Submission:
Daniel Adam - ID: 342475639

GMDL Project Summary

Contents

Method and Rationale

The Open Set Recognition (OSR) model designed in this project utilizes a combination of a baseline Convolutional Neural Network (CNN) Digit classifier and an Autoencoder (AE). The baseline model is responsible for classifying known digits from the MNIST dataset, while the Autoencoder is used to detect Out-of-Distribution (OOD) samples (i.e., samples that do not belong to the known classes).
The rationale behind this approach is to leverage the baseline model's classification capabilities while using the reconstruction error from the Autoencoder to identify samples that the model has not been trained on, thereby effectively handling open set recognition.
To distinguish between known and unknown samples, specific thresholds were introduced:

- **MNIST Data Threshold:** A threshold was set based on the softmax probabilities of the baseline model's output. Specifically, the threshold was determined by calculating the 1st percentile of the maximum softmax probabilities from the validation set. This means that only 1% of the validation samples had a maximum softmax probability lower than this threshold. Therefore, during evaluation, any sample from the test data with a maximum softmax probability below this threshold was classified as "unknown."

- **Reconstruction Error Threshold**: For the Out-of-Distribution (OOD) data, a threshold was applied to the reconstruction error produced by the Autoencoder. This threshold was determined by calculating the 99th percentile of the reconstruction errors on the MNIST validation set. This implies that 99% of the MNIST validation samples had a reconstruction error lower than this threshold. During evaluation, any sample with a reconstruction error exceeding this threshold was identified as "unknown," effectively classifying it as an OOD sample.

These thresholds were carefully selected to balance the trade-off between accurately classifying known samples and effectively identifying unknown samples, ensuring the OSR model's robustness and reliability.

Data Preparation

- Transformations:
  - The transformation pipeline includes resizing, normalizing, and converting images to tensors.
  - For MNIST and CIFAR-10 datasets, images are resized to a uniform size of 28x28 pixels.
  - This resizing ensures consistency across datasets, allowing them to be processed by the same model architecture.
  - The pixel values are normalized using the respective dataset's mean and standard deviation, which aids in faster convergence during training.

- Data Augmentation:
  - To further improve the model's robustness, data augmentation techniques such as random horizontal flipping and random rotation are applied.
  - Random Horizontal Flip: This augmentation randomly flips the images horizontally, introducing variability in orientation, helping the model learn to recognize objects irrespective of their orientation.
  - Random Rotation: Images are randomly rotated by up to 10 degrees, adding rotational variance, which improves the ability to handle different perspectives.

This approach ensures that the model is trained on a more diverse set of images, leading to better generalization when encountering unseen data, particularly useful in the context of OSR.

Hyperparameters

Key hyperparameters include the learning rate, number of epochs, and thresholds for MNIST classification and reconstruction errors. The final configuration of the thresholds was selected based on performance on the validation set.
Hence, the MNIST threshold was set at the 1st percentile of the softmax probabilities, and the reconstruction error threshold was set at the 99th percentile for OOD images.
These thresholds were chosen to minimize the false positive rate for known classes while effectively capturing OOD samples.
The best trade-off between a high accuracy on MNIST and also on OOD was achieved when setting the percentiles to 1% for MNIST and 99% for reconstruction within a set of trials of [[1%, 99%],[3%, 97%],[5%, 95%]].

Hence, the determined loss thresholds are:

- for MNIST data: 0.63862 (referencing to 5.1 in the notebook)
- for Reconstruction: 0.70454 (referencing to 5.2 in the notebook)

Results of Evaluation with Figures

- The training loss curves for both the baseline model and the Autoencoder (Figure 1 and Figure 2) indicate that the models converge within 10 training epochs used.

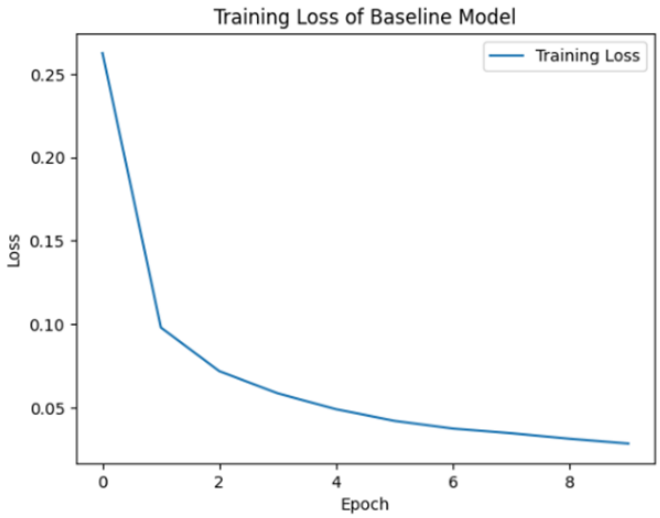

Figure 1
(4.1 in the notebook)

```
This is the trained Baseline Model:
CNNDigitClassifier(
  (conv1): Conv2d(1, 32, kernel_size=(3, 3), stride=(1, 1))
  (conv2): Conv2d(32, 64, kernel_size=(3, 3), stride=(1, 1))
  (fc1): Linear(in_features=9216, out_features=128, bias=True)
  (fc2): Linear(in_features=128, out_features=10, bias=True)
)
```

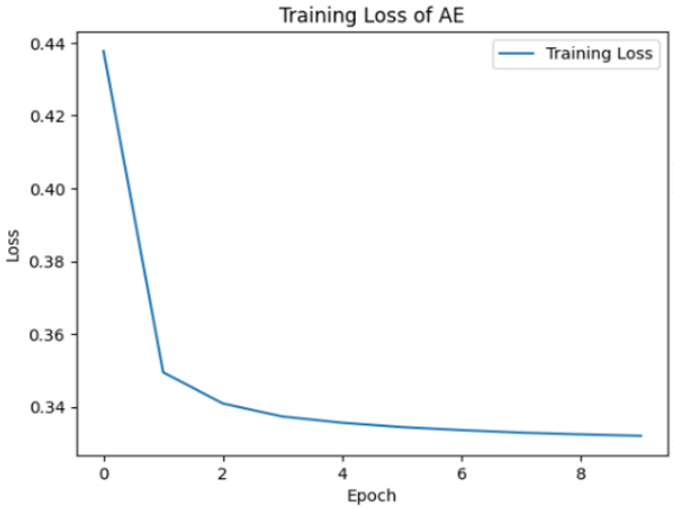

Figure 2
(4.2 in the notebook)

```
This is the trained Auto Encoder:
Autoencoder(
  (encoder): Sequential(
    (0): Conv2d(1, 16, kernel_size=(3, 3), stride=(2, 2), padding=(1, 1))
    (1): ReLU()
    (2): Conv2d(16, 32, kernel_size=(3, 3), stride=(2, 2), padding=(1, 1))
    (3): ReLU()
    (4): Conv2d(32, 64, kernel_size=(7, 7), stride=(1, 1))
  )
  (decoder): Sequential(
    (0): ConvTranspose2d(64, 32, kernel_size=(7, 7), stride=(1, 1))
    (1): ReLU()
    (2): ConvTranspose2d(32, 16, kernel_size=(3, 3), stride=(2, 2), padding=(1, 1), output_padding=(1, 1))
    (3): ReLU()
    (4): ConvTranspose2d(16, 1, kernel_size=(3, 3), stride=(2, 2), padding=(1, 1), output_padding=(1, 1))
    (5): Tanh()
  )
)
```

- The achieved accuracy of the baseline model on the MNIST test set is 98.46 %. (referencing to plot 7.1.1 in the notebook)
- Thus, also the confusion matrix for the baseline model on the MNIST test set (Figure 3) shows high accuracy with minimal misclassifications for each MNIST digit 0-9.

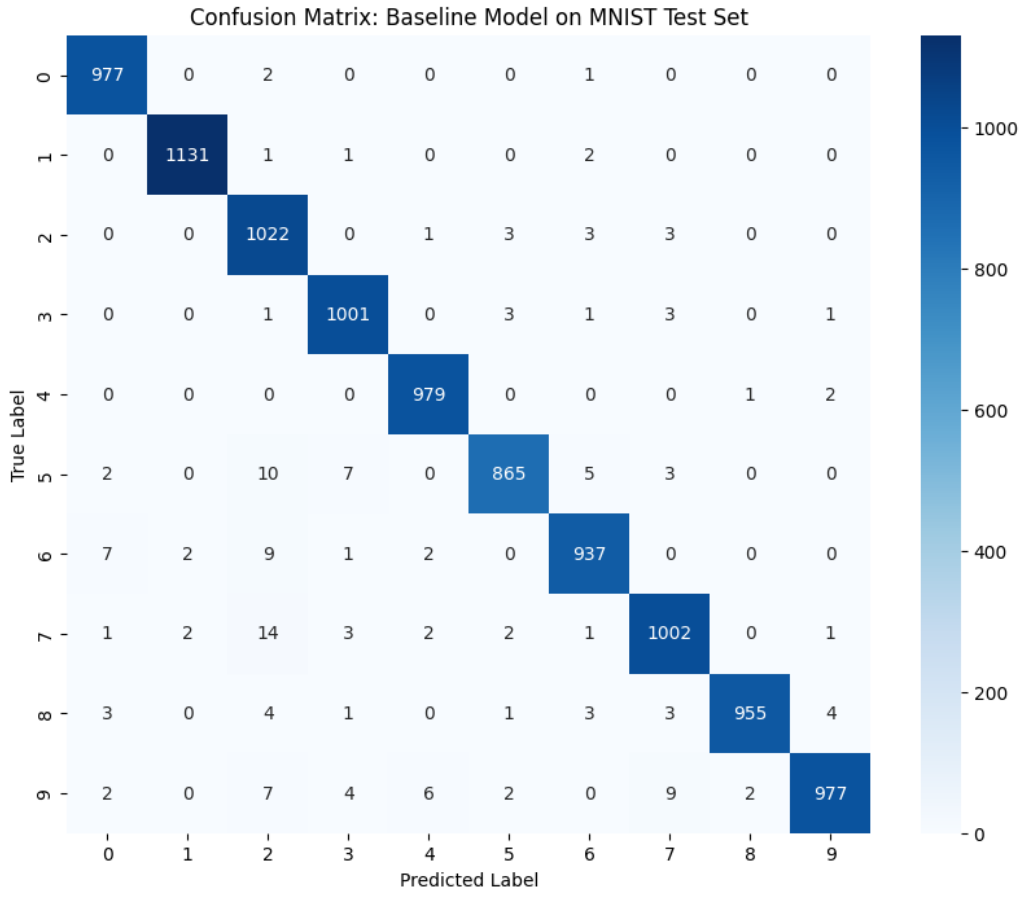

Figure 3
(7.1.2 in the notebook)

- The achieved binary accuracy of the OSR model on the combined data is 97.32 %. (referencing to plot 7.2.1 in the notebook)
- Hence, the binary confusion matrix for the OSR model on combined data (Figure 4) illustrates that the model effectively distinguishes between known and unknown classes.

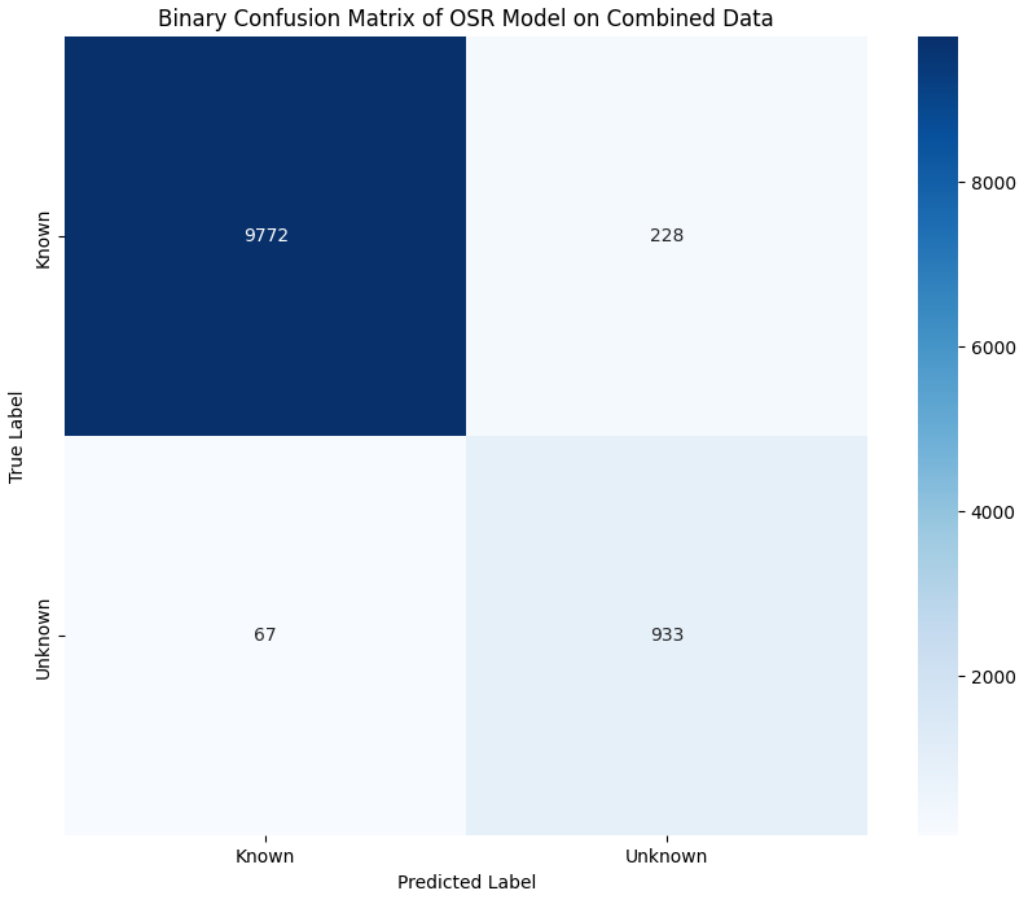

Figure 4
(7.2.2 in the notebook)

- The achieved accuracies of the OSR model on the combined data are:

  - MNIST Accuracy: 96.43%
  - OOD Accuracy: 93.30%
  - Total Accuracy: 96.15%

  (referencing to plot 7.3.1 in the notebook)

- This results in a confusion matrix (Figure 5) which shows that the OSR model can highly classify correctly each digit of the MNIST data and OOD-images.

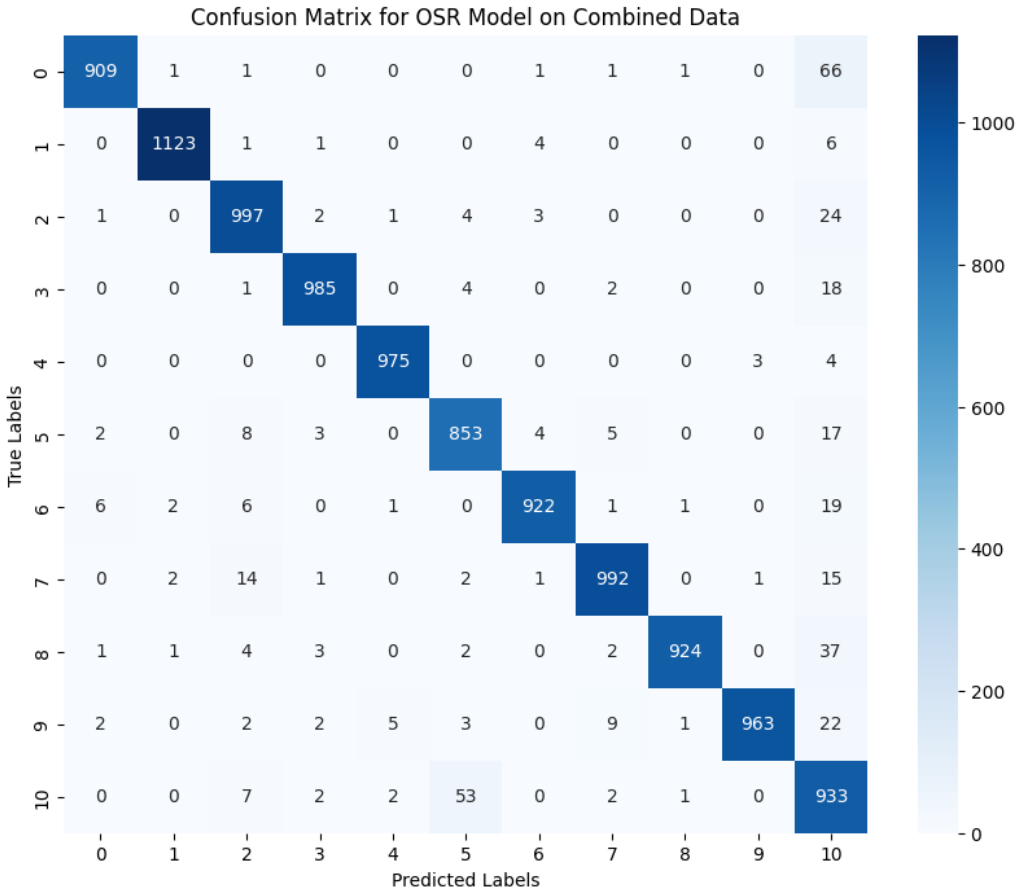

Figure 5
(7.3.2 in the notebook)

- The TSNE and PCA visualizations of embeddings (Figures 6 and 7) provide insights into the clustering of known and unknown classes. Visualizing that the OSR model highly classifies each digit of MNIST correctly and further classifies non-MNIST images correctly as unknown (unknown := **back colour**).

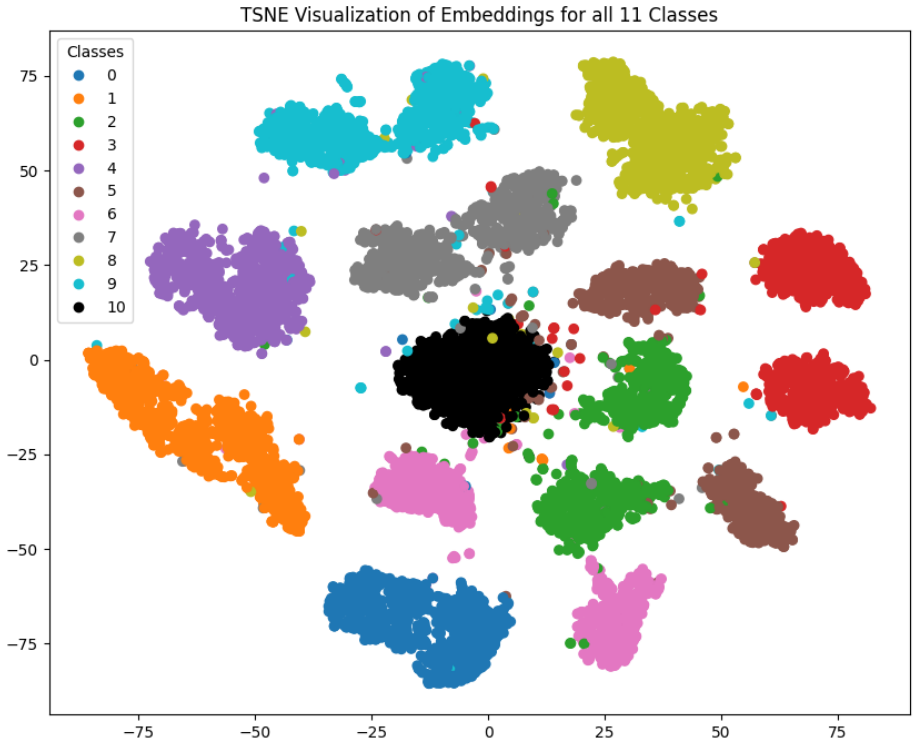

Figure 6
(7.4 in the notebook)

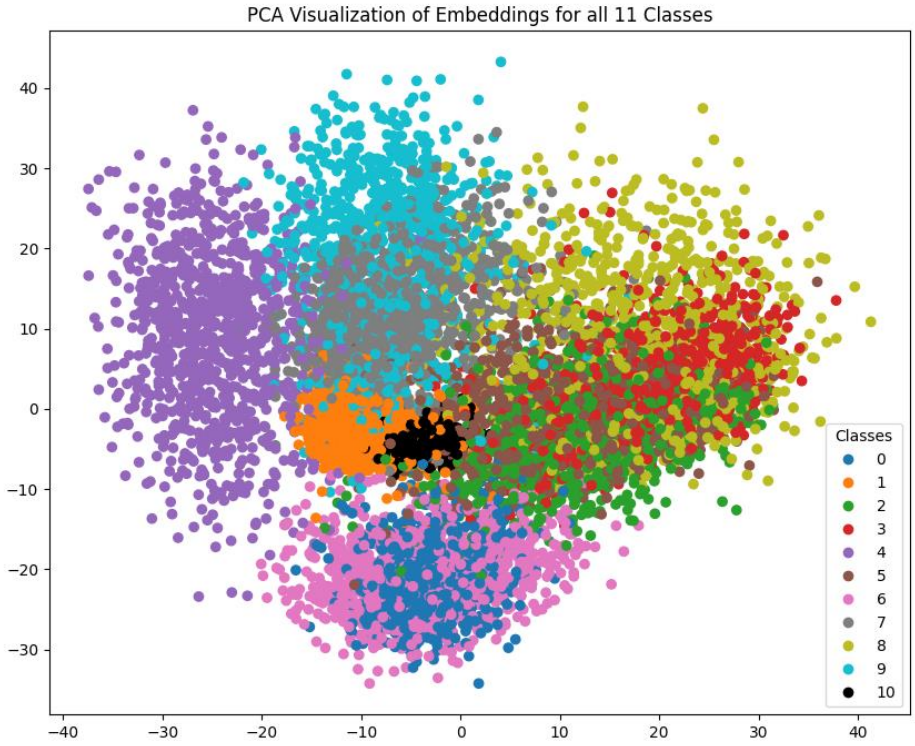

Figure 7
(7.5 in the notebook)

- The ROC curve (Figure 8) further demonstrates the model's effectiveness in distinguishing between known and unknown classes with an AUC of 0.97.

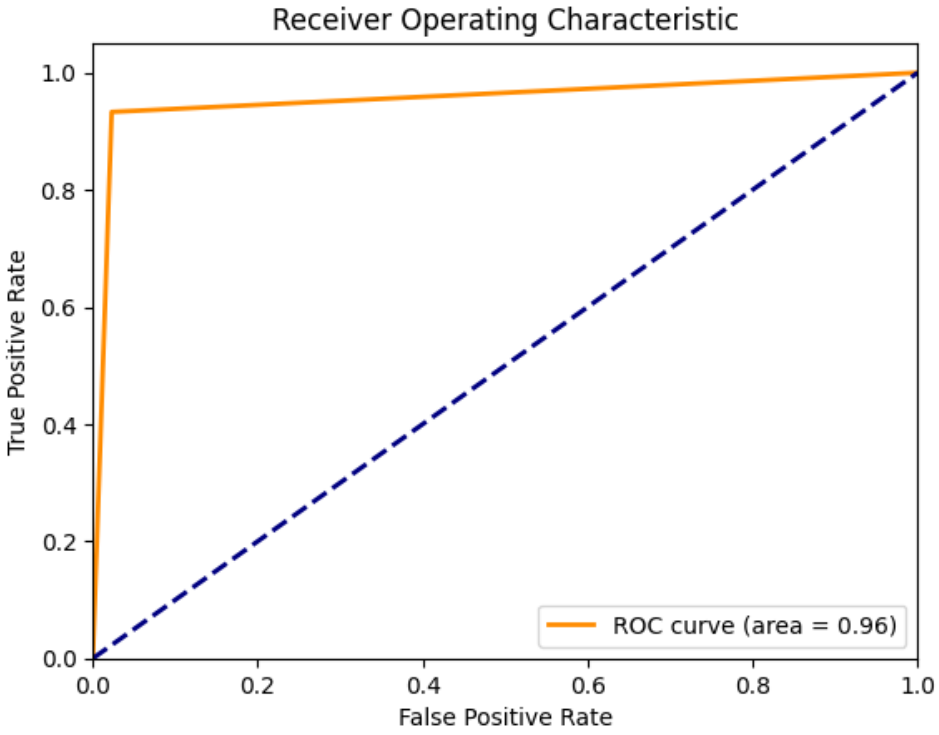

Figure 8
(7.6 in the notebook)

Limitations

The primary limitation of the approach lies in its dependency on the quality of the Autoencoder's reconstruction capabilities. If the Autoencoder fails to accurately reconstruct known samples, it may lead to false positives, labelling known classes as unknown. The model may also struggle with borderline samples where the reconstruction error is close to the threshold. The approach performs well on datasets with distinct class boundaries like MNIST and CIFAR-10 but may face challenges with more complex, overlapping classes in other datasets. The model may also struggle classifying correctly images which are more rotated and flipped images.

Conclusion

- The Open Set Recognition (OSR) model developed in this project effectively combines a baseline CNN classifier with an Autoencoder to address the challenge of recognizing both known and unknown samples.
- By leveraging data augmentation techniques and selection of thresholds on a validation set, the model achieved high accuracy in both closed and open-set scenarios.
- While the approach demonstrates strong performance on well-defined datasets like MNIST and CIFAR-10, its dependency on the quality of the Autoencoder highlights the need for further exploration when applied to more complex datasets.
- Future work could focus on refining the Autoencoder's reconstruction capabilities and exploring alternative architectures, like a multi-task autoencoder, to enhance the model's robustness in more diverse and challenging real-world use cases.