Bootstrapping Text Anomaly Detection with LLM-Generated Weak Supervision

Fabio Masaracchia Maia[1], **Anna Helena Reali Costa**[1]

[1]Escola Politécnica, Universidade de São Paulo, São Paulo, Brazil

fabio.masaracchia@usp.br, anna.reali@usp.br

Abstract. *Text anomaly detection is challenging because anomalous instances often share vocabulary and surface form with normal data, making them hard to distinguish without semantic understanding. Semi-supervised methods can significantly outperform unsupervised baselines, but rely on labeled anomalies that are rarely available in practice. LLMs encode rich semantic knowledge that can approximate human judgments, yet using them directly as detectors is costly at inference time and sensitive to prompt design, while generating synthetic outliers risks distribution mismatch with real anomalies. We propose a different strategy: treating a compact, locally deployed LLM as a noisy annotator over real data. The LLM scores a small subset of unlabeled documents once at training time, producing weak labels that refine a sentence encoder via contrastive fine-tuning and train a lightweight downstream detector — without cloud APIs, human annotation, or curated anomaly datasets. Across four datasets spanning two languages and two tasks, the approach recovers up to 79% of the gap to oracle upper bounds while using 14–91× fewer labeled anomalies. We further identify three failure modes that explain when LLM-generated weak supervision succeeds or breaks down.*

1. Introduction

Semi-supervised anomaly detection can dramatically outperform unsupervised methods — but only when labeled anomalies are available [Pang et al. 2021, Xu et al. 2023b]. In text domains, this creates a practical bottleneck: the labels that would most improve detection are precisely the hardest to obtain, because anomalies are rare, diverse, and may share vocabulary and register with normal instances (e.g. hate speech vs. neutral posts on the same platform).

Large Language Models (LLMs) offer a potential solution, as their broad pretraining enables them to approximate human judgments on many NLP annotation tasks [Wang et al. 2021, He et al. 2024]. Prior work in anomaly and out-of-distribution detection has largely explored this potential in two ways. *LLM-as-detector* approaches query the model directly for anomaly judgments, but their predictions can be sensitive to prompt design and often lack calibration across domains [Liu et al. 2024, Yang et al. 2025]. *LLM-as-generator* approaches produce synthetic outlier examples for training, but risk distribution mismatch between generated and real anomalies [Hendrycks et al. 2019, Xu et al. 2023a]. However, neither approach directly yields a lightweight detector that is both stable across domains and inexpensive at inference time.

We take a different approach: instead of using the LLM for detection or data generation, we use it as a *noisy annotator* over real data. A compact, locally deployed model (Qwen2.5-Instruct [Qwen Team 2025], 7B/14B, 4-bit quantization) scores a small subset of *real* unlabeled documents once at training time, producing weak anomaly labels. These labels are used to refine the sentence encoder via contrastive fine-tuning and to train a lightweight downstream detector — with no cloud APIs, no human annotators, and no annotation budget. By annotating real data rather than generating synthetic examples, and by decoupling annotation from inference, the approach avoids both the instability of LLM-based detection and the distribution mismatch of synthetic generation.

Our contributions:

- An LLM-in-the-loop framework that transfers weak semantic supervision into a lightweight text anomaly detector, eliminating the need for human annotation.
- Empirical evidence that the approach closes up to 79% of the gap to oracle upper bounds, using 14–91$\times$ fewer anomaly labels, across two tasks and two languages.
- Identification of three failure modes and the finding that the pipeline tolerates high volumes of false positives but fails under systematic label inversion — a distinction between quantitative and directional annotation noise with practical implications for deployment.

2. Related Work

2.1. Text Anomaly Detection

Anomaly detection is well established for tabular and image data [Pang et al. 2021], but the text domain introduces distinct challenges: documents are high-dimensional, semantically rich, and often require domain-specific knowledge to separate anomalous from normal content [Chandola et al. 2009, Boutalbi et al. 2023]. Classical one-class methods — OC-SVM [Schölkopf et al. 2001], Isolation Forest [Liu et al. 2008] — applied to sparse representations struggle with the semantic similarity between classes. Neural methods learn denser representations: CVDD [Ruff et al. 2019] adapts one-class deep learning to text, and DATE [Manolache et al. 2021] uses self-supervised corruption for fully unsupervised training.

A key finding from Xu et al. [Xu et al. 2023b], who benchmark 22 algorithms across 17 corpora, is that *encoder quality matters more than the choice of AD algorithm*. This motivates our design: rather than proposing a new detection architecture, we focus on improving the encoder via contrastive fine-tuning before detection. Das et al. [Das et al. 2023] confirm that even a handful of labeled anomalies yields large gains, and Ait-Saada & Nadif [Ait-Saada and Nadif 2023] address multi-topic short-text AD, a setting close to ours. Both assume clean human labels are available. A recent comparative study [Maia and Costa 2025] benchmarks semi-supervised text AD across multiple languages, confirming that the label scarcity bottleneck persists across linguistic settings. Our work follows this semi-supervised paradigm but replaces the human annotation assumption with LLM-generated weak labels.

2.2. LLMs for Anomaly and OOD Detection

Recent work explores LLMs as anomaly or out-of-distribution (OOD) detectors. Although the two tasks share the goal of flagging deviations from normality, OOD detection

typically assumes that anomalies form a separable distribution, whereas AD must handle anomalies that overlap semantically with normal data [Chandola et al. 2009] — as in our hate speech setting. Xu & Ding [Xu and Ding 2025] categorise approaches into two main roles: LLMs as detectors (prompting or embedding-based) and LLMs as generators (producing synthetic data or auxiliary labels).

In detection-based approaches, LLMs are queried directly for anomaly judgments. Liu et al. [Liu et al. 2024] and Yang et al. [Yang et al. 2025] find competitive but inconsistent performance: LLM effectiveness is highly task-dependent, with failure modes on fine-grained distinctions and sensitivity to prompt design. This limits their reliability as standalone detectors, especially where stable, calibrated decision boundaries are needed. In generation-based approaches, LLMs enrich training data rather than serving as detectors. Outlier exposure methods introduce proxy anomalies [Hendrycks et al. 2019], and CoNAL [Xu et al. 2023a] generates novel-class examples via prompting for contrastive OOD training. However, these strategies rely on *synthetic* samples whose distribution may not reflect real anomalies, risking distribution mismatch.

2.3. Weak Supervision and Few-Shot Fine-Tuning

Weak supervision refers to learning from imperfect supervisory signals, such as heuristic rules, noisy annotations, or weakly informative labels, instead of fully reliable human-curated ground truth [Ratner et al. 2020, Zhou 2018].Recent work has explored LLM annotation as an alternative to crowdsourcing [Wang et al. 2021, He et al. 2024]. However, these approaches assume *well-defined label spaces* with explicit class boundaries (e.g. sentiment polarity, named topics). Anomaly detection requires the annotator to infer an *implicit normality boundary* — what counts as "normal" for a specific corpus — without labeled examples and without a closed set of target classes. This makes LLM annotation both harder (the prompt must convey a distributional concept) and noisier (the LLM has no ground-truth anchors to calibrate against).

Since encoder quality is the dominant factor in text AD performance [Xu et al. 2023b], refining representations with the few available labels is a natural strategy. Contrastive fine-tuning reshapes the embedding space by pulling instances with the same label closer together while pushing instances with different labels farther apart [Khosla et al. 2020], effectively sharpening the normal–anomaly boundary before any detector is trained. Among few-shot approaches, SetFit [Tunstall et al. 2022] provides a practical implementation of this idea by fine-tuning a sentence transformer from few labeled pairs, achieving strong performance with as few as 8 examples per class.

3. Methodology

3.1. Problem Formulation and Pipeline

We frame the task as *semi-supervised anomaly detection*: a small set of labeled anomalies is available at training time, but the bulk of the data is unlabeled. An important subtlety is that "anomaly" can be defined both as statistically rare (*low-frequency*) and as content deviating from expectations (*semantic*). We construct the low-frequency anomaly by subsampling the minority class to $\sim 5\%$, but the LLM annotator operates semantically, judging content against natural-language criteria.

Let $\mathcal{D}$ be a dataset with a designated normal class. We construct X_{train} with contamination rate $\rho \approx 5\%$ and hold out X_{test} preserving the same proportions. No human-labeled anomalies are available.

Figure 1 illustrates the pipeline. All documents are encoded into dense vectors using the pre-trained multilingual sentence encoder *distiluse-base-multilingual-cased-v2* [Reimers and Gurevych 2019]. A candidate set $S \subset X_{\text{train}}$, $|S| = N \ll |X_{\text{train}}|$, is selected via one of two sampling strategies (Section 3.2). For each candidate $x_i \in S$, a structured prompt is constructed as:

$$P_i = T(x_i,\, d_{\text{task}},\, c_{\text{normal}},\, c_{\text{anomaly}}) \tag{1}$$

where $T(\cdot)$ is the prompt-building function, d_{task} is a task description, and $c_{\text{normal}}, c_{\text{anomaly}}$ define the normality boundary (all dataset-specific). The prompt is structured to provide a clear task definition, explicit criteria for both classes, and a request for a continuous score paired with a one-sentence explanation, reducing ambiguity and promoting consistent label generation across documents. The LLM returns a continuous anomaly score and a one-sentence explanation:

$$(s_i,\, r_i) = f_{\text{LLM}}(P_i), \quad s_i \in [0, 1] \tag{2}$$

The score is binarized at threshold $\tau = 0.5$ to obtain a weak label $\hat{y}_i = \mathbf{1}[s_i \geq \tau]$, yielding the weak anomaly set $\hat{A} = \{x_i \in S : \hat{y}_i = 1\}$. This threshold corresponds to the standard midpoint of the $[0, 1]$ score range; no tuning was performed, as our focus is on the weak supervision strategy rather than threshold calibration. This process can be noisy: the LLM correctly flags explicit anomalies but also produces false positives on ambiguous content.

We evaluate two open-weight models from the Qwen2.5-Instruct family [Qwen Team 2025]: the 7B (≈4 GB) and 14B (≈9 GB) variants, both in 4-bit GGUF quantization, executed locally via llama.cpp [Gerganov et al. 2023]. Qwen2.5-Instruct was selected as an open-weight instruction-tuned model that combines strong multilingual instruction-following capability with a favourable performance-to-size ratio, making it well-suited for local deployment and prompt-based weak-label generation without cloud infrastructure. 4-bit quantization substantially reduces GPU memory requirements while maintaining adequate inference quality for prompt-based annotation, enabling deployment on a single consumer-grade GPU. If $|\hat{A}| \geq 8$, the encoder is fine-tuned contrastively using SetFit [Tunstall et al. 2022]; otherwise the original embeddings are kept. All documents are then re-encoded, and a semi-supervised detector is trained on the full training set using $\hat{A}$ as labeled anomaly examples. Ground-truth labels are used *only* for evaluation.

3.2. Sampling Strategies

The choice of which documents to present to the LLM matters: in a corpus where $\sim 95\%$ of instances are normal, a naive sample will contain few or no anomalies. We evaluate two strategies: **Random** uniformly samples N instances from the training pool, reflecting the true $\sim 5\%$ anomaly rate; **Coverage** partitions the embedding space into N clusters via k-means++ and selects the centroid-nearest instance from each cluster [Sener and Savarese 2018], ensuring that peripheral regions — where anomalies are more likely to reside — are represented.

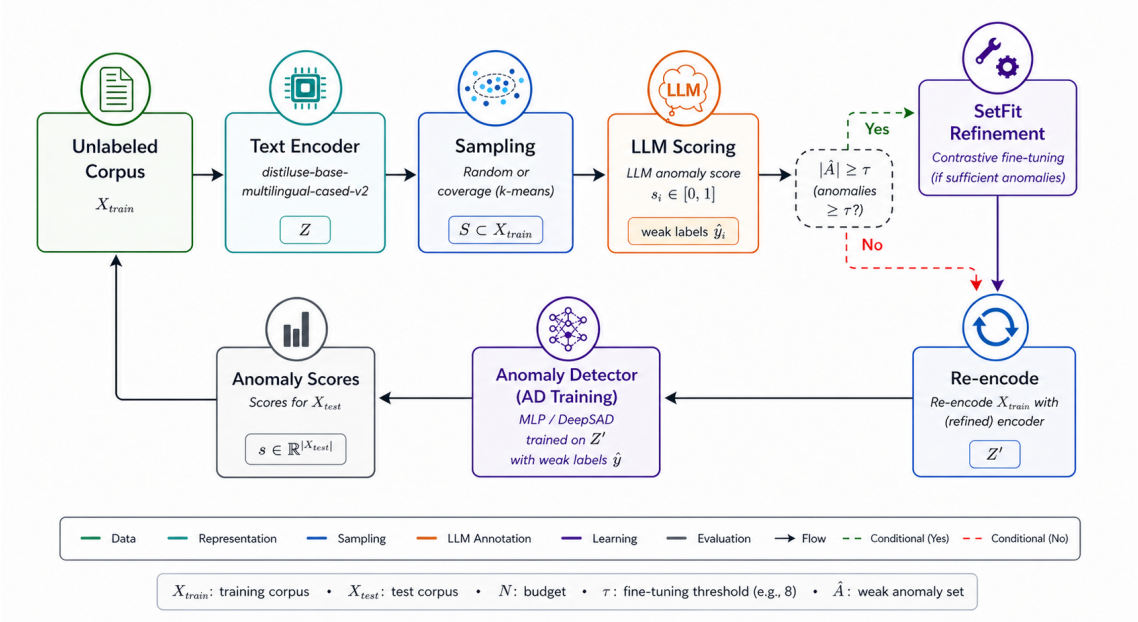

Figure 1. LLM supervision pipeline. Steps 1–3 encode the corpus, sample candidates, and query the LLM for weak anomaly labels ($\hat{A}$). Step 4 optionally refines the encoder via contrastive fine-tuning (skipped if $|\hat{A}| < 8$). Steps 5–6 re-encode and train the detector. Dashed arrow: $\hat{A}$ also provides labeled anomalies directly to the detector.

3.3. Anomaly Detection Models

We compare two representative semi-supervised AD paradigms whose training objectives respond differently to label noise.

DeepSAD [Ruff et al. 2020] learns a hypersphere in a latent space that encloses normal embeddings and pushes known anomalies away. Given encoder ϕ with centre $\mathbf{c}$, the loss for a sample (x_i, y_i) is:

$$\mathcal{L}_{\text{SAD}} = \frac{1}{n} \sum_{i=1}^{n} \begin{cases} \|\phi(x_i) - \mathbf{c}\|^2 & \text{if } y_i \in \{0, -1\} \\ \max(0,\ m - \|\phi(x_i) - \mathbf{c}\|^2) & \text{if } y_i = 1 \end{cases} \tag{3}$$

where $y_i \in \{0, 1, -1\}$ denotes normal, anomalous, or unlabeled, and m is a margin. A *single mislabeled* positive forces the encoder to push a normal instance away from the centre, directly distorting the boundary. We use the DeepOD [Xu 2022] implementation with a 3-layer MLP encoder (512→100→50→128) with ReLU activations, trained for 100 epochs.

MLP. A custom shallow binary classifier trained with standard cross-entropy:

$$\mathcal{L}_{\text{BCE}} = -\frac{1}{n} \sum_{i=1}^{n} \big[y_i \log \hat{y}_i + (1 - y_i) \log(1 - \hat{y}_i) \big] \tag{4}$$

Because the loss averages over all samples — including thousands of clean negatives — a handful of mislabeled positives has limited influence on the decision boundary. Architecture: two hidden layers (128, 64) with ReLU, sigmoid output, trained with Adam

(lr $= 10^{-3}$) for 50 epochs. Both models receive the same 512-dimensional distiluse embeddings as input.

Our proposed method, **MLP+SF**, combines the MLP with SetFit-refined embeddings. To isolate the contribution of each component, Section 4.2 introduces a 2×2 ablation that independently varies the classifier (MLP vs. DeepSAD) and the embedding (original vs. SetFit fine-tuned).

3.4. Datasets and Evaluation

We evaluate on four datasets spanning two tasks and two languages. Each is prepared by retaining all normal instances, merging remaining classes into an anomaly set subsampled to $\rho \approx 5\%$, and splitting 80/20 into train and test sets.

Table 1. Dataset overview. Sizes reflect the corpus after contamination adjustment and train/test split.

Dataset	Normal class	Anomaly class	Task	Lang	Size
20 Newsgroups [Lang 1995]	rec.sport.hockey posts	all other newsgroups	TC	en	1,048
WikiNews	Politics articles	Non-politics articles	TC	pt	6,120
Hate Speech Tweets [Sharma 2018]	Non-hate speech	Hate speech tweets	HS	en	31,205
HateBR [Leite et al. 2020]	Non-offensive	Offensive comments	HS	pt	3,675

TC = Topic Classification; HS = Hate Speech detection.

The four datasets form a 2×2 grid of task and language, following the experimental design of [Maia and Costa 2025], and vary considerably in detection difficulty: 20 Newsgroups is the easiest (distinctive hockey vocabulary), the hate speech datasets are the hardest (hateful and neutral text share register and style), and WikiNews sits in between. The WikiNews corpus was extracted from Portuguese Wikimedia dumps following the methodology of Garcia et al. [Garcia et al. 2024], with articles categorised by topic section (Politics as normal class, all other sections as anomalies).

Performance is measured by ROC-AUC on the held-out test set, averaged over 3 seeds $\times$ 2 sampling strategies at $N = 200$ (6 runs per configuration). We compare against seven unsupervised detectors and the best semi-supervised model with ground-truth labels ($\sim 5\%$ of training set).

4. Experiments and Results

All experiments ran on an NVIDIA Tesla T4 GPU (Google Colab). LLM annotation used Qwen2.5-Instruct [Qwen Team 2025] 7B and 14B in 4-bit quantization via llama.cpp [Gerganov et al. 2023] at zero API cost; labels were persisted so that all ablations consume identical annotations. We evaluated at $N = 200$, the smallest budget that consistently produced enough weak positives to activate SetFit ($|\hat{A}| \geq 8$) across datasets; both sampling strategies yielded statistically indistinguishable results and are aggregated for variance reduction. [1]

[1]The source code, prompts, and experimental configuration are available at https://github.com/FabioMMaia/bootstrapping-text-anomaly-detection

4.1. RQ1: Do LLM Labels Improve Over Unsupervised Detection?

Table 2 presents seven unsupervised baselines, four LLM-supervised configurations, and the best semi-supervised model with ground-truth labels ($\sim 5\%$ of the training set).

Table 2. ROC-AUC across all methods and datasets. *Unsupervised*: no labels. *Ours*: proposed pipeline with LLM-generated weak labels and contrastive fine-tuning when $|\hat{A}| \geq 8$ (no human annotation); ‡proposed configuration. *Oracle*†: trained with ground-truth labels for $\sim 5\%$ of training set (upper bound). All rows: mean$_{\pm \text{std}}$ over 6 runs. Gap↑: fraction of the [best unsup, best oracle] interval recovered by our best variant.

	Method	20 News	WikiNews	HS Tweets	HateBR
Unsupervised	IForest [Liu et al. 2008]	$0.864_{\pm.045}$	$0.687_{\pm.027}$	$0.531_{\pm.024}$	$0.380_{\pm.032}$
	LOF [Breunig et al. 2000]	$0.856_{\pm.000}$	$0.676_{\pm.000}$	$0.498_{\pm.000}$	$0.561_{\pm.000}$
	OCSVM [Schölkopf et al. 2001]	$0.770_{\pm.000}$	$0.666_{\pm.000}$	$0.553_{\pm.000}$	$0.372_{\pm.000}$
	DeepSVDD [Ruff et al. 2018]	$0.748_{\pm.056}$	$0.620_{\pm.042}$	$0.503_{\pm.025}$	$0.539_{\pm.041}$
	AutoEncoder	$0.900_{\pm.003}$	$0.743_{\pm.011}$	$0.575_{\pm.003}$	$0.468_{\pm.004}$
	VAE	$0.920_{\pm.000}$	$0.751_{\pm.012}$	$0.526_{\pm.004}$	$0.381_{\pm.000}$
	HBOS [Goldstein and Dengel 2012]	$0.917_{\pm.000}$	$0.743_{\pm.000}$	$0.540_{\pm.000}$	$0.377_{\pm.000}$
Ours	DeepSAD (Qwen 7B)	$0.811_{\pm.104}$	$0.692_{\pm.062}$	$0.808_{\pm.032}$	$0.669_{\pm.082}$
	DeepSAD (Qwen 14B)	$0.949_{\pm.022}$	$0.785_{\pm.043}$	$0.827_{\pm.027}$	$0.629_{\pm.116}$
	MLP (Qwen 7B)	$0.763_{\pm.132}$	$0.718_{\pm.090}$	$0.861_{\pm.018}$	$\mathbf{0.742}_{\pm.087}$
	MLP (Qwen 14B)‡	$\mathbf{0.967}_{\pm.015}$	$\mathbf{0.885}_{\pm.027}$	$\mathbf{0.867}_{\pm.025}$	$0.695_{\pm.087}$
Oracle†	DeepSAD [Ruff et al. 2020]	$0.995_{\pm.004}$	$0.837_{\pm.034}$	$0.904_{\pm.020}$	$0.791_{\pm.025}$
	DevNet [Das et al. 2023]	$0.910_{\pm.133}$	$0.731_{\pm.073}$	$0.786_{\pm.062}$	$0.764_{\pm.046}$
	MLP	$0.998_{\pm.000}$	$0.936_{\pm.003}$	$0.946_{\pm.003}$	$0.886_{\pm.004}$
	XGBOD [Zhao et al. 2019]	$0.996_{\pm.000}$	$0.943_{\pm.002}$	$0.946_{\pm.000}$	$0.877_{\pm.004}$
	Gap↑	60%	70%	79%	56%

‡Proposed configuration. †Oracle: semi-supervised with ground-truth labels ($\sim 5\%$ of training set).

Unsupervised ceiling. The unsupervised block reveals a sharp difficulty divide: on 20 Newsgroups almost every method exceeds 0.85 AUC, but on the hate speech datasets the best barely exceeds chance (HateBR: 0.561, HS Tweets: 0.575). WikiNews sits between (0.751). This confirms hate speech as the regime where supervision is most needed — and least available.

LLM-supervised gains. Every LLM-supervised configuration surpasses the unsupervised ceiling on the two hate speech datasets. The proposed **MLP (Qwen 14B)**† achieves the best AUC on three of four datasets: HS Tweets rises from 0.575 to 0.867 (79% gap recovered), WikiNews from 0.751 to 0.885 (70%), and 20 Newsgroups reaches 0.967 (60%). On HateBR the 7B model outperforms the 14B (0.742, 56% gap recovered) — a cross-lingual pattern analysed in Section 4.2. Globally, mean AUC rises from 0.702 to 0.865 with zero human annotation. Despite using only $N = 200$ LLM-annotated candidates — of which only 7–16 are flagged as anomalous — the best variant falls 3–13 pp short of the upper bound, which uses 14–91$\times$ more anomaly labels on the three harder datasets.

4.2. RQ2: What Drives the Approach's Gains?

To disentangle each component's contribution, we use a 2×2 factorial design varying classifier (MLP vs. DeepSAD) and embedding (original vs. SetFit-refined); all cells share

identical LLM labels.

Model choice. MLP outperforms DeepSAD on three of four datasets for 7B (global mean $+0.026$) and all four for 14B ($+0.056$), with lower variance (std 0.015 vs. 0.022 on 20 Newsgroups, 14B). The cross-entropy loss averages over thousands of clean negatives, making MLP less sensitive to mislabeled positives than the DeepSAD hypersphere.

Embedding refinement. Figure 2 visualises the full 2×2 ablation. Contrastive fine-tuning via SetFit improves AUC on three of four datasets but can hurt when labels are systematically noisy. On the **hate speech datasets**, where both LLMs produce directionally correct labels, SetFit yields the largest and most consistent gains ($\Delta = +0.110$–0.117 on HS Tweets; $+0.050$–0.099 on HateBR). On **20 Newsgroups**, gains are modest ($\Delta = +0.017$ for 7B, $+0.035$ for 14B) despite reliable activation, because the encoder already separates the classes well. The exception is **WikiNews** with the 7B model: low-quality labels shift embeddings in the wrong direction ($\Delta = -0.069$), the only case where SetFit degrades performance. The 14B avoids this failure ($\Delta = +0.019$), confirming that label quality gates the benefit of contrastive refinement.

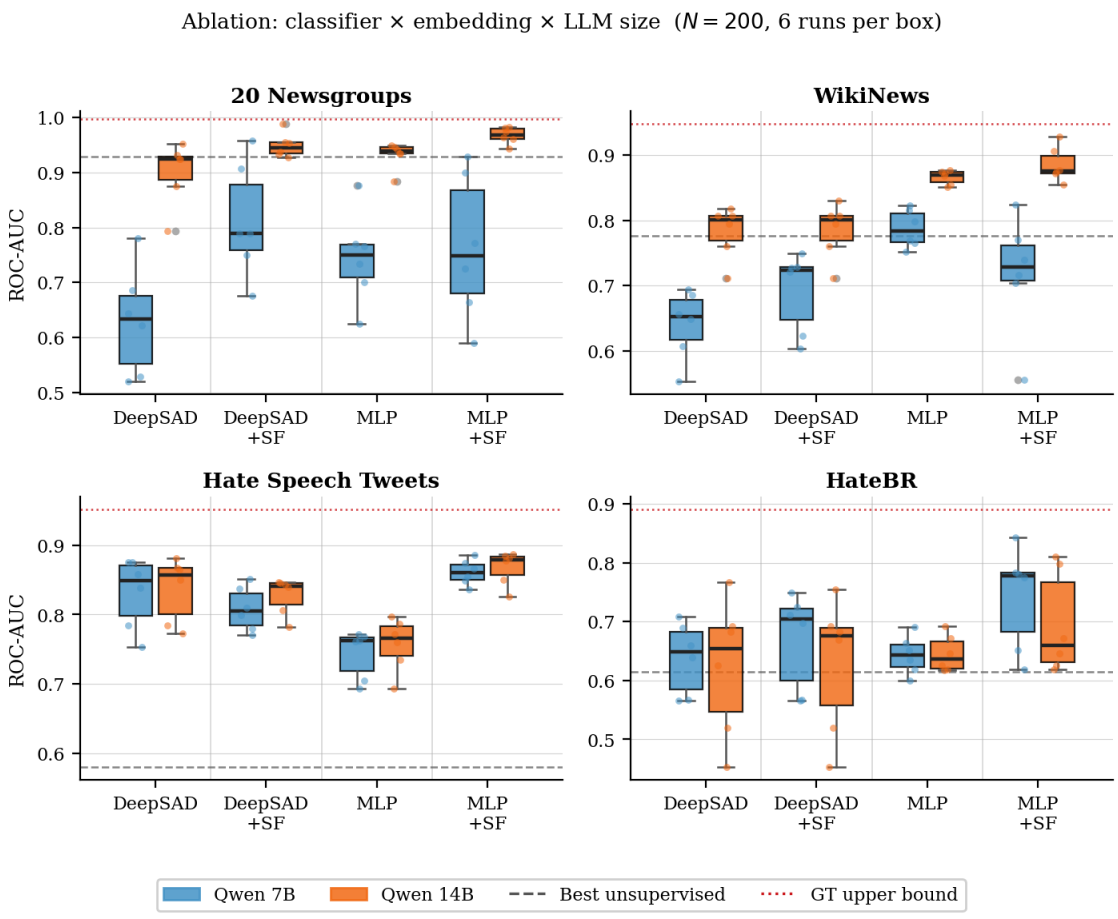

Figure 2. 2×2 **ablation at** $N = 200$ **(6 runs per box: 3 seeds** $\times$ **2 sampling strategies). Blue: Qwen 7B; orange: Qwen 14B. Dashed: best unsupervised; dotted: oracle upper bound.**

LLM capacity and noise propagation. Comparing the 14B and 7B rows in Table 2, the larger model is not uniformly superior: MLP 14B exceeds MLP 7B on WikiNews

(+0.167) and 20 Newsgroups (+0.204), but the 7B *outperforms* on HateBR (+0.047), suggesting the bottleneck for Portuguese hate speech is cultural alignment rather than capacity — validating the low-cost design (single T4 GPU). The ablation reveals a noise propagation chain: the LLM is the *source*, contrastive fine-tuning the *amplifier*, and the AD model the *filter*.

4.3. RQ3: Where Does the Approach Succeed and Fail?

The previous sections show that the approach recovers 56–79% of the gap but gains less on HateBR (14B) and 20 Newsgroups (7B). To understand *why*, we trace downstream AUC back to annotation quality (Table 3). The results reveal that annotation quality varies substantially across datasets and model sizes, yet downstream performance remains robust to low-quality labels in most cases. We identify three annotation regimes and trace their downstream impact.

Table 3. LLM annotation quality at $N = 200$ (random sampling, mean over 3 seeds). Precision and recall are computed against ground-truth labels; $\bar{n}$: mean number of positive labels returned.

Dataset	Model	Prec.	Recall	$\bar{n}$	AUC‡
20 Newsgroups	Qwen 7B	0.075	0.958	134.3	0.763
	Qwen 14B	0.172	1.000	59.3	0.967
WikiNews	Qwen 7B	0.038	0.090	24.3	0.718
	Qwen 14B	0.678	0.556	8.0	0.885
HS Tweets	Qwen 7B	0.369	0.475	13.0	0.861
	Qwen 14B	0.405	0.490	13.0	0.867
HateBR	Qwen 7B	0.565	0.463	9.7	0.742
	Qwen 14B	0.644	0.387	7.3	0.695

‡MLP downstream AUC (proposed configuration).

Implicit vocabulary gaps (20 Newsgroups). Both models achieve near-perfect recall (0.96–1.00) but low precision (0.075–0.172), flooding the weak label set with false positives from normal hockey posts that the LLM misclassifies — it fails to associate implicit domain vocabulary (team abbreviations like *Pens* for Pittsburgh Penguins, NHL jargon) with ice hockey. Crucially, the pipeline remains robust to moderate levels of this noise: the 14B still achieves 0.967 AUC because the MLP's cross-entropy loss dilutes the mislabeled positives among thousands of clean negatives, and the 14B's stronger world knowledge cuts the false-positive volume in half. The 7B's much higher noise level ($\bar{n} = 134$ flagged instances, precision 0.075) does reduce performance (AUC 0.763), marking an empirical limit of the pipeline's noise tolerance.

Criterion polarity error (WikiNews, 7B). Among all configurations, WikiNews (7B) is the only case where SetFit reduces downstream AUC relative to the corresponding non-SetFit baseline. Manual inspection of weak labels suggests that the model correctly identifies article semantics, but appears to assign the opposite label polarity for some normal political articles. For example, an article about Hugo Chávez receives score 0.8 from the 7B with the reason *"primarily discusses political actions…making it clearly political"* — a semantically correct explanation paired with a positive anomaly label. The 14B assigns the same article score 0.0, correctly recognising it as normal. This is

also the only configuration where SetFit amplifies the annotation error ($\Delta_{\text{AUC}} = -0.069$). With the 14B (precision 0.678), the same pipeline reaches AUC 0.885, suggesting that the failure originates in the annotator rather than in the downstream architecture.

Cross-lingual calibration mismatch (HateBR). On HateBR, the pipeline's downstream AUC is driven by the 7B model (0.742), which outperforms the 14B (0.695). The annotation table explains why: the 7B produces more positive labels ($\bar{n} = 9.7$ vs. 7.3) with comparable precision, suggesting greater sensitivity to low-intensity Brazilian Portuguese offense markers. The comment *"O cara não trabalha meu. . . "* (a personal jab) is flagged by the 7B (score 0.8) but missed by the 14B (score 0.2). The larger model appears to apply a higher decision threshold, missing culturally specific expressions and producing fewer labels that, despite higher individual precision, result in smaller downstream gains than the 7B variant.

When annotation quality is **consistent** — as on HS Tweets, where both models converge at precision ≈ 0.37–0.40 and recall ≈ 0.48–0.49 — the pipeline delivers its strongest gains (+0.110–0.117 AUC). Taken together, these regimes suggest that the pipeline is more tolerant to *quantitative* noise — large numbers of false positives — than to *directional* errors, where the annotator appears to assign the wrong label polarity. In our experiments, 20 Newsgroups (14B) reaches 0.967 AUC despite precision of only 0.172, whereas WikiNews (7B) is the only configuration where SetFit amplifies the annotation error. These observations suggest that verifying the LLM's interpretation of the normality criterion may be more important than maximising raw label count.

5. Conclusion

We presented a framework that transfers weak labels from a compact, locally deployed LLM into a lightweight text anomaly detector, requiring no human annotators, cloud APIs, or annotation budget. Across two languages and two tasks, the approach raises mean ROC-AUC from 0.702 to 0.865, closing 56–79% of the gap to oracle upper bounds.

Three findings emerge from ablating the pipeline. First, MLP consistently outperforms DeepSAD under noisy labels because cross-entropy averages over thousands of clean negatives, absorbing mislabeled positives that distort the DeepSAD hypersphere. Second, contrastive fine-tuning amplifies both good and bad label signals — yielding strong gains when annotation is reliable but degrading performance when labels are systematically inverted. Third, the pipeline tolerates high volumes of false positives(0.967 AUC despite annotation precision of 0.172) but appears less tolerant to directional noise, where the annotator may assign the wrong label polarity. This has a practical implication: practitioners should verify prompt alignment before scaling annotation.

Future work includes four directions: (i) LLM-based *data augmentation*, generating synthetic anomalies to increase the positive set beyond what the corpus naturally contains; (ii) *semantically guided sampling*, where the LLM or a richer embedding model selects candidates near the normal–anomaly boundary rather than relying on blind random or coverage strategies; (iii) *soft-label distillation*, replacing hard binary labels with continuous supervision from the LLM's calibrated probabilities to better exploit annotator uncertainty; and (iv) *deeper encoder adaptation*, such as full fine-tuning of the sentence transformer under weak supervision, to understand the trade-off between representational capacity and noise sensitivity.

6. Limitations

Our evaluation covers four datasets across two languages and two tasks, but all use a fixed $\sim 5\%$ contamination rate; performance under different rates and on other languages or domains remains untested. The LLM annotation relies on dataset-specific prompts whose sensitivity was not systematically evaluated. The approach assumes that the practitioner can articulate a normality boundary in natural language — when this prior knowledge is misspecified, annotation quality degrades (as shown in the WikiNews 7B failure mode).

7. Ethics Statement

All datasets are publicly available and previously published. The hate speech datasets contain offensive content by design; we use them solely for evaluation and do not redistribute the data. LLM annotations were generated locally without transmitting data to external APIs.

8. Generative AI Disclosure

This work uses generative AI in three roles: (i) *As the experimental method*: Qwen2.5-Instruct (7B and 14B) is used as the LLM annotator within the proposed pipeline (Section 3.2). (ii) *As a software engineering aid*: Claude Sonnet 4.6 (Anthropic) was used to assist with code prototyping, debugging, and implementation support during experimentation. (iii) *As a writing aid*: Claude Opus 4 (Anthropic) was used to assist with text revision.

All scientific content, experimental design, implementation choices, and claims are the sole responsibility of the authors.

Acknowledgments

This study was financed in part by the Coordenação de Aperfeiçoamento de Pessoal de Nível Superior – Brasil (CAPES) – Finance Code 001. The authors would like to thank the National Council for Scientific and Technological Development (CNPq grant #312360/2023-1) and INCT-TILDIAR (CNPq grant #408490/2024-1).

References

Ait-Saada, M. and Nadif, M. (2023). Unsupervised anomaly detection in multi-topic short-text corpora. In *Proceedings of the 17th Conference of the European Chapter of the Association for Computational Linguistics (EACL)*.

Boutalbi, K., Loukil, F., Verjus, H., Telisson, D., and Salamatian, K. (2023). Machine learning for text anomaly detection: A systematic review. In *2023 IEEE 47th Annual Computers, Software, and Applications Conference (COMPSAC)*, pages 1319–1324.

Breunig, M. M., Kriegel, H.-P., Ng, R. T., and Sander, J. (2000). Lof: identifying density-based local outliers. In *Proceedings of the 2000 ACM SIGMOD International Conference on Management of Data*, SIGMOD '00, page 93–104, New York, NY, USA. Association for Computing Machinery.

Chandola, V., Banerjee, A., and Kumar, V. (2009). Anomaly detection: A survey. *ACM Computing Surveys*, 41(3):71–97.

Das, A. S., Ajay, A., Saha, S., and Bhuyan, M. (2023). Few-shot anomaly detection in text with deviation learning. In *Neural Information Processing: 30th International Conference, ICONIP 2023, Changsha, China, November 20–23, 2023, Proceedings, Part II*, page 425–438, Berlin, Heidelberg. Springer-Verlag.

Garcia, K., Shiguihara, P., and Berton, L. (2024). Breaking news: Unveiling a new dataset for portuguese news classification and comparative analysis of approaches. *PLOS ONE*, 19(1):1–15.

Gerganov, G. et al. (2023). llama.cpp: LLM inference in C/C++. `https://github.com/ggerganov/llama.cpp`.

Goldstein, M. and Dengel, A. (2012). Histogram-based outlier score (HBOS): A fast unsupervised anomaly detection algorithm. In Wölfl, S., editor, *KI-2012: Poster and Demo Track*, pages 59–63, Saarbrücken, Germany.

He, X., Lin, Z., Gong, Y., Jin, A.-L., Zhang, H., Lin, C., Jiao, J., Yiu, S. M., Duan, N., and Chen, W. (2024). AnnoLLM: Making large language models to be better crowdsourced annotators. In Yang, Y., Davani, A., Sil, A., and Kumar, A., editors, *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 6: Industry Track)*, pages 165–190, Mexico City, Mexico. Association for Computational Linguistics.

Hendrycks, D., Mazeika, M., and Dietterich, T. (2019). Deep anomaly detection with outlier exposure. In *Proceedings of the International Conference on Learning Representations (ICLR)*.

Khosla, P., Teterwak, P., Wang, C., Sarna, A., Tian, Y., Isola, P., Maschinot, A., Liu, C., and Krishnan, D. (2020). Supervised contrastive learning. In Larochelle, H., Ranzato, M., Hadsell, R., Balcan, M., and Lin, H., editors, *Advances in Neural Information Processing Systems*, volume 33, pages 18661–18673. Curran Associates, Inc.

Lang, K. (1995). Newsweeder: Learning to filter netnews. `http://www.cs.cmu.edu/~kenlang`. CMU, unpublished manuscript.

Leite, J. A., Silva, D. F., Bontcheva, K., and Scarton, C. (2020). Toxic language detection in social media for brazilian portuguese: New dataset and multilingual analysis. *CoRR*, abs/2010.04543.

Liu, B., Zhan, L.-M., Lu, Z., Feng, Y., Xue, L., and Wu, X.-M. (2024). How good are LLMs at out-of-distribution detection? In Calzolari, N., Kan, M.-Y., Hoste, V., Lenci, A., Sakti, S., and Xue, N., editors, *Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING 2024)*, pages 8211–8222, Torino, Italia. ELRA and ICCL.

Liu, F. T., Ting, K. M., and Zhou, Z.-H. (2008). Isolation forest. In *2008 Eighth IEEE International Conference on Data Mining*, pages 413–422.

Maia, F. and Costa, A. (2025). Learning with few: A comparative study of multilingual text anomaly detection. In *Anais do XVI Simpósio Brasileiro de Tecnologia da Informação e da Linguagem Humana*, pages 233–246, Porto Alegre, RS, Brasil. SBC.

Manolache, A., Brad, F., and Burceanu, E. (2021). DATE: Detecting anomalies in text via self-supervision. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL)*.

Pang, G., Shen, C., Cao, L., and Hengel, A. V. D. (2021). Deep learning for anomaly detection: A review. *ACM Comput. Surv.*, 54(2).

Qwen Team (2025). Qwen2.5 technical report. arXiv preprint arXiv:2412.15115.

Ratner, A., Bach, S. H., Ehrenberg, H., Fries, J., Wu, S., and Ré, C. (2020). Snorkel: rapid training data creation with weak supervision. *The VLDB Journal*, 29(2-3):709–730.

Reimers, N. and Gurevych, I. (2019). Sentence-BERT: Sentence embeddings using Siamese BERT-networks. In Inui, K., Jiang, J., Ng, V., and Wan, X., editors, *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 3982–3992, Hong Kong, China. Association for Computational Linguistics.

Ruff, L., Vandermeulen, R., Goernitz, N., Deecke, L., Siddiqui, S. A., Binder, A., Müller, E., and Kloft, M. (2018). Deep one-class classification. In Dy, J. and Krause, A., editors, *Proceedings of the 35th International Conference on Machine Learning*, volume 80 of *Proceedings of Machine Learning Research*, pages 4393–4402. PMLR.

Ruff, L., Vandermeulen, R. A., Görnitz, N., Binder, A., Müller, E., and Kloft, M. (2020). Deep semi-supervised anomaly detection. In *International Conference on Learning Representations (ICLR)*.

Ruff, L., Zemlyanskiy, Y., Vandermeulen, R., Schnake, T., and Kloft, M. (2019). Self-attentive, multi-context one-class classification for unsupervised anomaly detection on text. In Korhonen, A., Traum, D., and Màrquez, L., editors, *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 4061–4071, Florence, Italy. Association for Computational Linguistics.

Schölkopf, B., Platt, J. C., Shawe-Taylor, J., Smola, A. J., and Williamson, R. C. (2001). Estimating the support of a high-dimensional distribution. *Neural Computation*, 13(7):1443–1471.

Sener, O. and Savarese, S. (2018). Active learning for convolutional neural networks: A core-set approach. In *International Conference on Learning Representations (ICLR)*.

Sharma, R. (2018). Twitter sentiment analysis for hate speech detection. `https://huggingface.co/datasets/tweets-hate-speech-detection/tweets_hate_speech_detection`. Accessed: 2025-09-03.

Tunstall, L., Reimers, N., Jo, U. E. S., Bates, L., Korat, D., Wasserblat, M., and Pereg, O. (2022). Efficient few-shot learning without prompts. Presented at the NeurIPS 2022 Workshop on Efficient Natural Language and Speech Processing (ENLSP-II).

Wang, S., Liu, Y., Xu, Y., Zhu, C., and Zeng, M. (2021). Want to reduce labeling cost? GPT-3 can help. In *Findings of the Association for Computational Linguistics: EMNLP 2021*, pages 4195–4205, Punta Cana, Dominican Republic. Association for Computational Linguistics.

Xu, A., Ren, X., and Jia, R. (2023a). Contrastive novelty-augmented learning: Anticipating outliers with large language models. In Rogers, A., Boyd-Graber, J., and Okazaki, N., editors, *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 11778–11801, Toronto, Canada. Association for Computational Linguistics.

Xu, H. (2022). Deepod: A benchmarking framework for deep outlier detection. `https://github.com/xuhongzuo/DeepOD`.

Xu, R. and Ding, K. (2025). Large language models for anomaly and out-of-distribution detection: A survey. In Chiruzzo, L., Ritter, A., and Wang, L., editors, *Findings of the Association for Computational Linguistics: NAACL 2025*, pages 6007–6027, Albuquerque, New Mexico. Association for Computational Linguistics.

Xu, Y., Gábor, K., Milleret, J., and Segond, F. (2023b). Comparative analysis of anomaly detection algorithms in text data. In Mitkov, R. and Angelova, G., editors, *Proceedings of the 14th International Conference on Recent Advances in Natural Language Processing*, pages 1234–1245, Varna, Bulgaria. INCOMA Ltd., Shoumen, Bulgaria.

Yang, T., Nian, Y., Li, L., Xu, R., Li, Y., Li, J., Xiao, Z., Hu, X., Rossi, R. A., Ding, K., Hu, X., and Zhao, Y. (2025). AD-LLM: Benchmarking large language models for anomaly detection. In Che, W., Nabende, J., Shutova, E., and Pilehvar, M. T., editors, *Findings of the Association for Computational Linguistics: ACL 2025*, pages 1524–1547, Vienna, Austria. Association for Computational Linguistics.

Zhao, Y., Nasrullah, Z., and Li, Z. (2019). Pyod: A python toolbox for scalable outlier detection. *Journal of Machine Learning Research*, 20(96):1–7.

Zhou, Z.-H. (2018). A brief introduction to weakly supervised learning. *National Science Review*, 5(1):44–53.