Project Summary and Model Rationale

1. Introduction

This project involves training a multi-task AutoEncoder with one head for classification and another for out-of-distribution (OOD) detection while training on the MNIST dataset, and testing our model's performance on MNIST, FashionMNIST, and CIFAR-10 datasets. The model is designed to classify MNIST digits while simultaneously performing image reconstruction. A key feature of this approach is integrating an OOD detection mechanism using reconstruction error scaling.

2. Method and Rationale

Our method leverages a multi-task AutoEncoder that jointly learns classification and reconstruction. This allows the model to:

- Improve robustness by learning meaningful latent representations.
- Detect OOD samples based on reconstruction error.
- Enhance classification accuracy using a shared encoder.

Previous Attempts and Learnings:

1. **Baseline CNN Classifier:** Performed well on MNIST but lacked OOD detection.
2. **CNN Classifier + OpenMax Method:** Tried to create something similar to OpenMax, which was mentioned in the article "Recent Advances in Open Set Recognition," but the implementation was complex.
3. **Simple AutoEncoder with output layer of 10 classes:** Provided high reconstruction and classification ability, but performed the classification of OOD samples only at the evaluation stage, which wasn't compatible with the eval_model method from project_utils.

Current Multi-Task AutoEncoder (with 11 classification logits): Balances both tasks efficiently.

Latest Attempt and Adjustments:

Initially, our model returned only 10 predictions and discovered unknown samples during evaluation based on a fixed confidence threshold. However, this approach did not align with the project's evaluation method, which assumed that the model should output 11 predictions, with the 11th class representing the unknown category. To address this, we modified our model to explicitly include the unknown category as an output class, ensuring compatibility with the evaluation process.

3. Data and Preprocessing

Datasets Used:

- MNIST: Primary classification task.
- FashionMNIST and CIFAR-10: Used for OOD detection.

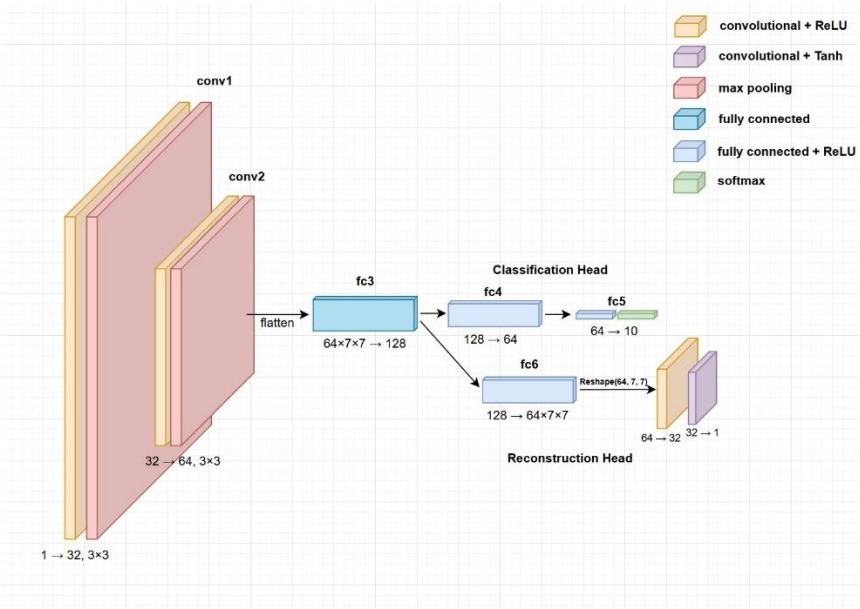
Preprocessing Steps:

- Resizing: All images are resized to 28x28.
- Normalization: Pixel values are normalized.
- Augmentations:
    - Rotation (15 degrees)
    - Affine transformation (10% translation)
- Grayscale Conversion: CIFAR-10 images are converted to grayscale.

Data Splits:

- MNIST Training Set: A subset of 10,000 samples (90% train, 10% validation)
- OOD Samples: 1,500 (50% FashionMNIST, 50% CIFAR-10)
- Test Set: MNIST and OOD samples combined.

4. Model Architecture



MultiTask AutoEncoder

- Encoder:
    - Conv2D(1 → 32), ReLU, MaxPool(2)
    - Conv2D(32 → 64), ReLU, MaxPool(2)
    - Fully Connected layer (Latent Space: 128)
- Decoder:
    - Fully Connected layer (64x7x7 output)

- ConvTranspose2D(64 → 32), ReLU
- ConvTranspose2D(32 → 1), Tanh

- Classifier:
  - Fully Connected layers mapping from latent space to 10-class output.
- OOD Detection:
  - Reconstruction error is scaled using a factor (alpha, tuned based on a selected percentile) and used to classify samples as OOD.

5. Hyperparameters

- Learning Rate: 0.001 (Adam optimizer for stable convergence)
- Batch Size: 512 (efficient use of GPU memory)
- Epochs: 120 (extended for better convergence)
- lambda_recon: 2.0 (balances classification and reconstruction loss; we chose to give an advantage to reconstruction ability due to the necessity to identify OOD examples in later stages)
- Alpha: 362 (tuned based on the calibration percentile-based step, post-processing). The chosen percentile was selected to get good results on both CIFAR-10/FashionMNIST OOD sets as well as on sets that can be harder to distinguish from the in-distribution MNIST set.

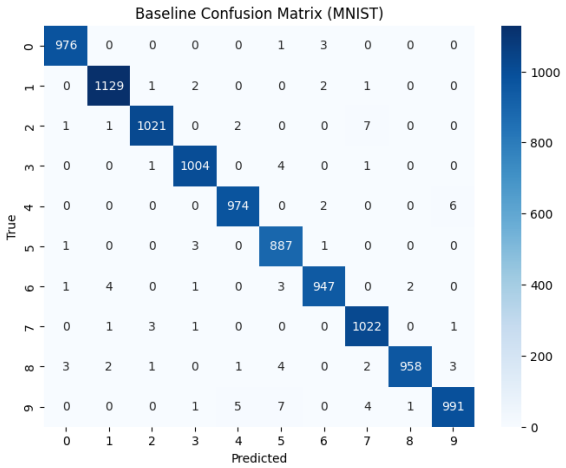
```
loader_full_mnist = DataLoader(mnist_train_full, batch_size=512, shuffle=True) #the full mnist dataset
alpha_opt = calibrate_alpha_percentile(baseline_model, loader_full_mnist, device, percentile=0.75)
print(f"Calibration alpha: {alpha_opt}")
```

```
Calibration alpha: 362.36761474609375
```

6. Evaluation and Results

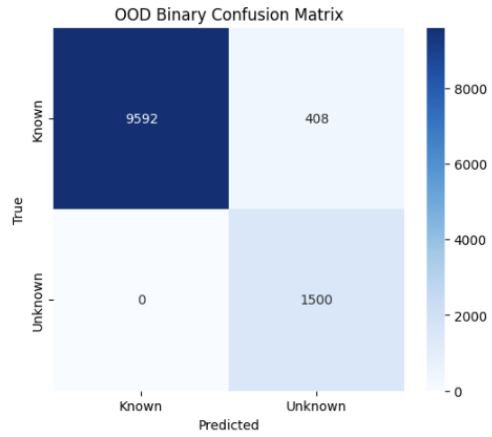
MNIST Classification Performance

- Accuracy: 99.06%
- Confusion Matrix:

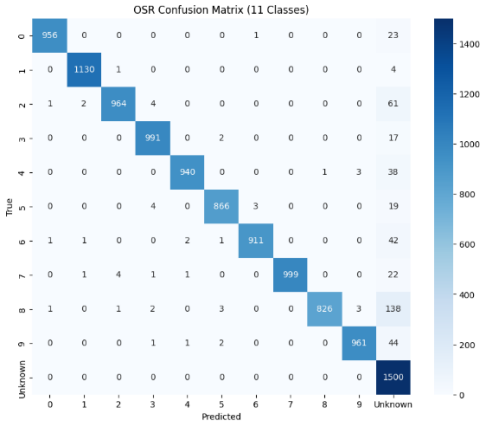


OOD Detection Performance

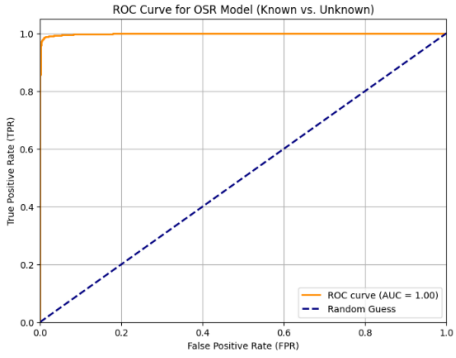
- Binary Classification Accuracy: 96.45% %



- Confusion Matrix for OOD Detection:



- ROC Curve for OOD Detection:



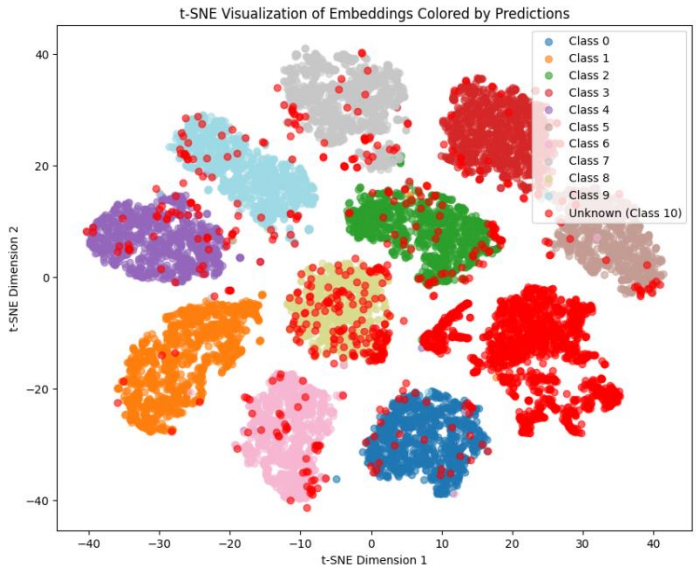
Hyperparameter Tuning for Alpha

- Optimal Alpha Found: 362 -> based on a 75% percentile (for challenging OOD set's)

t-SNE Visualization of Embeddings

- Observations:
    - Clear separation between MNIST classes.

- OOD samples distinctly clustered away from MNIST digits. **Even if some OOD points visually appear within known digit clusters in the 2D plot, the model's **actual** classification is made in the higher-dimensional space (using both the digit logits and reconstruction error). Hence, the t-SNE visualization may show overlap, but under the hood, the model can still reliably distinguish OOD data, leading to the observed 100% OOD accuracy (with CIFAR10+FASHIONMNIST as OOD).



OSR Model Performance

- MNIST Accuracy: 95.44%
- OOD Accuracy: 100%
- Total OSR Accuracy: 96.03%

7. Limitations

- **Hyperparameter Sensitivity:** The performance of the OOD detection mechanism heavily depends on the precise calibration of hyperparameters—particularly the alpha scaling factor and lambda_recon. Small deviations can lead to significant variations in performance, making the model sensitive to the chosen settings.
- **Performance on Complex OOD Samples**: While CIFAR-10 and FashionMNIST work well, some degradation in accuracy may occur with structurally similar datasets.

8. Conclusion

In summary, the multi-task AutoEncoder demonstrates a promising and versatile approach for addressing both classification and OOD detection in a unified framework. By jointly optimizing for reconstruction and classification, the model not only learns robust latent representations but also effectively differentiates between known and unknown samples using reconstruction error. This dual capability significantly enhances the reliability of the system in scenarios where encountering novel or unexpected inputs is inevitable.

Moreover, the model's design underscores the potential of multi-task learning in overcoming traditional challenges associated with open-set recognition. While the approach has shown strong performance on benchmark datasets such as MNIST, FashionMNIST, and CIFAR-10, it also opens up opportunities for further refinement. Future work could explore improvements in hyperparameter tuning and the extension of the framework to handle more complex and diverse real-world datasets.