Project 9

Computer Vision - A.Y. 2024/2025

Advanced Out-of-Distribution Detection for Multi-Class Classification

Santavicca Federico
2003442

Pedicillo Marco
1983285

Outline

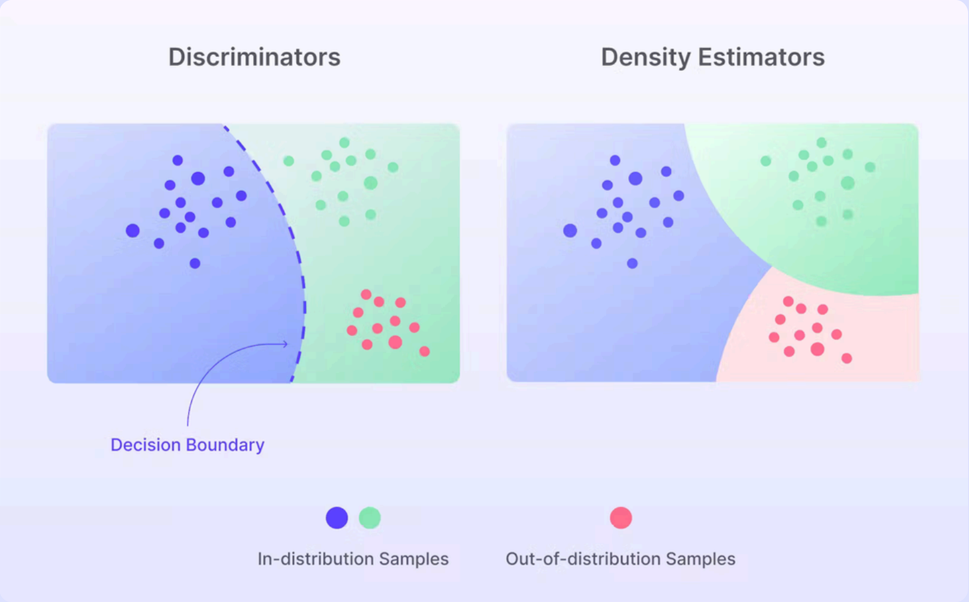
- Problem Statement
- State of the Art
- Dataset
- Our Approach
- Model Evaluation
- Challenge the model
- Conclusions
- References

What is the challenge?

Classification models often encounter **out-of-distribution (OOD)** data, i.e., data that differs from the training distribution.
These inputs can lead to incorrect and unreliable predictions.

The goal of the project is to develop a system that can:

- Recognize known data (in-distribution, Food-101)
- Effectively detect OOD data
- Evaluate the performance of different OOD detection techniques to improve model robustness.



In-Distribution vs Out-of-Distribution

Current Solutions for OOD Detection

Several methods have been proposed to detect OOD inputs in deep learning models.

The main research directions include:

- **Energy-Based Models (EBMs)**: Use the model's output energy score to distinguish between in-distribution and OOD samples.
- **Gradient-Regularized Methods**: Introduce regularization terms based on gradients to improve OOD sensitivity during training.
- **Response-Based Scores**: Use neural response statistics (like activations) to compute OOD scores.

These approaches aim to improve model reliability by better separating known and unknown data distributions.

Datasets

10

Food-101	We have 101.000 images, 1000 images for 101 different categories of food.
SVHN	House Number Images from Google Street View, formed by digits 0-9 (10 classes).
Food Packaging	40.000 images of food packaging for 100 different classes .

Our Approach

Main Libraries:

- PyTorch – Core deep learning framework.
- Torchvision – Datasets, pretrained models, image transforms.
- Scikit-learn – Evaluation metrics (accuracy, precision, recall, ROC AUC, etc.).
- Matplotlib & Seaborn – Plotting and visualization.
- Kaggle – Development environment.

Model Backbone:

- ResNet50, pretrained on ImageNet and adapted to the Food-101 dataset. This header includes a dropout mechanism to reduce overfitting and a linear layer that produces the 101 output classes.

Train in two stages

- **Vanilla:** The model is trained as a standard image classifier on in-distribution (ID) data (Food-101), using only cross-entropy loss to learn class discrimination.

- **Robust:** Outlier Exposure is introduced. The loss function is extended with an energy-based regularization term, encouraging the model to assign:

  - Low energy to ID samples
  - High energy to OOD samples (e.g., SVHN)

  This teaches the model to separate known and unknown inputs based on energy scores.

In both stages, the full ResNet50 is fine-tuned. The second stage guides the model toward OOD-awareness through energy-based learning.

Datasets Split

To effectively train and evaluate our model, we divide the Food-101 dataset into two main parts:

- Training Set (75%): Used to train the model to learn patterns and features from in-distribution (ID) data. This dataset is split into 90% actual train and 10% validation set.
- Test Set (25%):Used only for final evaluation. This set simulates real-world data the model hasn’t seen during training.

In addition, we used two datasets, SVHN and Food Packaging, for robust method in order to include OOD examples in training phase.

Training Phase

Key Components:

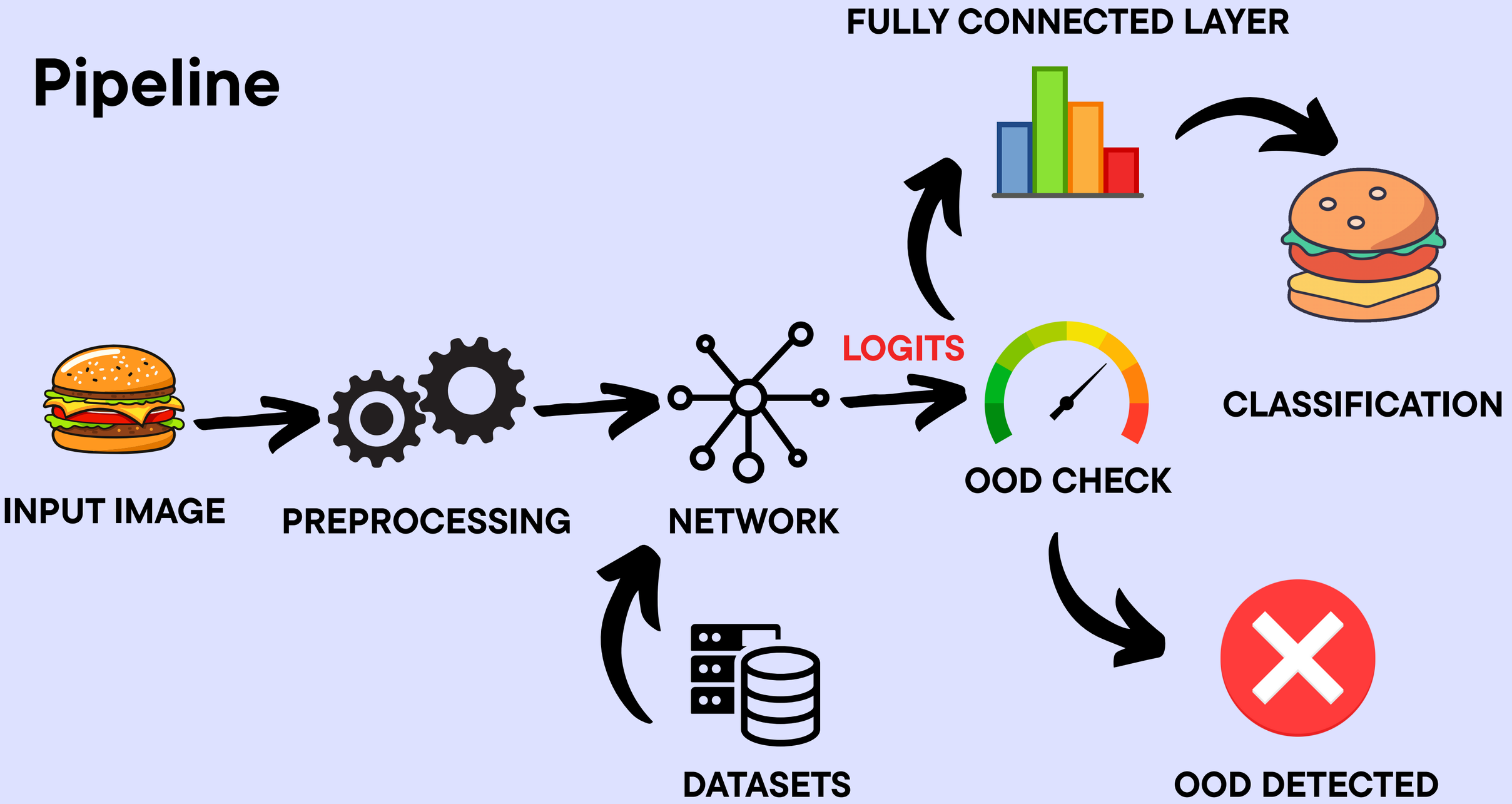
- **CrossEntropyLoss** – Standard loss used to train the classifier on in-distribution classes.
- **Adam optimizer** – Optimizer with weight decay for regularization.
- **StepLR Scheduler** – Reduces learning rate every 5 epochs to stabilize training.
- **Early Stopping** – Monitors validation accuracy and stops training if no improvement is seen.
- **Checkpointing** – Automatically saves the best model weights based on validation performance.
- **Data Augmentation** – Random crops, flips, and color jittering improve generalization during training.

Energy-bounded learning

Key Components:

- **Energy Score** - $E(x) = -T \cdot \log(\sum_n \exp(z_i / T))$
- **Regularization Loss** - Combines CrossEntropy (ID) with a hinge-based energy loss :
  - Enforces low energy for ID samples: $E(x) < m\_in$
  - Enforces high energy for OOD samples: $E(x) > m\_out$
- **λ (lambda)** - Weight controlling the influence of energy regularization in the total loss.
- **Margins** -
  - m_in = upper energy bound for ID (e.g., −6.0)
  - m_out = lower energy bound for OOD (e.g., −1.0)

Pipeline


FULLY CONNECTED LAYER
LOGITS
CLASSIFICATION
INPUT IMAGE
PREPROCESSING
NETWORK
OOD CHECK
DATASETS
OOD DETECTED

Model Evaluation

- Accuracy
- ROC Curve
- Precision & Recall
- Energy Score Distribution
- Softmax Distribution

Accuracy

Accuracy measures the proportion of correctly classified samples over the total number of samples.

$$\text{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN}$$

A high accuracy score **doesn't necessarily** imply robustness against OOD inputs, which may still be confidently misclassified.

These are the values for accuracy and other useful metrics:

Vanilla:

(ID = Food-101, OOD = SVHN)

- AUROC (Energy): 0.8772
- AUROC (Softmax): 0.9921
- **FPR@95TPR (Energy): 3.79%**
- **FPR@95TPR (Softmax): 4.30%**
- AUPR-In (Energy): 0.9511
- AUPR-In (Softmax): 0.9958
- **Final Test Accuracy: 87.54%**

Robust:

(ID = Food-101, OOD = SVHN)

- AUROC (Energy): 0.9934
- AUROC (Softmax): 0.9998
- **FPR@95TPR (Energy): 0.06%**
- **FPR@95TPR (Softmax): 0.09%**
- AUPR-In (Energy): 0.9961
- AUPR-In (Softmax): 0.9998
- **Final Test Accuracy: 87.47%**

The main difference can be observed in the FPR@95, which indicates how many OOD samples are incorrectly classified as ID when the model can correctly detect 95% of the ID samples.

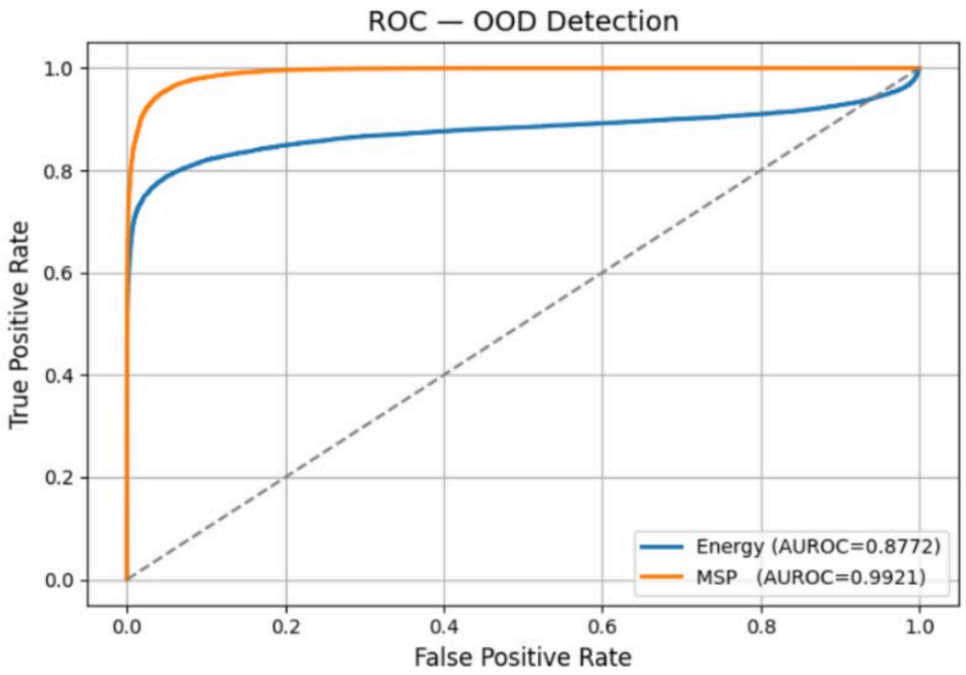
ROC Curve

The ROC curve shows the trade-off between the True Positive Rate and the False Positive Rate at various classification thresholds.

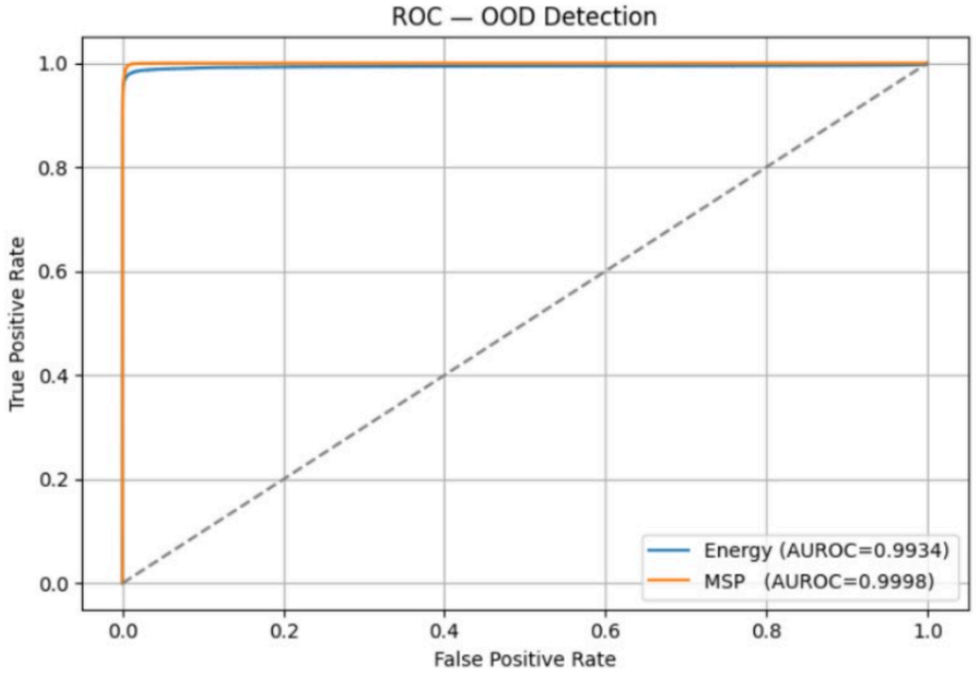
A point on the ROC curve shows how well the model distinguishes between classes at a specific threshold.

The closer the curve is to the top-left corner, the better the performance.

Vanilla:



Robust:



Precision & Recall

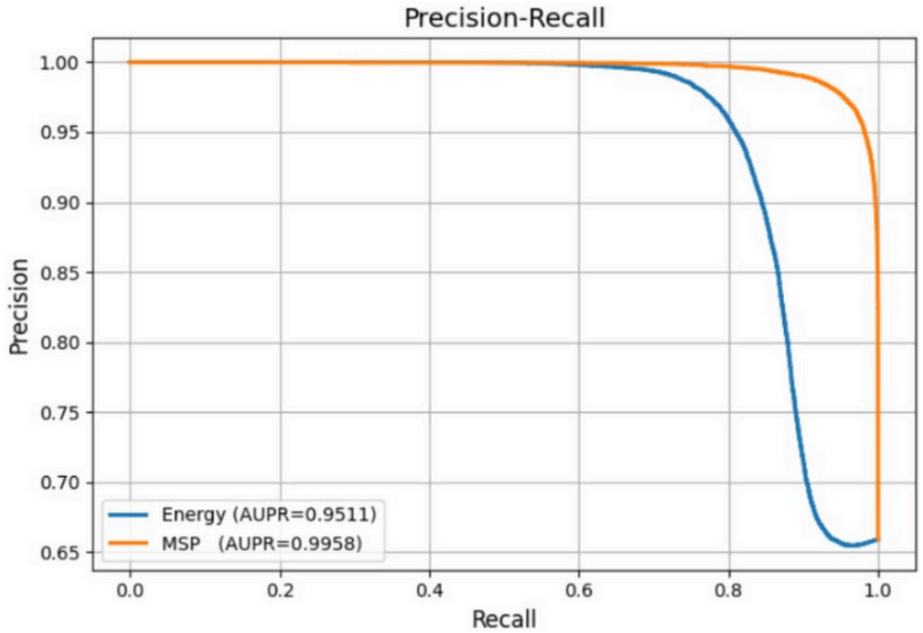
- Precision measures how many of the predicted positive results are actually correct.

$$\text{Precision} = \frac{TP}{TP + FP}$$

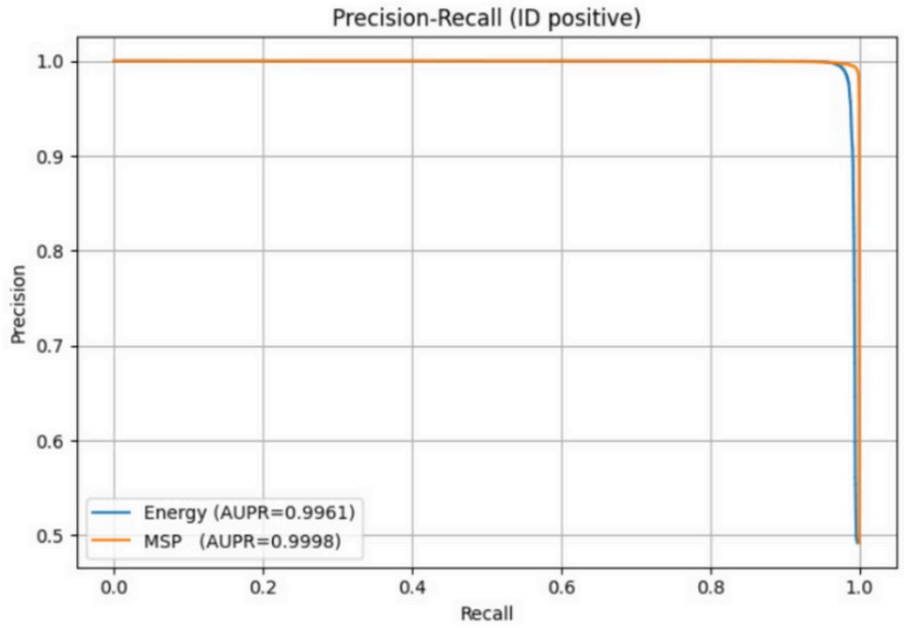
- Recall measures how many of the actual positive cases were correctly predicted.

$$\text{Recall} = \frac{TP}{TP + FN}$$

Vanilla:



Robust:



Energy Score Distribution

The Energy Score quantifies the model's confidence about an input sample:

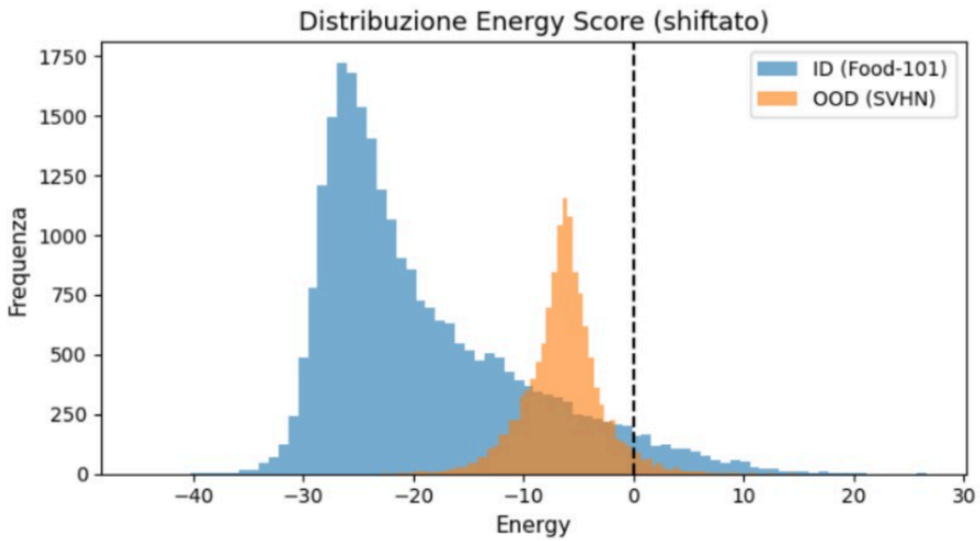
- Lower energy = more confident (likely ID)
- Higher energy = less confident (likely OOD)

It's computed as:

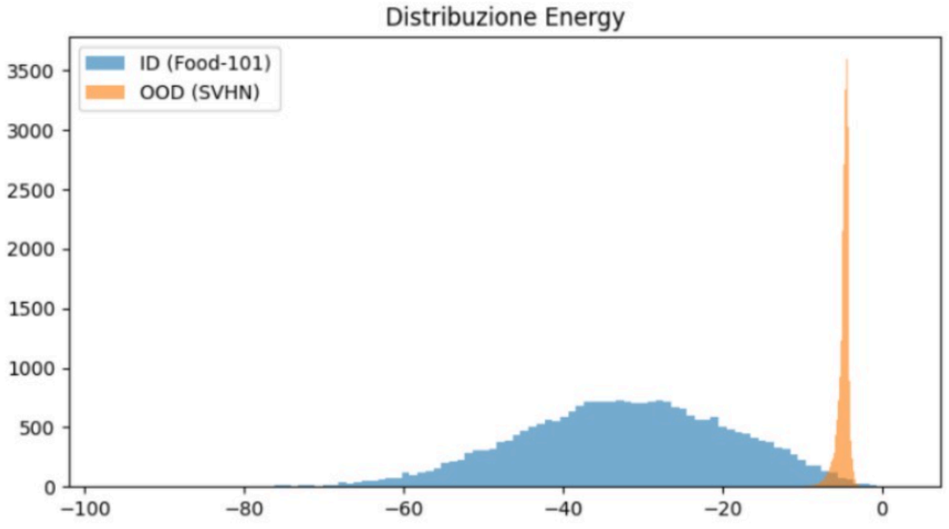
$$E(x) = -T \cdot \log \sum_i e^{z_i/T}$$

where z_i are the model’s logits and T is a temperature parameter.

Vanilla:



Robust:



Softmax Distribution

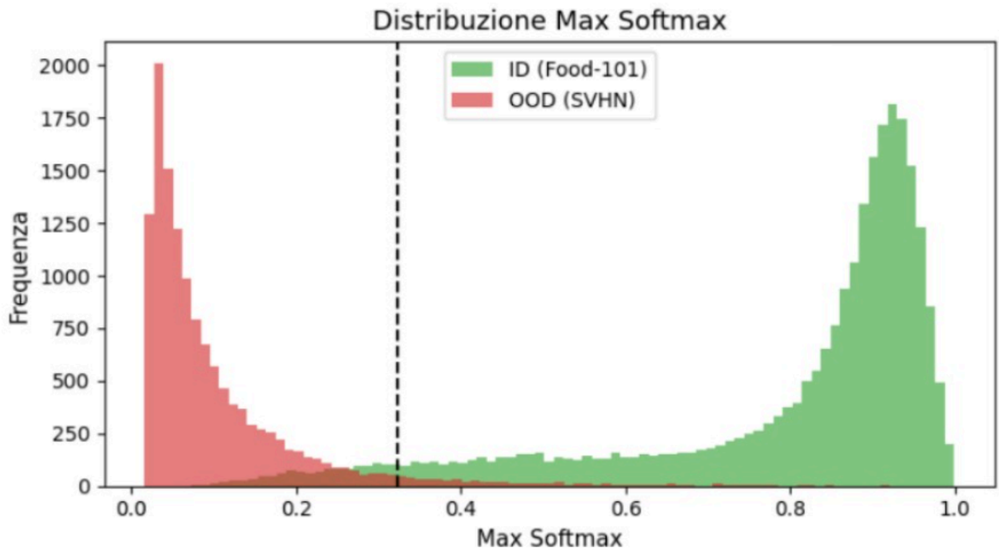
The Softmax Score represents the maximum class probability predicted by the model.
It reflects how confident the model is in its top-1 prediction.

It's computed as:

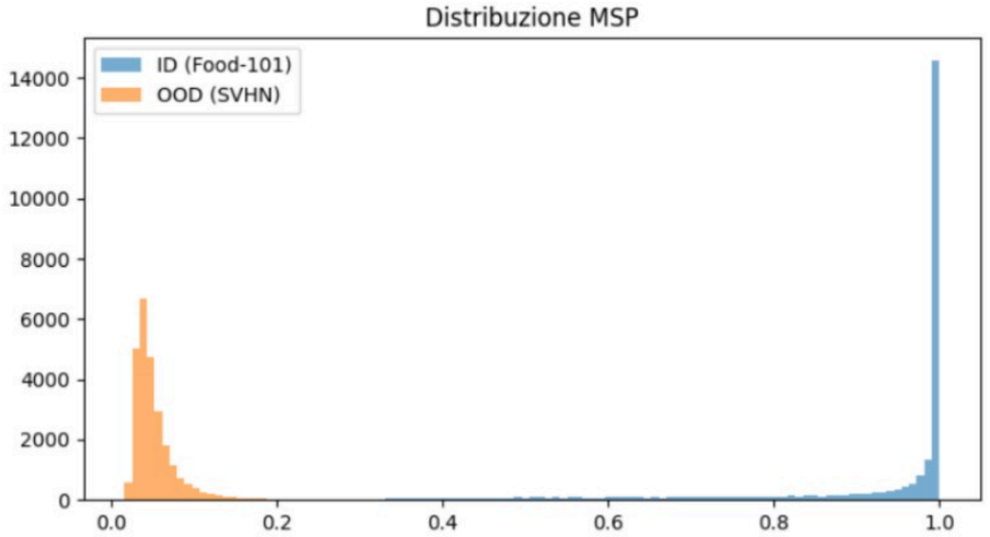
$$\text{Softmax}(x) = \max_i \left(\frac{e^{z_i}}{\sum_j e^{z_j}} \right)$$

where z_i are the logits of each class.

Vanilla:



Robust:

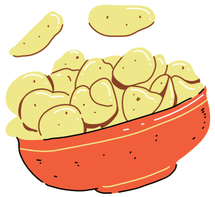


Food Packaging

Given the high performance in distinguishing Food-101 from SVHN, we decided to further challenge the model by introducing a new dataset consisting of food packaging images.

The model, trained only to classify food categories, appeared exceptionally good at rejecting non-food images, perhaps too good, raising doubts about whether SVHN was a sufficiently challenging OOD dataset.

By using a dataset with **visually similar** but semantically different images, we aim to verify whether this performance holds in a more realistic and ambiguous setting.

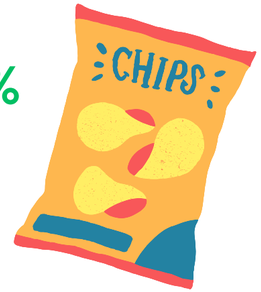


(ID = Food-101, OOD = SVHN)

- AUROC (Energy): 0.8772
- AUROC (Softmax): 0.9921
- **FPR@95TPR (Energy): 3.79%**
- **FPR@95TPR (Softmax): 4.30%**
- AUPR-In (Energy): 0.9511
- AUPR-In (Softmax): 0.9958

(ID = Food-101, OOD = Food Packaging)

- AUROC (Energy): 0.9369
- AUROC (Softmax): 0.8821
- **FPR@95TPR (Energy): 62.32%**
- **FPR@95TPR (Softmax): 48.02%**
- AUPR-In (Energy): 0.9225
- AUPR-In (Softmax): 0.8043



SVHN is too visually different from food, making detection easier. Food Packaging, instead, is semantically similar and therefore exposes the model's limitations in fine-grained OOD detection.

FPR@95TPR (Energy) increases drastically from 3.79% to 62.32%, suggesting a much higher false positive rate when facing visually similar out-of-distribution samples.

Conclusions

The results demonstrate that the model is capable of effectively distinguishing between in-distribution and out-of-distribution (OOD) data, achieving strong accuracy and solid OOD detection metrics. A key factor contributing to these results is the significant difference in nature between the ID and OOD datasets.

However, there are several areas for improvement and many interesting directions for further analysis:

- **Robust Testing**: Incorporate the Food Packaging dataset into the training phase to evaluate whether including more challenging OOD samples can help mitigate the false positive problem;
- **Infrastructure**: Consider switching to Colab Pro and moving away from Kaggle (a nightmare);
- **Hyperparameter Tuning**: Conduct a more thorough grid search, especially to optimize key parameters like temperature;
- **Dataset Diversification**: Experiment with combining different OOD datasets to assess the impact on generalization and robustness.

References

- **Liu, W., Wang, X., Owens, J. D., & Li, Y. (2020). Energy-based Out-of-distribution Detection.**
- Sharifi, S. et al. (2024). Gradient-Regularized Out-of-Distribution Detection.
- Tang, K., et al. (2024, June). CORES: Convolutional Response-based Score for Out-of-distribution Detection.

Thank you for your attention