.load_data()`.
- Normalized the pixel values and reshaped the dataset to (28, 28, 1).
- Built a CNN with:
 - 2 convolutional layers
 - MaxPooling layers
 - Dense and softmax output
- Trained the model for **3 epochs**.
- Training happens **once per session** using `st.session_state` to save time.

2. User Upload and Prediction

- The app accepts only **28x28 grayscale white-on-black PNG images**.
 - The image is:
 - Converted to grayscale (`convert("L")`)
 - Inverted (to match MNIST's black-on-white structure)
 - Normalized to [0, 1] range
 - Reshaped to match model input (1, 28, 28, 1)
 - Model then predicts the digit class and displays confidence score.
-

Test Results

You tested the following digits:

Image	True Label	Predicted	Confidence
HAND 1.png	1	<input checked="" type="checkbox"/> 1	1.00
HAND 2.png	2	<input checked="" type="checkbox"/> 2	0.64
2.png	2	<input checked="" type="checkbox"/> 7	0.71
HAND 4.png	4	<input checked="" type="checkbox"/> 4	0.89

 Observation: The model performs well on clearly written digits, but struggles with ambiguous shapes or poor-quality samples (e.g., "2" misclassified as "7").

Challenges & Solutions

Challenge	Solution
Model stuck on previous prediction	Used <code>st.session_state.model</code> to persist the trained model and prevent reloading or rerunning training unnecessarily.
Slow prediction on upload	Reduced training epochs to 3 for faster in-app training and removed external .h5 model dependencies.
Images not recognized	Enforced image preprocessing: 28x28 format, grayscale conversion, inversion (<code>255 - np.array(img)</code>), and normalization to match MNIST input.
Streamlit Cloud Deployment Failed	TensorFlow 2.10.0 does not support Python 3.13 , which is used in the online environment. 
	<input checked="" type="checkbox"/> Solution: Downgraded to Python 3.10 locally to match TensorFlow compatibility and run Streamlit without errors.
h5 Model Compatibility Issues	.h5 models trained elsewhere or using newer Keras versions had unsupported config (e.g., <code>batch_shape</code>). <input checked="" type="checkbox"/> Solution: Stopped using .h5 files and trained the model directly inside the Streamlit app , ensuring full compatibility.

Missing packages (e.g., scipy, matplotlib)	Installed only essential dependencies and trimmed unused libraries to improve performance and prevent import errors.
Misclassified digits (e.g., 2 predicted as 7)	Recommended clearer handwriting and white-on-black image format. Could be improved with more diverse training samples in future.



Ubuntu, Ethics, and Bias Reflection



Ubuntu and Humanity in AI

In the spirit of **Ubuntu** ("I am because we are"), this tool empowers anyone—especially students and communities—to test and build intelligent models without expensive infrastructure. Sharing tools openly aligns with collaborative, inclusive AI.



Bias and Fairness

- The model is trained on MNIST—a balanced dataset, but only reflects U.S. handwriting patterns.
- Bias may appear when testing on non-standard or non-English digit shapes.
- Ethics tools like **TensorFlow Fairness Indicators** could be integrated in the future to test and mitigate prediction bias.



Data Responsibility

- We only used open datasets and did not collect or store user data.
 - All model decisions are transparent (shown with confidence levels).
-



Final Output & Deployment

- Model successfully predicts digit from uploaded PNGs.
- Deployed using Streamlit on localhost (localhost:8505).
- Dependencies are listed in requirements.txt:

txt

CopyEdit

tensorflow==2.10.0

streamlit==1.22.0

numpy==1.23.5

pillow

Spicy

Conclusion

This project proved that it's **possible to build and deploy a complete deep learning model inside a Streamlit web app** without external saving or uploading. Despite minor misclassifications, the model works with a high degree of accuracy on well-formatted images.

Future improvements:

- Add a drawing canvas for real-time input
- Include a confidence bar chart
- Train with more handwritten variation to improve generalization

Annexure: Test Results

localhost:8505

MNIST Handwritten Digit Classifier (Trained in Streamlit)

Upload a 28x28 grayscale digit image (white on black) to classify it.

Upload a 28x28 white-on-black PNG

Drag and drop file here
Limit 200MB per file • PNG

Browse files

HAND 2.png 0.8KB

x

Uploaded Digit

Predicted Digit: 2 (Confidence: 0.64)

localhost:8505

MNIST Handwritten Digit Classifier (Trained in Streamlit)

Upload a 28x28 grayscale digit image (white on black) to classify it.

Upload a 28x28 white-on-black PNG

Drag and drop file here
Limit 200MB per file • PNG

Browse files

2.png 120.0B

x

Uploaded Digit

Predicted Digit: 7 (Confidence: 0.71)

← ⌛ ⓘ localhost:8505 ⚙️ 🌐 ⋮

MNIST Handwritten Digit Classifier (Trained in Streamlit)

Upload a 28x28 grayscale digit image (white on black) to classify it.

Upload a 28x28 white-on-black PNG

Drag and drop file here
Limit 200MB per file • PNG

HAND 4.png 0.8KB

Uploaded Digit

Predicted Digit: 4 (Confidence: 0.89)

MNIST Handwritten Digit Classifier (Trained in Streamlit)

Upload a 28x28 grayscale digit image (white on black) to classify it.

Upload a 28x28 white-on-black PNG

Drag and drop file here
Limit 200MB per file • PNG

HAND 1.png 0.6KB

Uploaded Digit

Predicted Digit: 1 (Confidence: 1.00)

← ⌛ ⓘ localhost:8505 ⚙️ 🌐 ⋮

MNIST Handwritten Digit Classifier (Trained in Streamlit)

Upload a 28x28 grayscale digit image (white on black) to classify it.

Upload a 28x28 white-on-black PNG

Drag and drop file here
Limit 200MB per file • PNG

Made with Streamlit