Image Classification under Distribution Shift: Food-101 to In-the-Wild Evaluation

Iacob Iasmina

1. Overview & Design

This project implements a deterministic, reproducible food image classification pipeline designed to evaluate robustness, calibration, and failure modes under distribution shift. ImageNet-pretrained models are trained on Food-101 and evaluated under synthetic corruptions and real-world "in-the-wild" images collected from Wikimedia Commons. All experiments are config-driven, reproducible, and runnable via CLI and Docker.

2. Dataset and Exploratory Data Analysis

Food-101 (ethz/food101 via Hugging Face Datasets) contains 101 food categories with noisy labels, great class balance, and provides a validation set of 250 images per class.

Detailed EDA artifacts are provided in *outputs/baseline_resnet18/*. Findings:

- Class counts: uniform (min = max = 750 per class)
- Image resolutions: variable, but cluster around a consistent upper bound
- Preprocessing choice: resize and crop to 224×224, standard ImageNet normalization

3. Models and training setup

We use ImageNet-pretrained backbones (ResNet-18, EfficientNet-B0), with the classifier head replaced in order to match the 101 food classes.

Training configuration (5 epochs, max 300 steps/epoch — resource-safe):

- Optimizer: Adam
- Loss: Cross-Entropy (baseline), Generalized Cross-Entropy (from scratch)
- Augmentations: resize + center/random crop, horizontal flip, color jitter
- Mixed precision (AMP)
- Strict determinism: global seed = 1337 (Python, NumPy, PyTorch CPU/CUDA), deterministic PyTorch flags, seeded DataLoader (generator + workers), config-driven transforms

To test representation capacity under identical training conditions, EfficientNet-B0 was added, for the same data, optimizer, augmentations, and step budget. Under the fixed training budget, representation capacity dominates performance (+20% Macro-F1).

Model	Val Macro-F1
ResNet-18	~0.47
EfficientNet-B0	**~0.67**

Table 1. Difference in Macro-F1 values across models

Training logs can be found at *outputs/*/train_history.csv*.

While a command-line interface supports batch inference on arbitrary image folders, the same interface runs inside Docker without modification.

4. Public "Wild" Dataset Construction

To evaluate real-world domain shift, we programmatically collect ~300 images across 10 food classes from Wikimedia Commons, using the MediaWiki API. License metadata is preserved, and the "wild" dataset is used strictly for evaluation. The artifacts exist at *data/wild_images/*, including a *wild_metadata.csv*.

5. Robust Loss & Imbalance Handling

Generalized Cross-Entropy (GCE) was implemented from scratch due to its ability to interpolate between cross-entropy and MAE. It is designed to be more tolerant to label noise by down-weighting low-confidence examples during training; we use $q=0.7$, a commonly adopted value in prior works. In addition, we incorporate effective number reweighting, in order to mitigate class imbalance by down-weighting frequent classes without over-amplifying rare ones.

Model	Val Macro-F1
ResNet-18 (CE)	~0.47
ResNet-18 (GCE + eff-num)	~0.16

Table 2. Difference in Macro-F1 values after loss and imbalance handling

Artifacts can be found at *outputs/medium5_gce_effnum_resnet18/train_history.csv*. This outcome is expected: Under a short training budget on a balanced dataset, GCE underperforms cross-entropy in terms of clean Macro-F1. This reflects the known robustness—accuracy tradeoff of noise-tolerant losses rather than an implementation failure.

6. Calibration via Temperature Scaling

Temperature scaling is learned on validation logits using own implementation (LBFGS, scalar T).

ResNet-18 (CE)			EfficientNet-B0		
Metric	**Before**	**After**	**Metric**	**Before**	**After**
ECE (15 bins)	0.0433	**0.0292**	ECE (15 bins)	0.0231	**0.0178**
NLL	2.0155	**2.0103**	NLL	1.1963	**1.1953**
Temperature	–	**1.071**	Temperature	–	**1.033**

Table 3. Temperature scaling values

The results (available at *outputs/*/temperature_scaling.json* and *outputs/*/calibration_reliability.png* — calibration plot) show that both models are over-confident; temperature scaling consistently improves calibration with minimal effect on accuracy. Unit tests were done in order to cover empty bins and extreme logits.

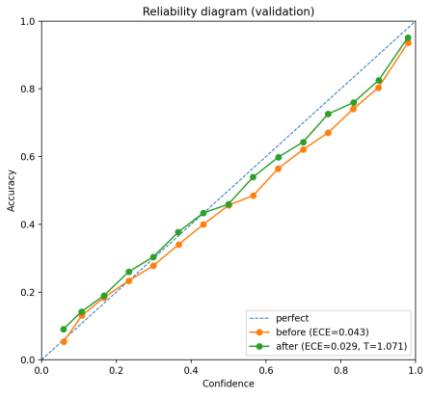

Figure 1. Reliability diagram before and after temperature scaling

7. Corruption Robustness

Evaluation uses Gaussian noise, blur, brightness, and JPEG compression at severities 1–5. Metrics used are Top-1 accuracy and Macro-F1. Artifacts are viewable at *outputs/*/corruptions.csv*.

Corruption (severity 5)	ResNet-18	EfficientNet-B0
Gaussian Noise	~0.30	~0.37
Blur	~0.39	~0.54
Brightness	~0.27	~0.57
JPEG	~0.37	~0.51
Clean (no corruption)	**~0.47**	**~0.67**

Table 4. Model differences in corruption evaluation

EfficientNet-B0 consistently outperforms ResNet-18 across all corruption types and severities, with smooth degradation as severity increases. On wild data, domain shift dominates over synthetic corruption effects.

8. Failure Analysis

Confusion matrices highlight a clear gap between in-distribution and out-of-distribution behavior. On Food-101, the matrix (*outputs/easy2_ce_resnet18/confusion_val.png*) shows a strong diagonal, with errors concentrated among visually similar classes. In contrast, the wild dataset exhibits a much weaker diagonal (*outputs/easy2_ce_resnet18/confusion_wild.png*), reflecting substantial domain shift and partial class collapse.

Frequent confusions (*outputs/easy2_ce_resnet18/top_confusions.csv*) —such as *tiramisu-chocolate cake*, *ice cream-frozen yogurt*— arise from ingredient overlap, texture similarity, and plating style rather than random noise. A qualitative failure gallery was created, along with diagnosis for each image (*outputs/easy2_ce_resnet18/failure_gallery/, outputs/easy2_ce_resnet18/failure_gallery/README.md*) further shows that many mistakes occur with very high confidence (>0.9), often due to presentation bias, loss of contextual cues in close-up images, or dataset noise (ingredient-only or packaging images). These high-confidence failures motivate the need for both improved calibration and explicit out-of-distribution awareness.

Potential fixes include incorporating fine-grained visual attributes (ingredients, textures), expanding training data to cover more diverse real-world conditions, using hierarchical or multi-label supervision for overlapping food categories, and introducing OOD detection or abstention mechanisms to limit overconfident predictions on unfamiliar inputs.

9. Cross-Dataset Generalization and OOD detection

We evaluate out-of-distribution (OOD) detection by training on Food-101 and testing on two external datasets: a "wild" image set collected from Wikimedia Commons and a small Roboflow Universe classification dataset. Wikimedia images include per-image license metadata, and the selected Roboflow dataset is distributed under a permissive research-compatible license as documented on its dataset page.

Because external datasets do not share identical label vocabularies with Food-101, explicit label harmonization is applied. Roboflow class names are mapped to Food-101 labels using a curated mapping, and samples without a clear semantic correspondence are discarded. Due to limited overlap, only 26 Roboflow samples remain after harmonization; these are retained as a small but clean OOD set rather than introducing noisy label mappings.

OOD detection is evaluated using Maximum Softmax Probability (MSP), Energy score, and Mahalanobis distance, reporting AUROC and AUPRC against the Food-101 validation set. MSP (*outputs/d1_ood/ ood_results_msp.json*) performs best across both OOD datasets (e.g., AUROC 0.59 on Wikimedia and 0.55 on Roboflow), while Energy (*outputs/d1_ood/ ood_results_energy.json*) and Mahalanobis (*outputs/d1_ood/ ood_results_mahalanobis.json*) are closer to chance under strong domain shift. Although temperature scaling improves in-distribution calibration, OOD samples still receive high-confidence predictions, indicating calibration drift under distribution shift.

10. Summary

This work demonstrates that strong ImageNet-pretrained backbones substantially improve robustness, while post-hoc temperature scaling reliably reduces miscalibration. Corruption evaluations expose expected degradation patterns, and failure analysis reveals that real-world errors are driven primarily by domain shift and visual ambiguity rather than random noise. While robust losses require careful tuning and do not outperform cross-entropy on this balanced benchmark under short training, the analysis highlights clear directions for improvement, including stronger augmentations, longer training, domain adaptation, and explicit out-of-distribution handling. Overall, the project prioritizes reproducibility, interpretability, and honest evaluation, with all experiments fully deterministic, config-driven, Dockerized, and supported by unit tests (see *MANIFEST.md*).