Project Summary

Participants:
Ilay Yoeli - 318260965
Yael Zorea - 209308675
Yarden Greenpeter - 318517653

Methodology and Rationale

Overview of the approach:

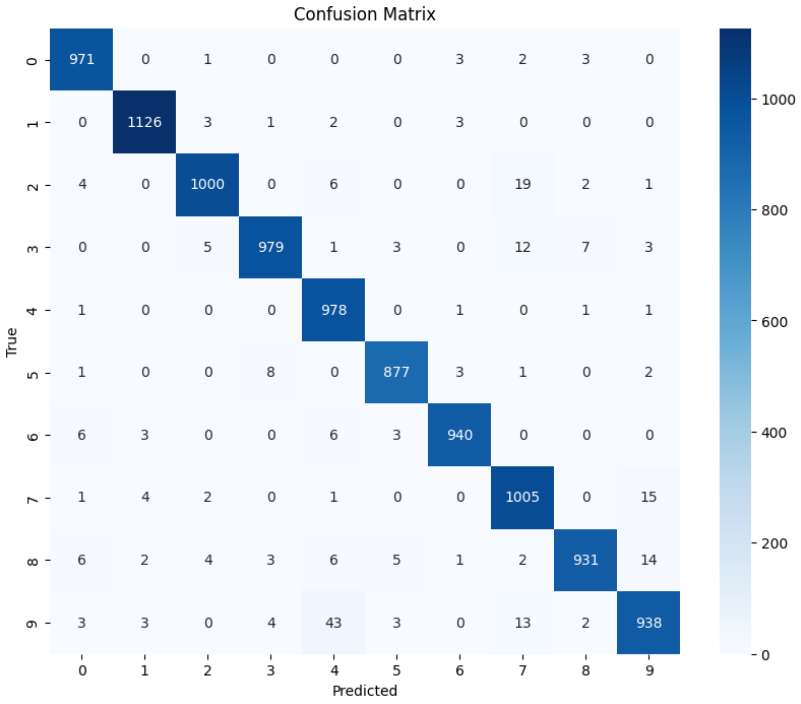
Our model features an autoencoder with both encoder and decoder comprising three convolutional layers each, alongside a classifier that processes the encoded data.
The model is trained on input data that goes through several augmentations and using a loss function that combines the Mean Squared Error (for reconstruction) and Cross-Entropy (for classification), optimized with Adam.
We then refine the model by calibrating the logits with an optimally learned temperature.
Finally, during evaluation, data is classified as out-of-distribution (unknown) if the calibrated maximum probability falls below a set threshold; otherwise, it's classified according to the highest probability label.

Rationale for the Approach (Construction Process and further description):

The auto-encoder was originally chosen as out back-bone architecture of the model for its known effectiveness in the Closed-Set-Recognition task of classifying MNIST digits. Upon further research, we concluded that the auto-encoder, with several tweaks, will also be effective for the Open-Set-Recognition. Our conclusion was based on a paper [1] which presents a model combining both classification error and reconstruction error in the model learning phase. The classification error ensures that the latent representations are discriminative, allowing the model to accurately classify known classes. This helps the model focus on features that differentiate between classes, which is essential for tasks like digit classification. The reconstruction error, on the other hand, encourages the model to capture the underlying data distribution and retain general features that are common across all inputs. This is beneficial in the

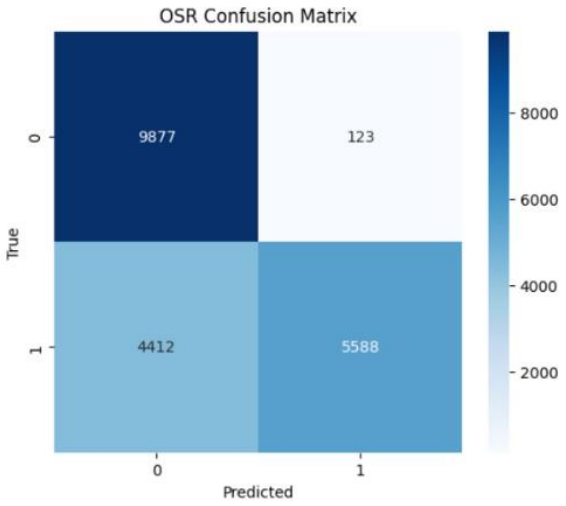
OSR case where the model needs to be certain enough that a given input belongs to one of the known classes.

Implementing validation sets along the training to ensure we avoid over-fitting, the results we achieved for the MNIST test set were encouraging:



Accuracy: 97.45%

We then went on to construct the evaluation model of the OSR, first attempting the simple approach of feeding the input into the original model and checking whether its maximum probability score crosses a certain threshold.
Attempting several different thresholds, few major issues surfaced.
We first encountered low accuracy in Out-Of-Distribution data recognition involving FashionMNIST dataset:

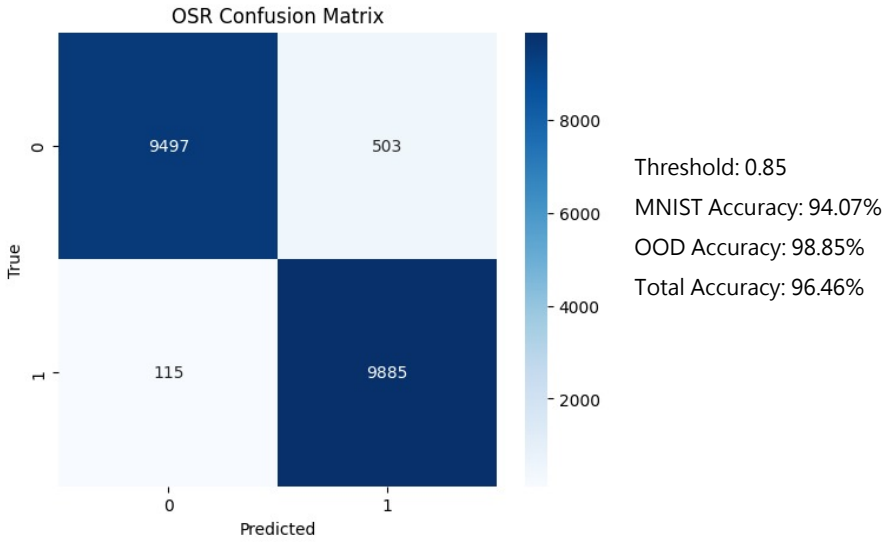


Threshold: 0.9
MNIST Accuracy: 98.45%
OOD Accuracy: 55.88%
Total Accuracy: 77.16%

Noticing the model large portion of the OOD to be '8', we plotted the inputs which were misclassified as "unknown", for inference:

We concluded that this has to with symmetry and, more generally, dissatisfactory feature extraction.

Our solution was adding a rotation to the data augmentations. We incorporated Gaussian noise, color-jitter, and rotations to the data to account for any possible OOD data. The results improved dramatically:



Second issue was some fluctuation in the results as well as a concern that the threshold, which must be empirically selected, would fit only to a certain dataset (as there was a slight difference in the results CIFAR-10 and FashionMNIST datasets).

We then concluded that the probability distribution outputs (=logits) of the original model might not reflect well enough the true likelihood of the predication being correct. We decided to incorporate temperature scaling to the model [2], calibrating the probabilities to better reflect the model confidence.

We incorporated a second stage temperature scaling model which learns an optimal temperature and scales the input logits by it.
The results over both datasets improved in both accuracy and consistency.

References:

[1] Yoshihashi, R., You, S., Shao, W., Iida, M., Kawakami, R., & Naemura, T. (2019). Classification-Reconstruction Learning for Open-Set Recognition. https://arxiv.org/abs/1812.04246

[2] Guo, C., Pleiss, G., Sun, Y., & Weinberger, K. Q. (2017). On Calibration of Modern Neural Networks. https://arxiv.org/abs/1706.04599

Hyper-Parameter Configuration

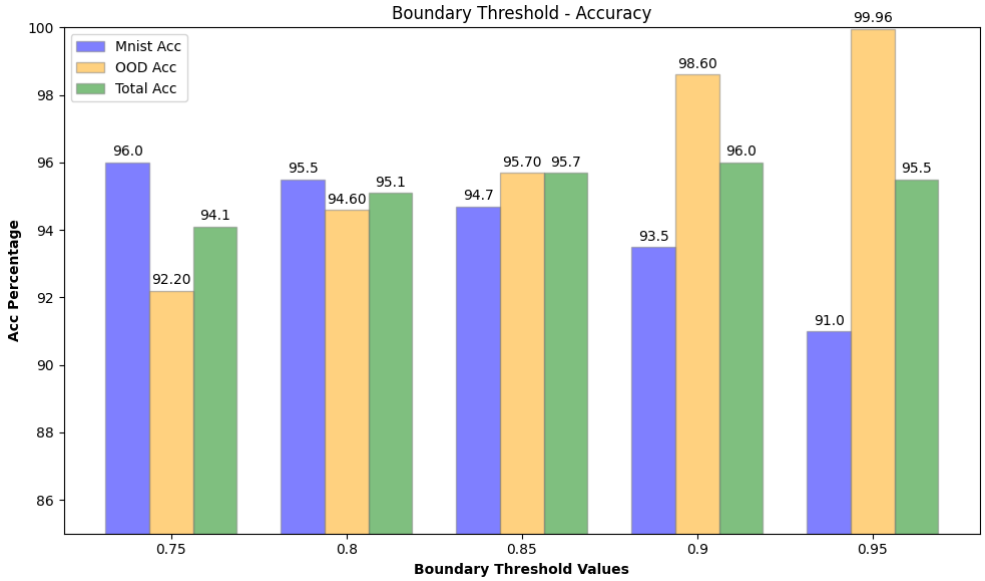
Hyper-Parameters:

Parameter name	value
Learning Rate	0.01
Batch Size	1024
Number of Epochs	5
Base Temperature	1.5
Boundary Threshold	0.85
Temp Scaling Epoch count	10

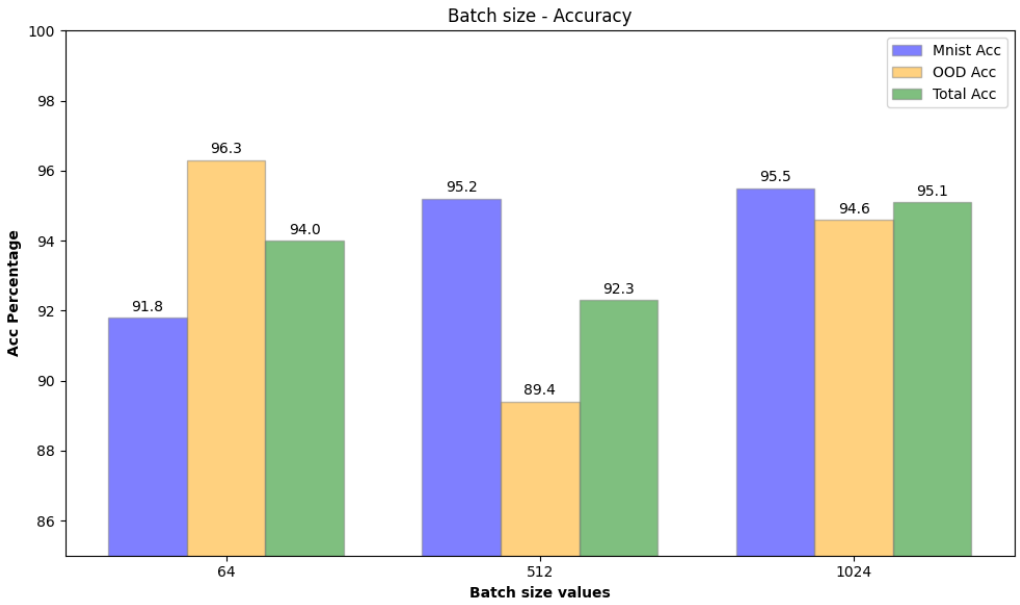
Final Configuration:

The graphical representations below illustrate the partial derivatives with respect to each hyperparameter under investigation. To isolate the effects of individual hyperparameters, we employed a controlled experimental design. In this approach, all parameters were maintained at their baseline values, expect for specific hyperparameter being evaluated. This methodology ensures that any

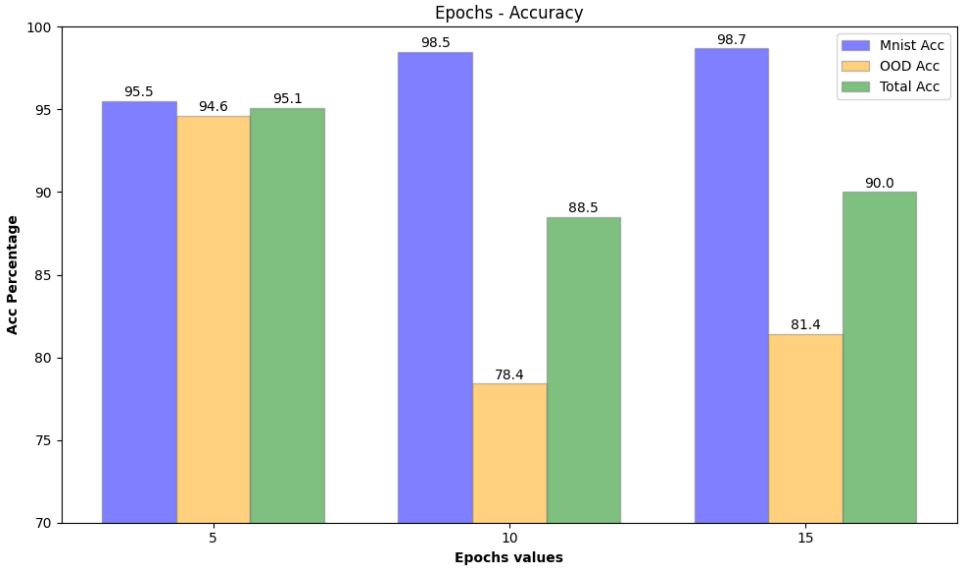
observed variations in the results can be attributed solely to the changes made to the parameter under examination.



Boundary Threshold: in the graph above we can see that there is an exchange between a MNIST accuracy and OOD accuracy when considering different values for the threshold, we saw on average that the value 0.85 gives a good total accuracy over both datasets, we also considered that taking a high threshold might result in high OOD over the fashion dataset, but might not be well suited for other datasets, and it was also important to us to maintain an above 90% accuracy for the MNIST, the value chosen seems as it could take these two considerations.



Batch size: after some training and tuning different hyperparameters we observed that a small batch size could produce better OOD accuracy and cost a lower MNIST accuracy, we found that at value 1024 we got more predictable results and a higher total accuracy, we thought that the higher batch size could improve the model's inference and provide better results with the unknown dataset.



Number of Epochs: We initially set the epoch count to 10 and observed that the model converged well at lower epochs based on validation accuracy. While this could be influenced by the learning rate, we believe a higher epoch count might improve certainty on the MNIST dataset, lowering OOD accuracy. For OSR evaluation, we opted for a lower epoch count to help the model better distinguish between MNIST and the unknown data.

Learning Rate: Initially, the project began with a learning rate of 0.001. Through trial and error, we observed that increasing the learning rate to 0.01 yielded good results with faster convergence. This higher learning rate aligns with standard values we have used in previous works.

Temperature Scaling Epoch Count: when we trained the model on temperature scaling, we observed a lower convergence rate, since it was important to us to have a precise temperature, we opted for a higher epoch count for this training.

Base Temperature: We set the initial temperature to 1.5 to allow flexibility in adjusting overconfident predictions. A temperature of 1.0 means no scaling, and starting below 1.0 would leave less room for downward adjustment to reduce overconfidence. By starting above 1.0, we can better address the model's overconfidence, while allowing the optimization process to fine-tune the temperature based on calibration.

Limitations of the Approach

Identified Limitations:

The main obstacles we thought about:

a) Having trouble distinguishing from unknown dataset that are remarkably similar to the MNIST dataset.

b) The fact that our solution might bring good results on the FashionMNIST \ CIFAR-10 datasets but overall might not be compatible with other datasets. The performance of the OSR model could depend on the fine tuning we made to the hyperparameters, and while performing well on FashionMNIST \ CIFAR-10 might work poorly on the unknown dataset.

Our approach to tackle these limitations:

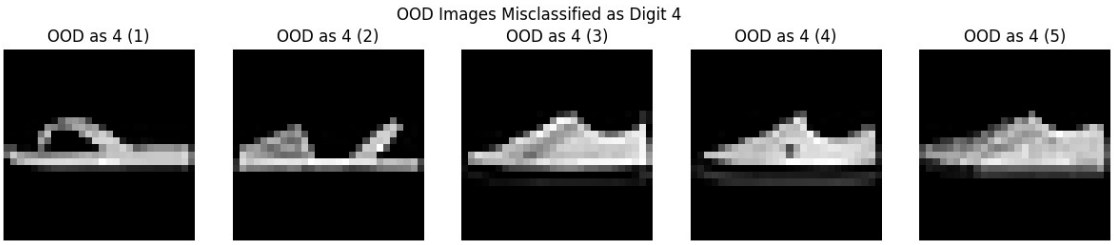
- We used **data augmentation** such as rotation and gaussian noise to the train dataset, so that the model would create a more robust representation of the MNIST images.
- In the OSR decision algorithm, we used a threshold on the probabilities of the classes, so we used **Temperature scaling** to calibrate the output probabilities of the base model. Neural networks, tend to produce overconfident probability estimates, even when they're wrong. This overconfidence can make the predicted probabilities misleading, which is a problem when you need reliable uncertainty estimates.
- At the **hyperparameter tuning stage**, our goal was to achieve a balanced accuracy, paying close attention to avoid making the model overfitting on

the FashionMNIST / CIFAR-10 datasets. As a guideline, we considered an accuracy range of 90%-95% to be optimal. Exceeding this threshold could indicate that the model is too specialized for FashionMNIST or CIFAR-10, which might lead to poor generalization due to excessive parameter tuning.

Data Suitability:

Potentially challenging data:

- We assume that images with distinct or sharp outlines, such as simple icons or letters, might be mistakenly classified as a digit for having similar features.
- In addition, in some tests we run, we observed there were images from the FashionMNIST which were classified as the digit '4':



  Probably due to the similar curve and shape to the digit 4.
- We also predict that images of digits, such as house numbers or license plates, might not be classified as an OOD by the model.

Well-suited data:

- We assume that complex images, such as landscape or images with many objects displayed in them, will be correctly classifies as OOD, due to the lack of similarity to any known class to the model.
- Data like the FashionMNIST / CIFAR-10 datasets, on which the model achieved reliable results.

Conclusion

In conclusion, our approach of combining an autoencoder architecture with temperature scaling and data augmentation proved effective in Open-Set Recognition (OSR) tasks on the tested datasets. By fine-tuning the model's hyperparameters and incorporating temperature scaling to adjust for overconfident predictions, we achieved a high level of accuracy across both known and out-of-distribution datasets. While our model performed well on FashionMNIST and CIFAR-10, some limitations remain, particularly in recognizing data with similarities to MNIST digits. However, the integration of augmentations like rotation and Gaussian noise helped create a more robust model. Our final configuration strikes a balance between accuracy and generalizability, making it a viable solution for OSR tasks while maintaining a strong performance on the MNIST dataset.

For future work, we could explore the use of embedded layers in our model to handle more complex classification tasks. By leveraging these embeddings, the model could potentially capture more nuanced features and relationships within the data, allowing it to achieve higher accuracy on more challenging datasets. This approach would not only enhance the model's performance on difficult classification problems but also improve its ability to generalize across a wider variety of out-of-distribution examples.