Part 3: Ethics & Optimization Report

1. Ethical Considerations

A. MNIST Model Biases

Although the MNIST has become a standard dataset and it is deemed to be non-biased, their training models my incur a number of possible biases:

- Homogeneity of Data: The dataset is on the ones written by the census workers and students, which would not represent all the forms of handwritings worldwide.
- Visual Contrast Assumptions: The model presupposes the black-on-white digits. Inverted-color images (black backgrounds, with white digits) produce poor performance unless specially addressed.
- Age of Dataset: MNIST has been over 20 years old. It would not take into consideration present day variations of inputs like stylus drawn or touchscreen input.

Mitigation Strategies

- Data Augmentation: add various and other types of styles to the digits, such as: inverted colours, thick and thin, and distortions.
- Input Normalization: bypassed normalization and automatically invert the pixel color to black-ground and white-digit.

Tools of Model Auditing:

  - TensorFlow Fairness Indicators: This is typically more applicable to demographic fairness, but it would be possible to reuse it to audit performance across digit varieties or in writer sets.
  - Custom Evaluation Metrics: Form test splits based on stylistic differences and compare model accuracy on each of the splits.

B. Amazon Reviews NLP Biases

The sentiment analysis and NER pipeline of Amazon Reviews is open to some ethical issues:

- Training Label Bias: The star-based sentiment labels do not necessarily demonstrate the proper tone or emotion.
- Polarity Bias: A word (e.g., cheap) may have different meanings in various contexts but rule-based tools such as TextBlob treat this word in the same way.
- Entity Recognition Gaps: spaCy may fail to recognize domain specific entities like not so common names of or names of brands or products which it prefers to recognize common or famous entities.

Mitigation Strategies

- Fine-Tuning: Enhance the model of spaCy by fine-tuning it with a domain-specific set of data.
- Rule-Based Layer Upgrade: Up-level the TextBlob or the spaCy sentiment with lists or patterns of keywords discussing a specialised language.
- Cross-Validation: Testers should check how accurately sentiment varies across product categories, or price to identify performance bias.
- Transparency: Reveal confidence of prediction and give explanations to the final consumer, particularly in reviews with mixed rating.

2. Troubleshooting Challenge – Buggy TensorFlow Code Fixes

Problem 1: Dimension Mismatch

Original Error:

```
model.add(Dense(10))

model.compile(loss='categorical_crossentropy', ...)
```

But labels were integers (0–9), not one-hot encoded.

Fix:
Change the loss function to:

```
loss='sparse_categorical_crossentropy'
```

This allows direct use of integer class labels.

Problem 2: Incorrect Input Shape

Original Error:

```
model.add(Input(shape=(28, 28)))
```

This caused shape errors in Conv2D, which expects 4D tensors (batch, height, width, channels).

Fix:
Explicitly reshape inputs:

```
x_train = x_train.reshape(-1, 28, 28, 1)

x_test = x_test.reshape(-1, 28, 28, 1)
```

And use:

```
model.add(Input(shape=(28, 28, 1)))
```

Problem 3: Evaluation Logic

Issue:
Model wrongly predicted a digit due to inverted image colors (white digit on black).

Fix:
Add image inversion logic before prediction:

```
if np.mean(image) > 127:
    image = 255 – image
```

Conclusion

Even such seemingly neutral activities as digit recognition or review analysis require bias and fairness. Transparency, rule-based logic, fairness tools may help unearth and reduce such biases even before they affect the outcome of diverse test sets. Similarly, the process of troubleshooting and debugging typical machine learning challenges, including the shape of the data, flawed model evaluation, among others, allows both a technically and ethically conscious model performance**.**