Defense-in-Depth for Dietary Safety

A Two-Stage On-Device / Cloud AI Architecture for Gluten Classification of Food Ingredient Lists

Technical Whitepaper

Model Generations v1 → v4

Fabian Stöhr

Managing Director, DKS Analytics GmbH

For individuals managing coeliac disease, an ingredient list is a safety-critical document. This paper describes the engineering philosophy, architecture and evaluation methodology behind the GlutenFreeToday classifier — a system designed so that no single component has to be perfect for the user to stay safe.

GlutenFreeToday

a brand of **DKS Analytics GmbH**

www.dks-analytics.de · www.glutenfreetoday.de

July 4, 2026

Marketing-oriented technical disclosure. Certain implementation details are intentionally withheld.

Abstract

For people with coeliac disease, trace quantities of gluten are harmful, and the internationally recognised "gluten-free" threshold of 20 ppm is defined by the international food standard [1, 2]. Reading a dense, fine-print, often multilingual ingredient list under supermarket conditions is exactly the kind of error-prone task where machine assistance helps — provided the assistance is trustworthy. GlutenFreeToday approaches this as an **ingredient-text classification problem**: the system reads and classifies label text as an informational aid; it is not a medical device and makes no diagnostic claims.

We present the classification system behind **GlutenFreeToday**, built on a **two-stage, defense-in-depth** principle. A compact **on-device classifier** runs **fully locally**, producing an instant, privacy-preserving *Quick Scan* verdict (gluten-free ↔ contains gluten) at zero marginal cost. A cloud-based **Deep Analysis** stage can be **triggered by the user** whenever higher assurance is wanted — a deliberate *human-in-the-loop* step that examines every ingredient separately and returns a detailed, source-backed rationale. Safety emerges from the *cascade* of two independent judgements rather than from any single model.

The design is deliberately **cost-sensitive**: a false negative (declaring "gluten-free" when gluten is present) is the only dangerous error, so the operating point is calibrated to prioritise recall at the safety threshold [4]. Our latest generation (v4) roughly **halves the false-alarm rate versus its predecessor (13.6 % → 6.8 %)** on a leakage-controlled, disjoint gold set, **with no false negatives observed in either generation** (observed recall 100 % on this sample; Wilson lower bound 92.6 %). The model quantises to **INT8** at **≈106** MB (∼4× compression) **with no observed increase in false negatives**, enabling fully offline, on-device inference. In live operation, the binary agreement between the two stages is monitored continuously and currently sits at **≈93** %.

Scope note. This document describes philosophy, architecture principles and results. It intentionally does *not* disclose the base checkpoint, training hyperparameters, dataset composition, extraction patterns, decision thresholds or the deterministic safety-net lexicon. The defensible moat is data, labelling and calibration — not the model class.

1 Motivation and Context

Coeliac disease is an immune-mediated enteropathy triggered by the ingestion of gluten in genetically susceptible individuals; its management is lifelong and strictly dietary [3]. The relevant fact for a classification system is that the harm function is steep at very low doses: even trace contamination can be damaging, which is why the Codex Alimentarius standard fixes "gluten-free" at ≤ 20 ppm [1, 2]. A tool whose output users act on when choosing food therefore has to be engineered to a far stricter error budget than a typical consumer feature.

Scope. GlutenFreeToday is an *informational reading aid*: it recognises and classifies printed ingredient text. It is not a medical device, performs no diagnosis, and does not replace the product label or professional dietary advice. The engineering question this paper addresses is a feasibility one — how reliably can photographed ingredient lists be classified on a phone?

Why on-device. The moment of decision is in the aisle: possibly offline, always time-pressured, and involving personal health data. On-device inference addresses all three — it removes network latency, works without connectivity, and keeps the ingredient text on the phone, aligning with privacy-by-design and GDPR data-minimisation principles [25, 26]. Edge deployment is not a convenience feature here; it is part of the trust proposition.

2 Problem Formulation: Safety First

We frame the task as **binary classification** over German-language ingredient text: does the product *contain gluten* (positive class) or is it *gluten-free* (negative class)? The pipeline is nevertheless **multilingual in practice**: when a non-German label is detected, the extracted text is **translated to German on-device** before classification, so a single, well-calibrated German classifier serves all input languages — rather than maintaining one model per language. What makes the problem distinctive is not the label space but the **asymmetry of error costs**.

What the labels mean. The two classes refer strictly to **declared ingredients**: "contains gluten" means a gluten-bearing ingredient is declared on the label; "gluten-free" means none is. This is **ingredient-declaration classification, not laboratory analysis** — the system makes no statement about ppm levels, production conditions or actual contamination, and it is not a substitute for the 20 ppm analytical standard [1]. Precautionary trace statements ("may contain traces of gluten") are **not** classified as containing gluten, following the guidance line of the German Coeliac Society (Deutsche Zöliakie-Gesellschaft, DZG) [24], under which such precautionary labelling does not, by itself, make a product unsuitable for a gluten-free diet. Trace statements are surfaced to the user as information rather than folded into the binary verdict. This keeps the classification task well-defined and honest about what an ingredient list can and cannot tell.

Let $y \in \{0, 1\}$ be the true label (1 = contains gluten) and $\hat{y}$ the prediction. The two error types are far from symmetric:

- **False Negative** ($\hat{y} = 0$, $y = 1$): the system says "gluten-free" while gluten is present. This is the *only* dangerous error — it can directly harm the user.
- **False Positive** ($\hat{y} = 1$, $y = 0$): the system flags a safe product. This is inconvenient but harmless.

Assigning costs $c_{\mathrm{FN}} \gg c_{\mathrm{FP}}$, cost-sensitive decision theory [4] tells us the optimal decision threshold $t^\star$ on the model's gluten-probability $p = \Pr(y = 1 \mid x)$ shifts *below* the naïve 0.5:

$$t^\star = \frac{c_{\mathrm{FP}}}{c_{\mathrm{FP}} + c_{\mathrm{FN}}}, \qquad c_{\mathrm{FN}} \gg c_{\mathrm{FP}} \implies t^\star \ll 0.5. \tag{1}$$

Intuitively, we predict "contains gluten" unless the model is quite confident the product is safe. This is a Neyman–Pearson-style stance: minimise false positives *subject to* keeping the false-negative rate at or near zero. The concrete value of $t^\star$ is a calibrated safety parameter and is deliberately not disclosed here. A short derivation of Eq. (1), together with two corollaries that connect it to the calibration requirement and the cascade bound, is given in Appendix B.

3 System Architecture

The system is a cascade of complementary stages, each cheap where it can be and thorough where it must be. Figure 1 gives the high-level, implementation-agnostic view.

3.1 Stage 1 — On-device classifier

Raw OCR output is passed through a heuristic ingredient-extraction step with strict **train/serve parity**: the exact same preprocessing runs in training and in the app, so the model never sees a distribution at inference time that it did not see during training. If the label is not in German, **language detection and on-device translation to German** run first — entirely locally, preserving the offline and privacy guarantees. The extracted (German) text is scored by a compact transformer-based encoder, fine-tuned for this task and deployed as an **INT8** model. A single predicted probability is then mapped to a three-state display (green / "unclear" / red); the middle state is a *UX rendering of uncertainty*, not a third learned class. A **deterministic safety net**

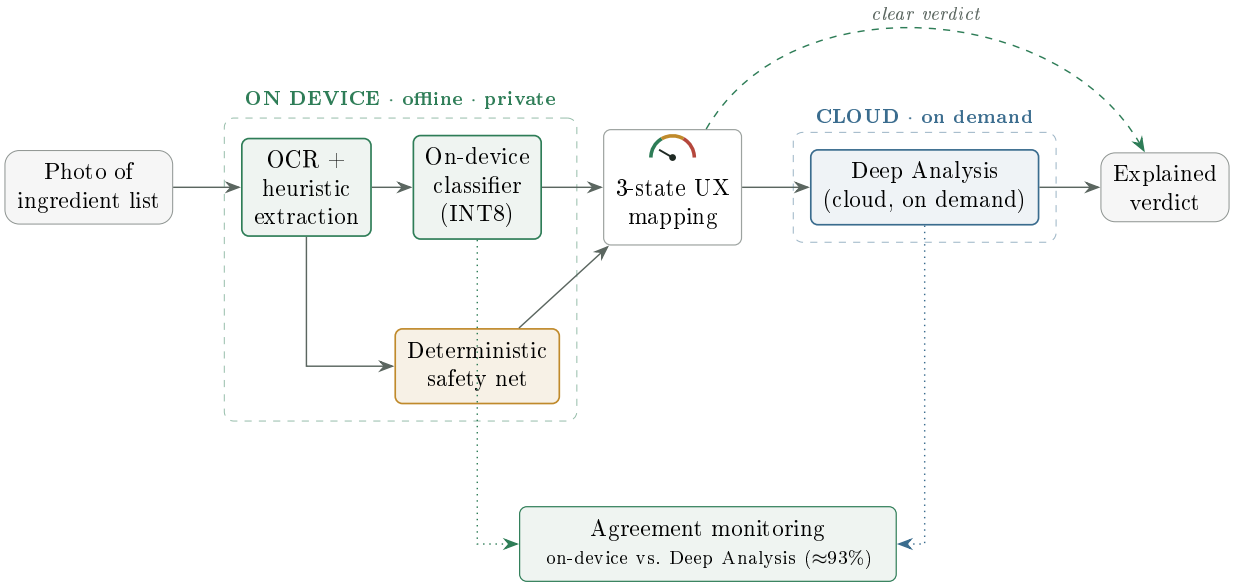

Figure 1: High-level architecture. Stage 1 (green) runs entirely on the device: OCR and heuristic extraction feed a quantised classifier whose single probability is mapped to a three-state UX signal (green / "unclear" / red). A deterministic safety net covers rare, unambiguous gluten grains. Stage 2 (blue), the cloud Deep Analysis, verifies cases on the user's request (human in the loop) and returns a source-backed explanation. Their binary agreement is monitored continuously as an operational quality signal. Implementation details of each block are intentionally omitted.

complements the learned model: a small set of unambiguous, high-risk tokens deterministically overrides the classifier towards "contains gluten" in safety-critical cases — it can only make the verdict more conservative, never less. The token list itself is withheld. The specific architecture family follows the now-standard masked-language-model (BERT) lineage [5]; the concrete checkpoint and patterns are withheld.

3.2 Stage 2 — Deep Analysis: the human in the loop

Escalation to the second stage is **initiated by the user**, not by an opaque automatic routing rule. Whenever a person wants a higher level of assurance — an unclear verdict, an unfamiliar product, a higher-stakes decision — they deliberately trigger Deep Analysis. This is a **human-in-the-loop** design [19]: the person retains control over when the more thorough check runs, instead of the system silently deciding for them.

A cloud-based analysis stage then examines **every ingredient separately** — not just the aggregate verdict — and checks beyond gluten for **cross-allergen and contamination risks**, taking the person's dietary context into account (intolerances, allergies, diet form, pregnancy / breast-feeding). Privacy follows strict **data minimisation**: only the ingredient text and the dietary parameters required for the analysis are processed in the cloud — **no identifying user data**. The result is an itemised, per-ingredient rationale (which ingredient, why, what risk), **backed by cited, publicly accessible sources** that are shown to the user. The citations are *verification aids*: they let the person check the reasoning against the referenced material rather than trust a black box — they are not a guarantee of correctness, and unresolved ambiguity is surfaced as such instead of being smoothed over. This is the human-interpretable, auditable layer of the system — deliberately a complement to the on-device classifier, not the centre of the architecture.

3.3 Why the cascade is safer

The cheap stage scales to *every* scan; the thorough stage double-checks the cases the *user chooses* to escalate. No single model has to be perfect. This is the classical cascade principle from detection

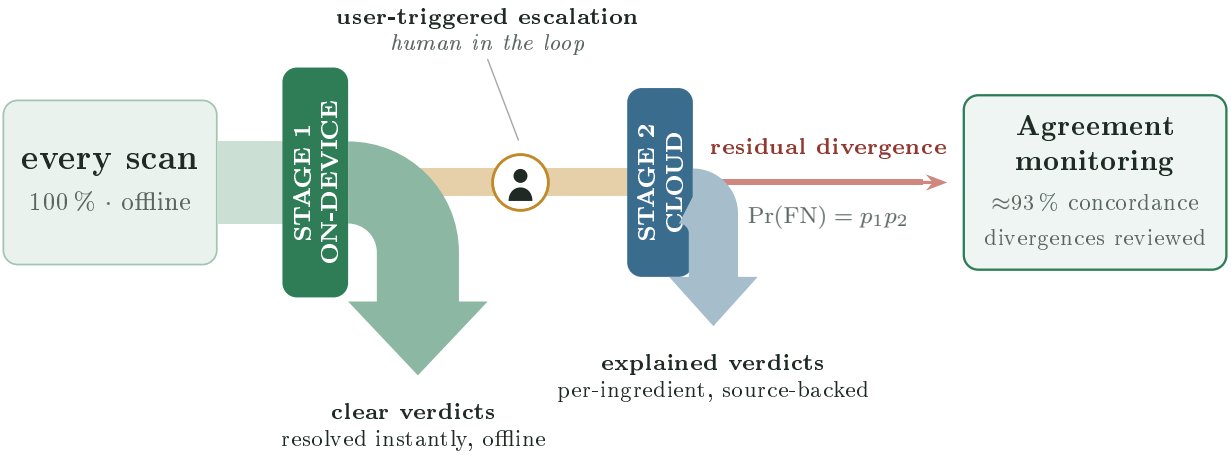

Figure 2: The safety funnel. Every scan passes the on-device stage; the overwhelming majority resolves instantly and offline. Escalation to Deep Analysis is a deliberate *human-in-the-loop* step: the user triggers it whenever higher assurance is wanted. The cloud stage then examines every ingredient separately (including cross-allergen and contamination checks) and returns a source-backed, per-ingredient rationale. What remains is a thin stream of divergences whose probability is the product of both stages' miss rates (Eq. 2) and is continuously monitored. Band widths are illustrative, not to scale.

[18], applied to a safety-critical dietary setting — with a human, not a heuristic, deciding when the second, independent opinion is worth obtaining.

If the two stages fail *independently* with per-stage false-negative rates p_1 and p_2 on the escalated cases, the probability that a genuine gluten case slips through *both* is their product:

$$\Pr(\text{FN through cascade}) \;=\; p_1 \cdot p_2 \;\ll\; \min(p_1, p_2). \tag{2}$$

Equation (2) is a *theoretical intuition under an independence idealisation*, not an operational safety calculation: the two stages read the same text and may share failure modes, and positively correlated errors push the true rate *above* $p_1 p_2$ (though never above $\min(p_1, p_2)$). That is precisely why we *measure* inter-stage agreement rather than assume it (Section 7).

4 Training Methodology

This section describes the training approach at the method level. It deliberately omits concrete hyperparameter values (learning rate, epochs, dropout, weight decay, batch size), the search-space bounds and the OCR noise profile; these are part of the moat, not the contribution.

4.1 Data at a glance

The v4 training snapshot comprises **≈40,000 labelled German ingredient texts**, split **70/15/15** (train / validation / test hold-out) with **stratification** to preserve the class distribution in every partition. The class balance is **moderately imbalanced** in favour of gluten-free lines — enough to matter, which is why the class-weighted loss of Section 4.3 is used rather than ignoring or re-sampling the imbalance. The disjoint gold set used for the headline results (Section 5) sits entirely *outside* this snapshot. Dataset sources, curation pipeline and label provenance are withheld; the scale is disclosed, the recipe is not.

4.2 Fine-tuning setup

The Stage-1 model is a **BERT encoder** [5] with a sequence-classification head, fine-tuned for the binary gluten task. Optimisation uses **AdamW** with decoupled weight decay [6], a linear/cosine learning-rate schedule with warm-up, and **mixed-precision (fp16)** for throughput. The specific checkpoint, schedule constants and regularisation values are withheld.

4.3 Class-weighted objective

The data is class-imbalanced (far more gluten-free than gluten-bearing lines). Rather than *down-weight* the minority gluten class — which would be actively unsafe given the asymmetric cost structure of Section 2 — we use a **class-weighted cross-entropy** that *up-weights* the safety-critical class. With per-class weights w_c set by the balanced heuristic

$$w_c \;=\; \frac{N}{K\, n_c}\,, \tag{3}$$

where N is the number of training examples, K the number of classes and n_c the count of class c, the training loss becomes

$$\mathcal{L} \;=\; -\frac{1}{N}\sum_{i=1}^{N} w_{y_i}\Big[\, y_i \log p_i + (1-y_i)\log(1-p_i)\,\Big]. \tag{4}$$

Because $n_{\text{gluten}} < n_{\text{gluten-free}}$, Eq. (3) assigns $w_{\text{gluten}} > w_{\text{gluten-free}}$: errors on the dangerous class are penalised more heavily, aligning the training signal with the safety-oriented error budget of Section 2.

4.4 On-the-fly OCR-noise augmentation

The real input is a photographed label, so the model is trained against that reality. A **dynamic collator** applies OCR-style corruption *per batch* at training time, so the model sees fresh noise every epoch rather than a fixed corrupted copy. Crucially, the *same* noise model is used in the noisy evaluation split (Section 5), giving **train/serve consistency**: the distribution the model is selected on matches the distribution it is trained against. The exact corruption profile is not disclosed.

4.5 Model selection at the safety operating point

This is the methodological centrepiece. Standard practice selects the best epoch and configuration at a naïve threshold ($t = 0.5$, $\arg\max$). We do not: because the deployed system operates at a *low, cost-sensitive* threshold (Eq. (1)), we select the model at that same operating regime. Concretely, for each candidate configuration θ and epoch e we *sweep* the decision threshold on the OCR-noised validation split and take the best attainable F_2; the selected model maximises this quantity:

$$(\theta^\star, e^\star) \;=\; \arg\max_{\theta,\,e}\; \max_{t\in[0,1]}\; F_2^{\text{noisy}}\big(\theta,\, e,\, t\big). \tag{5}$$

The inner maximisation over t (the "sweep") means we never reward a model that only looks good at 0.5; we reward the model that is best *at its real safety operating point* under realistic input noise. This directly couples model selection to the cost asymmetry of Section 2 — a single, consistent safety objective from loss to selection to deployment.

5 Evaluation Methodology

Credibility in a safety domain rests on the evaluation, not the headline. Ours is built to resist the two failure modes that most often flatter a model: **data leakage** and **easy test sets**.

Gold set. We evaluate on a hand-curated gold standard plus a set of adversarial "traps" (must-catch cases). The gold sample is **disjoint** from the training snapshot under a near-duplicate audit (leakage control [16]; see Section 7.3 for a fully reproducible re-audit), and every label is **human-verified**: each ingredient list was read in full by a reviewer, then checked adversarially — not merely validated by the same rules the model might exploit.

Conditions. We test on clean text and on **OCR-noised** text (augmentation), because the real input is a photographed label, not a clean string.

Metrics. Because the cost structure is asymmetric, accuracy is the wrong headline. We report:

- F_2 — the recall-weighted member of the F_β family [11], appropriate when recall matters more than precision:

$$F_\beta = (1+\beta^2)\,\frac{\text{precision}\cdot\text{recall}}{\beta^2\,\text{precision}+\text{recall}}, \qquad \beta = 2. \tag{6}$$

- **Matthews Correlation Coefficient (MCC)** — robust on imbalanced data and more informative than F_1 or accuracy [12]:

$$\text{MCC} = \frac{\text{TP}\cdot\text{TN} - \text{FP}\cdot\text{FN}}{\sqrt{(\text{TP}+\text{FP})(\text{TP}+\text{FN})(\text{TN}+\text{FP})(\text{TN}+\text{FN})}}. \tag{7}$$

- **Balanced Accuracy**, the mean of per-class recalls: $\frac{1}{2}\big(\frac{\text{TP}}{\text{TP}+\text{FN}} + \frac{\text{TN}}{\text{TN}+\text{FP}}\big)$.
- **Binary agreement with Deep Analysis**, an operational quality signal.

Statistical rigour. Sample sizes here are small but clean, so we report **Wilson score intervals** [13] rather than point estimates alone; paired model comparisons use **McNemar's test** on the discordant predictions [14, 15]. The Wilson interval for an observed rate $\hat{p} = k/n$, with z the standard-normal quantile ($z \approx 1.96$ for 95 % confidence), is

$$\frac{\hat{p} + \dfrac{z^2}{2n} \;\pm\; z\sqrt{\dfrac{\hat{p}(1-\hat{p})}{n} + \dfrac{z^2}{4n^2}}}{1 + \dfrac{z^2}{n}}, \tag{8}$$

which, unlike the normal approximation, stays inside $[0,1]$ and behaves well at the boundary $\hat{p} \to 1$ — exactly the regime of a 100 %-recall safety classifier. For the paired comparison, with b and c the discordant counts (model A wrong / model B right, and vice versa), the *exact* McNemar test conditions on $n_d = b + c$: under the null hypothesis both directions are equally likely, $b \sim \text{Bin}(n_d, \frac{1}{2})$, and the two-sided p-value is

$$p \;=\; \min\left(1,\; 2\sum_{k=\max(b,c)}^{n_d} \binom{n_d}{k} \big(\tfrac{1}{2}\big)^{n_d}\right), \tag{9}$$

appropriate for the small discordant counts encountered here (the χ^2 approximation would not be).

Gold-set sizes. Disjoint held-out: **180 lines** (48 "contains gluten" / 132 "gluten-free"), verified disjoint from training via near-duplicate audit. Adversarial traps: **56** (26 must-catch / 30 gluten-free), zero leakage.

6 Model Evolution: v1 → v4

The central engineering story is not a bigger model — it is a better *dataset*. Each generation moved the data axis, and the gains followed.

This reflects a now well-documented pattern in applied ML: for deployed systems, disciplined *data work* tends to dominate model-centric tuning [17]. Concrete dataset sources, sizes and composition are withheld; they are the core of the moat.

Generation	Focus of iteration	Core improvement
v1 (March 2026)	First model	Basic capability; baseline established.
v2 (May 2026)	Calibration / training	More stable verdicts through improved calibration.
v3 (May 2026)	Operating-point calibration	Reference generation; baseline for the paired comparison in Section 7.
v4 (July 2026)	**Data-centric retrain**	Class rebalancing via an expanded catalogue; swept-F_2 objective; threshold calibration; disclaimer handling.

Table 1: Model generations. The decisive lever between v3 and v4 was the data axis — catalogue expansion and clean labels — not model size or hyperparameter search.

6.1 Hyperparameter optimisation: evidence for "data > HPO"

The claim that data was the lever is not rhetorical — it is what the tuning *showed*. We ran **Bayesian hyperparameter optimisation** [7, 8] (deliberately sequential, so each trial informs the next) over a bounded search space, with the objective set to the same **swept-F_2 on the OCR-noised validation split** used for model selection (Eq. (5)). Formally, the surrogate-guided search seeks

$$\theta^\star \;=\; \arg\max_{\theta\in\Theta} \Big[\max_{t\in[0,1]} F_2^{\text{noisy}}(\theta, t) \Big], \tag{10}$$

over a restricted hyperparameter domain Θ. Across the trial budget the objective **plateaued**: additional search bought little. That plateau is the empirical signal that the model-centric axis was largely exhausted and that the remaining gains lived on the *data* axis — which is exactly where v4 found them. We report the fact of the sweep and that it saturated; the number of trials and the winning configuration are not disclosed.

7 Results

All figures below are on the leakage-free disjoint held-out set at a common operating point, with paired comparison between v3 (Run C) and v4.

Metric	v3	v4
Observed recall (48/48 gluten caught)	100 %	**100 %**
False-alarm rate (safe wrongly flagged)	13.6 %	**6.8 %**
F_2 (recall-weighted, $\times 100$)	93.0	**96.4**
MCC ($\in [-1, 1]$)	0.793	**0.886**
Balanced Accuracy	93.2 %	**96.6 %**

Table 2: v3 → v4, disjoint held-out, common operating point. Adversarial traps were unchanged (no regression on must-catch cases).

Headline result

The false-alarm rate is **roughly halved** (13.6 % → 6.8 %), with **no false negatives observed in either generation** (observed recall 100 % on this sample), higher F_2, MCC and balanced accuracy — and no regression on adversarial must-catch traps. These are controlled gold-set results, not yet a large-scale real-world validation (Section 9).

7.1 Confusion matrices

For full transparency, Table 3 gives the complete confusion matrices on the disjoint gold set (48 gluten-bearing / 132 gluten-free lines) at the common operating point.

		v3 predicted		v4 predicted	
		gluten	gluten-free	gluten	gluten-free
actual	gluten (48)	48	**0**	48	**0**
	gluten-free (132)	18	114	**9**	**123**

Table 3: Confusion matrices, v3 vs. v4, disjoint gold set at the common operating point. No false negatives were observed in either generation; v4 halves the false positives (18 → 9 of 132).

7.2 Uncertainty quantification

With $n = 132$ gluten-free lines, the Wilson 95 % interval for the false-alarm rate narrows from [8.8, 20.5]% (v3) to [3.6, 12.5]% (v4). Recall of 100 % on $n = 48$ gluten-bearing lines corresponds to a Wilson lower bound of 92.6 % — i.e. even under conservative interval estimation, recall remains high. We report these intervals explicitly rather than lean on point estimates; the sample is small, and honest error bars are part of the claim.

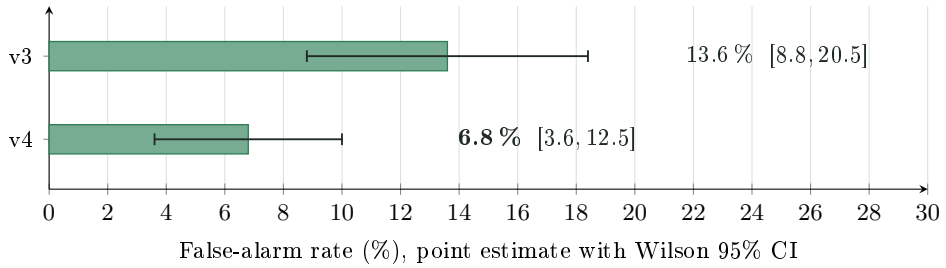

Figure 3: False-alarm rate with Wilson 95 % confidence intervals ($n = 132$ gluten-free lines). v4 approximately halves the point estimate, and the entire interval shifts downward. Recall is 100 % in both generations.

7.3 Statistical significance and calibration

Because the gold set is small, the paired v3 ↔ v4 comparison [14, 15] was re-run on the larger internal held-out split (428 lines) — with one additional safeguard: a **near-duplicate audit** against the full v4 training snapshot (≈40,000 lines). Only lines that are **genuinely disjoint** from training under a transparent, reproducible normalisation enter the paired analysis: **209 lines (152 gluten-free / 57 gluten-bearing)**. On this leakage-free subset, **v4 substantially reduces the false-alarm rate at every operating point examined, and more than halves it at the more sensitive screening point** (Table 4).

Across the sensitive operating range the improvement is **highly significant** ($p < 0.001$). At the strict deployment point the paired test is **underpowered** — only 13 discordant pairs ($p \approx 0.09$) — so significance there cannot yet be claimed; the direction, however, is consistent

Operating point	v3 FP-rate (Wilson 95 %)	v4 FP-rate (Wilson 95 %)	b / c	exact McNemar
deployed operating point	12.5 % [8.2, 18.7]	**7.9 %** [4.6, 13.3]	10 / 3	$p \approx 0.09$ (underpowered)
more sensitive screening point	37.5 % [30.2, 45.4]	**14.5 %** [9.8, 20.9]	40 / 5	$p < 0.0001$

Table 4: Paired v3 vs. v4 comparison on the leakage-free disjoint subset (152 gluten-free lines). b = v3 wrong / v4 right, c = v3 right / v4 wrong. Concrete threshold values are withheld.

throughout. The definitive, high-powered test will follow with post-cutoff production scans. On the 57 disjoint gluten-bearing lines, **both generations produced zero false negatives**. The reproducible re-audit also cross-validates the headline result: it yields a larger leakage-free evaluation subset from the internal held-out split (209 lines) than the original hand-curated gold set used for the headline result (180 lines), with closely matching false-alarm rates (12.5 % → 7.9 % vs. 13.6 % → 6.8 %) — two distinct evaluation bases, one conclusion. The result is robust to the audit definition and, if anything, conservative.

The model's probabilities are also usefully *calibrated*, which matters for a system whose safety behaviour is defined by a probability threshold. With $\hat{p}_i$ the predicted gluten probability and $y_i \in \{0, 1\}$ the true label, we report the Brier score [21] and the expected calibration error (ECE) [22, 23],

$$\mathrm{BS} = \frac{1}{N}\sum_{i=1}^{N}(\hat{p}_i - y_i)^2, \qquad \mathrm{ECE} = \sum_{m=1}^{M}\frac{|B_m|}{N}\,\big|\,\mathrm{acc}(B_m) - \mathrm{conf}(B_m)\,\big|, \tag{11}$$

where B_m are M=10 equal-width confidence bins. On a larger internal held-out split (428 lines, clean text), v4 attains BS ≈ 0.027 and ECE ≈ 0.035. Two caveats apply: these figures are measured on curated evaluation data, not the production distribution, and the full 428-line split partially overlaps the training snapshot under the near-duplicate audit above — leakage-free calibration figures will follow with post-release data.

Finally, every evaluation report includes a **50-point threshold sweep** recording false positives, false negatives, recall and false-positive rate across the candidate operating range. The sweep serves two purposes: it drives model selection (Eq. (5)), and it verifies that the chosen operating point does not sit on a knife's edge — performance degrades smoothly, not abruptly, in its neighbourhood. The concrete operating point remains withheld.

7.4 Quantisation and on-device performance

We apply **dynamic post-training quantisation** (INT8) [9, 10]: each weight tensor is stored as 8-bit integers under an affine mapping

$$x \;\approx\; s \cdot q, \qquad q \in \{-127, \ldots, 127\}, \qquad s = \frac{\max_i |x_i|}{127}, \tag{12}$$

with per-tensor scale s, while activations are quantised *dynamically* at runtime with scales computed on the fly — **no re-training and no calibration dataset** required. This compresses the model from ≈ 419 MB (FP32) to ≈ 106 MB (INT8), roughly 4× smaller — comfortably app-sized for fully offline use on the device. Crucially, in this validation setup the compression **did not increase the observed number of false negatives**. Measured on the large noisy *validation* split (a broader and harder measurement than the disjoint gold set of Table 2), the INT8 model produced **45 false negatives vs. 46 for FP32**, at recall ≈ 0.987 — i.e. no observed safety cost from quantisation in our measurements.

7.5 Operational signal

In live operation, the binary agreement between the on-device classifier and Deep Analysis currently sits at ≈**93** %. This is a continuous quality signal built into the architecture (Section 3), not a one-off benchmark; sustained divergence — especially safety-critical divergence — is prioritised for review. v4 live-data figures will follow from post-release scans.

8 Privacy

Quick-Scan inference is **entirely on-device**: the ingredient text does not leave the phone to obtain the fast verdict. When the user triggers Deep Analysis, **data minimisation** still applies: only the ingredient text and the dietary parameters needed for the analysis are transmitted — **no identifying user data** is processed in the cloud. This is a privacy-by-design choice consistent with GDPR data-minimisation [26, 25] — health-adjacent inference should default to the edge, and cloud processing should be the deliberate, user-initiated exception rather than the rule.

9 Limitations and Outlook

Stating limits plainly earns trust with a technical audience.

- **Deliberately small, hand-curated gold set.** The disjoint evaluation set (48 gluten / 132 gluten-free) is not a bulk sample but a *hand-curated gold standard*: every single ingredient list was read in full and labelled by a human reviewer. Human verification is what makes the labels trustworthy — and what bounds the set's size. We therefore report Wilson intervals and are extending validation to a larger post-release scan set.
- **OCR robustness at the tail.** Very rare malt/grain tokens under heavy OCR noise remain the hardest residual case; this is where the deterministic safety net and Deep Analysis matter most.
- **Translation chain.** For non-German labels, the on-device translation step adds a further potential error source before classification. Its robustness is part of the ongoing validation; residual risk is mitigated by the conservative operating point and by user-triggered Deep Analysis, but it is a real, distinct failure path and is treated as such.
- **Correlated failures.** The cascade bound of Eq. (2) assumes independence; real errors are partly correlated, which is why inter-stage agreement is monitored rather than assumed.

9.1 Failure modes and mitigations

Defense-in-depth is only credible if the failure paths are named. Table 5 summarises the main modes and which layer of the system addresses each.

A compact model card summarising intended use, scope and known limits is provided in Appendix A.

9.2 Evidence status

To keep validated evidence and product intent clearly separated: **validated so far** — the disjoint, hand-curated gold set and the adversarial trap set (controlled evidence, small n, Wilson intervals reported); **currently monitored** — live binary agreement between the on-device classifier and Deep Analysis (≈93 %); **planned** — a larger post-release, real-world validation on production scans. Claims in this paper should be read against this hierarchy.

Outlook. Continued data-centric iteration, broader robustness to OCR noise, and validation against live post-release scans. The guiding principle is unchanged: the *evaluation objective* is zero observed false negatives in every controlled evaluation, and false alarms are reduced through better data rather than through relaxed safety margins.

Failure mode	Primary risk	Mitigation layer(s)
OCR misreads a gluten grain token	false negative	OCR-noise training; deterministic safety net; conservative threshold
Translation loses ingredient meaning	false negative	conservative operating point; user-triggered Deep Analysis
Ambiguous or truncated label entry	false positive / unclear	three-state UX ("unclear" instead of a forced verdict)
Novel ingredient / rare token	uncertainty	escalation to Deep Analysis; agreement monitoring
Correlated errors across both stages	residual false negative	continuous agreement monitoring; data-centric retraining

Table 5: Main failure modes and the system layer that addresses each. No single layer is assumed to be perfect; the design goal is that every plausible path to a dangerous error crosses at least one independent mitigation.

References

[1] Codex Alimentarius Commission. *Standard for Foods for Special Dietary Use for Persons Intolerant to Gluten (CODEX STAN 118-1979)*. FAO/WHO, rev. 2008.

[2] Catassi C., Fabiani E., Iacono G., et al. A prospective, double-blind, placebo-controlled trial to establish a safe gluten threshold for patients with coeliac disease. *Am. J. Clin. Nutr.*, 85(1):160–166, 2007.

[3] Ludvigsson J.F., Leffler D.A., Bai J.C., et al. The Oslo definitions for coeliac disease and related terms. *Gut*, 62(1):43–52, 2013.

[4] Elkan C. The foundations of cost-sensitive learning. *Proc. IJCAI*, 2001.

[5] Devlin J., Chang M.-W., Lee K., Toutanova K. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. *Proc. NAACL-HLT*, 2019.

[6] Loshchilov I., Hutter F. Decoupled Weight Decay Regularization. *Proc. ICLR*, 2019.

[7] Snoek J., Larochelle H., Adams R.P. Practical Bayesian Optimization of Machine Learning Algorithms. *Proc. NeurIPS*, 2012.

[8] Bergstra J., Bengio Y. Random Search for Hyper-Parameter Optimization. *J. Mach. Learn. Res.*, 13:281–305, 2012.

[9] Jacob B., Kligys S., Chen B., et al. Quantization and Training of Neural Networks for Efficient Integer-Arithmetic-Only Inference. *Proc. CVPR*, 2018.

[10] Zafrir O., Boudoukh G., Izsak P., Wasserblat M. Q8BERT: Quantized 8Bit BERT. *NeurIPS EMC² Workshop*, 2019.

[11] van Rijsbergen C.J. *Information Retrieval*, 2nd ed. Butterworths, 1979.

[12] Chicco D., Jurman G. The advantages of the Matthews correlation coefficient (MCC) over F1 score and accuracy in binary classification evaluation. *BMC Genomics*, 21:6, 2020.

[13] Wilson E.B. Probable inference, the law of succession, and statistical inference. *J. Am. Stat. Assoc.*, 22(158):209–212, 1927.

[14] McNemar Q. Note on the sampling error of the difference between correlated proportions or percentages. *Psychometrika*, 12(2):153–157, 1947.

[15] Dietterich T.G. Approximate statistical tests for comparing supervised classification learning algorithms. *Neural Computation*, 10(7):1895–1923, 1998.

[16] Kaufman S., Rosset S., Perlich C., Stitelman O. Leakage in data mining: Formulation, detection, and avoidance. *ACM TKDD*, 6(4):1–21, 2012.

[17] Sambasivan N., Kapania S., Highfill H., et al. "Everyone wants to do the model work, not the data work": Data Cascades in High-Stakes AI. *Proc. CHI*, 2021.

[18] Viola P., Jones M. Rapid Object Detection using a Boosted Cascade of Simple Features. *Proc. CVPR*, 2001.

[19] Amershi S., Cakmak M., Knox W.B., Kulesza T. Power to the People: The Role of Humans in Interactive Machine Learning. *AI Magazine*, 35(4):105–120, 2014.

[20] Mitchell M., Wu S., Zaldivar A., et al. Model Cards for Model Reporting. *Proc. ACM FAT**, 220–229, 2019.

[21] Brier G.W. Verification of forecasts expressed in terms of probability. *Monthly Weather Review*, 78(1):1–3, 1950.

[22] Naeini M.P., Cooper G.F., Hauskrecht M. Obtaining Well Calibrated Probabilities Using Bayesian Binning. *Proc. AAAI*, 2015.

[23] Guo C., Pleiss G., Sun Y., Weinberger K.Q. On Calibration of Modern Neural Networks. *Proc. ICML*, 2017.

[24] Deutsche Zöliakie-Gesellschaft e.V. (DZG). Guidance on the gluten-free diet and precautionary "may contain traces" labelling. `https://www.dzg-online.de`.

[25] Dhar S., Guo J., Liu J., et al. A Survey of On-Device Machine Learning: An Algorithms and Learning Theory Perspective. *ACM Trans. IoT*, 2(3):1–49, 2021.

[26] European Parliament and Council. Regulation (EU) 2016/679 (General Data Protection Regulation), Art. 5 (data minimisation) and Art. 25 (data protection by design). 2016.

Appendix A: Model Card (v4)

Following the model-reporting practice of Mitchell et al. [20], Table 6 condenses the key facts — disclosed scale, no secret details.

Model	Binary gluten classifier; BERT-family encoder, INT8 (dynamic PTQ), ≈106 MB; version v4 (July 2026).
Intended use	Informational classification of *declared ingredients* in photographed ingredient lists; German text (other languages via on-device translation); three-state UX (green / unclear / red).
Out of scope	Trace / ppm / contamination assessment; "may contain traces" statements (surfaced as information, not classified as gluten, per DZG guidance); medical diagnosis; substitute for reading the label.
Training data	≈40,000 labelled German ingredient texts (v4 snapshot); 70/15/15 stratified split; sources and curation withheld.
Evaluation	Disjoint hand-curated gold set (180 lines) + 56 adversarial traps; leakage-free disjoint subset (209 lines, near-duplicate audit) for paired significance testing; calibration on a 428-line held-out split (partial training overlap, see Section 7.3).
Known limits	Rare grain tokens under heavy OCR noise; translation chain for non-German labels; small gold set (Wilson intervals reported).
Monitoring	Live classifier ↔ Deep-Analysis agreement (≈93 %); divergences reviewed; post-release validation planned.

Table 6: Model card for the v4 on-device classifier.

Appendix B: Derivation of the Safety Operating Point

Setup. Let $p = \Pr(y = 1 \mid x)$ be the (true) posterior probability that the ingredient text x declares gluten. Correct decisions incur no cost; a false positive costs $c_{\mathrm{FP}} > 0$ and a false negative costs $c_{\mathrm{FN}} > 0$. The expected conditional risk of each action is

$$R(\text{predict gluten-free} \mid x) \;=\; p\, c_{\mathrm{FN}}, \qquad R(\text{predict contains gluten} \mid x) \;=\; (1-p)\, c_{\mathrm{FP}}.$$

Derivation. The Bayes-optimal rule predicts "contains gluten" whenever doing so has no greater expected cost:

$$(1-p)\, c_{\mathrm{FP}} \;\le\; p\, c_{\mathrm{FN}} \iff c_{\mathrm{FP}} \;\le\; p\,(c_{\mathrm{FP}} + c_{\mathrm{FN}}) \iff p \;\ge\; \frac{c_{\mathrm{FP}}}{c_{\mathrm{FP}} + c_{\mathrm{FN}}} \;=:\; t^{\star},$$

which is Eq. (1) [4]. Writing the cost ratio as $r := c_{\mathrm{FN}}/c_{\mathrm{FP}}$ gives $t^{\star} = 1/(1+r)$: strictly decreasing in r, equal to 0.5 only in the symmetric case $r = 1$, and $t^{\star} \to 0$ as $r \to \infty$. The safety-first stance of Section 2 is therefore not a heuristic but the exact Bayes decision under the asymmetric cost structure of coeliac risk: the more a false negative outweighs a false positive, the lower the evidence bar for flagging.

Corollary 1 (why calibration matters). The rule above is optimal only if the score used at inference is the true posterior. If the deployed model outputs $\hat{p}$ with calibration error $\varepsilon(x) = \hat{p}(x) - p(x)$, then thresholding $\hat{p}$ at $t^{\star}$ is equivalent to thresholding the true posterior at $t^{\star} - \varepsilon(x)$ —

miscalibration silently *moves* the operating point. This is precisely why the calibration figures of Eq. (11) are monitored: in a threshold-based safety design, the safety behaviour of the threshold rule depends directly on calibration.

Corollary 2 (cascade bound). Let A_1 be the event that Stage 1 misses a genuine gluten case, and A_2 the event that Stage 2, given escalation, also misses it. A cascade false negative requires both: $\Pr(A_1 \cap A_2) = \Pr(A_1)\Pr(A_2 \mid A_1) = p_1 \Pr(A_2 \mid A_1)$. Under independence, $\Pr(A_2 \mid A_1) = p_2$, giving Eq. (2); in general $\Pr(A_2 \mid A_1) \leq 1$, so the cascade rate never exceeds p_1. If, moreover, $p_2 := \Pr(A_2)$ is defined on the *same* case distribution, the cascade rate is bounded by $\min(p_1, p_2)$; if p_2 is measured only on escalated cases, the bound must be read relative to that escalation distribution. Positive correlation raises the rate towards this ceiling, which is why inter-stage agreement is measured (Section 7) rather than assumed.

