

Training Lightweight PII Text Detectors via Rubric-Guided Contrastive Data Synthesis

Shota Ekuni^{1,3} Jie Zeng² Yohei Kobashi³
Takeshi Kojima³ Yusuke Iwasawa³ Yutaka Matsuo³

¹A-wave, Inc., Osaka, Japan

²Seikei University, Tokyo, Japan

³The University of Tokyo, Tokyo, Japan

ekunish@gmail.com, jie-zeng@st.seikei.ac.jp

{yohei.kobashi, t.kojima, iwasawa, matsuo}@weblab.t.u-tokyo.ac.jp

Abstract

The widespread use of large language models (LLMs) has increased the risk of leaking personally identifiable information (PII), making PII detection increasingly important. PII detection must process large text volumes efficiently, but labeled PII text is difficult to collect, share, and redistribute for privacy reasons. We aim to develop rubric-guided contrastive data synthesis for training lightweight PII text detectors from a few labeled seeds and to determine, through comparative experiments on PII benchmark datasets, whether the proposed method improves PII text detector classification performance. From about ten labeled examples, our method induces explicit criteria for distinguishing the two labels (rubrics), generates contrastive Sensitive/Non-sensitive sentence pairs, verifies and deduplicates them, and trains RoBERTa. Experiments on SPeDaC1 and SPY with Qwen3.5-9B and Gemma-4-26B-A4B showed that our method outperformed four baselines in all four settings and significantly improved Macro F1 over the Self-Instruct-style baseline in three of four settings. Overall, we developed rubric-guided contrastive data synthesis for training lightweight PII text detectors from a few labeled seeds and confirmed through comparative experiments on PII benchmark datasets that the proposed method improves classification performance. Our code is available at <https://github.com/ekunish/emnlp2026-code>.

1 Introduction

With the widespread use of large language models (LLMs), technologies for detecting personally identifiable information (PII) in text have become increasingly important. While LLMs have enabled diverse applications such as translation, summarization, and question answering, their pre-training typically relies on massive corpora collected from the Internet, including social media, which are likely

to contain PII. As a result, models risk reproducing and leaking the PII they have learned (Huang et al., 2022; Lukas et al., 2023). Moreover, PII can also appear in user inputs and generated text when LLMs are used. Detecting and appropriately handling PII in text is therefore necessary.

Rule-based methods have long been studied for PII detection, but fixed vocabulary lists and regular expressions alone are less effective on unstructured text; data-driven and hybrid approaches are therefore also used (Negash et al., 2023; Savkin et al., 2025). Although an LLM can be used to classify text directly, a lightweight classifier that does not invoke an LLM at inference time is more practical for processing large volumes of text quickly, such as for filtering training data and auditing large-scale logs. However, training such a classifier requires large-scale annotated data containing PII, and such data is difficult to collect, share, and redistribute for privacy reasons (Pilán et al., 2022; Savkin et al., 2025).

Synthesizing training data with an LLM from a small number of labeled examples is a promising remedy (Liu et al., 2024; Santoso et al., 2024; Savkin et al., 2025). However, synthetic data must maintain label fidelity and diversity (Liu et al., 2024). Moreover, the label can be contextual: in SPY, entities of the same type are distinguished according to whether they concern the text’s author or an unrelated person (Savkin et al., 2025).

Accordingly, this study aims to develop rubric-guided contrastive data synthesis for training lightweight PII text detectors from a few labeled seeds and to determine, through comparative experiments on PII benchmark datasets, whether the proposed method improves PII text detector classification performance. Our contributions are three-fold.

- We propose a method that induces rubrics from a few labeled examples and generates

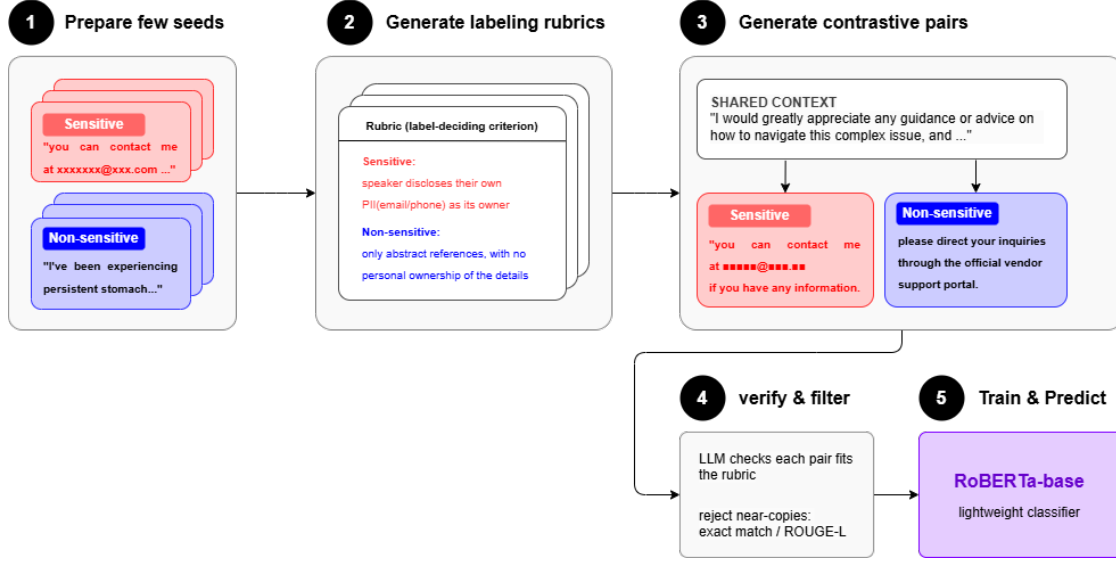


Figure 1: Overview of the proposed method. From a small set of labeled seed examples, we induce rubrics and, based on each rubric, generate Sensitive / Non-sensitive contrastive pairs. The generated pairs are verified by a separate prompt and, after rejection by exact match and ROUGE-L, become training data for RoBERTa.

contrastive synthetic training data based on them.

- On SPeDaC1 (Gambarelli et al., 2023) and SPY (Savkin et al., 2025), using Qwen3.5-9B and Gemma-4-26B-A4B-NVFP4, we show gains over four baselines and over the Self-Instruct-style baseline in three of four settings; the remaining setting has no confirmed difference.
- Through ablations of the method’s components, we clarify which design elements account for the improvement and what limitations remain.

2 Related Work

PII detection and benchmarks PII detection is a foundational technology for anonymization, data sharing, and preprocessing for model training, and it has been studied across multiple domains such as email, healthcare, legal, and finance. At the same time, because the training data itself contains sensitive information, data collection and sharing are constrained. SPeDaC is a benchmark for sensitive personal data classification using Web-derived text (Gambarelli et al., 2023), and SPY is a synthetic PII benchmark covering scenarios such as healthcare and legal (Savkin et al., 2025). Minamoto et al. (2026) train corpus-filtering classifiers using LLM annotations of a large Japanese Web corpus, whereas our method

constructs classifier training data from a small labeled seed set through rubric-guided contrastive synthesis. Other synthetic datasets targeting specific domains such as financial documents have also been proposed (Mishra et al., 2024).

Synthesizing text from few examples Data synthesis with LLMs is widely used for low-resource classification and harmful-content detection (Chen et al., 2023; Santoso et al., 2024; Hartvigsen et al., 2022). ZeroGen (Ye et al., 2022) and Self-Instruct (Wang et al., 2023) use LLM-generated data for downstream tasks and instruction tuning. Self-Instruct in particular uses similarity measures such as ROUGE-L to suppress duplication in the generated data. To avoid generated text becoming a superficial copy of seed examples or previously generated text, we also combine rejection by normalized exact match and ROUGE-L.

3 Methods

Figure 1 shows an overview of the proposed method. From a few labeled examples in a prepared seed pool, we induce rubrics and, based on these rubrics, synthesize Sensitive / Non-sensitive contrastive pairs. We then verify the synthetic text and use the examples that satisfy our conditions as training data for a binary sentence classifier. A *prompt* is the fixed instruction template supplied to the LLM, whereas a *rubric* is a dataset-specific criterion for distinguishing the two labels, induced by that prompt from the labeled seed examples.

3.1 Rubric induction

We extract a few labeled Sensitive and Non-sensitive texts each. We give them to the LLM as few-shot prompts and have it output, in multiple patterns, the conditions that make the Sensitive side Sensitive and the conditions under which a comparable Non-sensitive side does not become Sensitive. We call these output conditions rubrics. Rubrics are created by the LLM based solely on the provided support examples, without using external PII definitions or rules. Each rubric includes a Sensitive-side condition and a Non-sensitive-side condition. This inserts a step that makes the LLM explicitly articulate, once, the basis for separating labels within the seed pool, rather than merely producing “sentences similar to the seed examples.” In the main condition, we induce 25 rubrics.

3.2 Contrastive pair generation

Based on each rubric, we generate one Sensitive sentence and one Non-sensitive sentence. The two sentences share topic and style to some extent while differing in the basis that separates the labels. The aim of this design is to make it easier for the classifier to learn differences relevant to distinguishing the labels rather than label-irrelevant stylistic differences. For contrastive pair generation and verification, we provide not only the examples used for rubric induction but the entire seed pool as few-shot prompts. This keeps the style, length, and labeling tendency of the generated text from deviating substantially from the whole seed pool, while still using local rubrics.

3.3 Verification of generated text

After generation, the LLM verifies whether the Sensitive and Non-sensitive sentences are correctly labeled in light of the rubric and support examples, whether the pair forms a valid contrast, and whether the text is not a copy of the support examples. In addition to normalized exact match, we reject any candidate whose ROUGE-L F1 against a seed sentence or an already-accepted generated sentence is at least 0.95, and we resample when a candidate is rejected.

For sampling at generation time, we use temperature = 1.0 and top-p = 0.9 for rubric induction and contrastive pair generation, and temperature = 0.0 and top-p = 0.9 for verification.

3.4 Classifier training

Using the generated text that passes verification as training data, we train RoBERTa. Following the SPeDaC setup (Gambarelli et al., 2023), we fine-tune roberta-base for 15 epochs using AdamW. We use a batch size of 8, a learning rate of 1×10^{-5} , a dropout rate of 0.3, and a maximum sequence length of 256.

4 Experiments

Using a small labeled seed pool extracted from the train split of each target benchmark dataset, we generate training data with each synthesis method, train a RoBERTa (Liu et al., 2019) classifier, and evaluate English sentence-level Sensitive / Non-sensitive classification on each benchmark’s test split. What we evaluate is not a method that lets an LLM directly classify test sentences, but a method that trains a compact encoder using training data synthesized by an LLM. For rubric induction, synthetic-data generation, and verification, we use Qwen3.5-9B and Gemma-4-26B-A4B-NVFP4; both run on a single RTX 3090 with 24 GB VRAM.

4.1 Datasets

As binary Sensitive / Non-sensitive classification tasks, we use two datasets, SPeDaC1 (Gambarelli et al., 2023) and SPY (Savkin et al., 2025). SPeDaC1 (Gambarelli et al., 2023) is a benchmark for personal data classification using Web-derived text, framed as binary Sensitive / Non-sensitive classification. SPY (Savkin et al., 2025) is a dataset in which seven types of PII entities, such as email addresses and personal IDs, are embedded as placeholders into LLM-generated text in the Legal and Medical domains. We replace the placeholders with realistically formatted values using Faker (Faraglia, 2012), split the text into sentences, and treat sentences containing a PII entity as Sensitive and those without as Non-sensitive. From the resulting sentences, we extract and use a label-balanced subset (10,000 train, 1,000 validation, 1,000 test, with Sensitive : Non-sensitive = 1 : 1 in each split).

Table 1 shows the number of examples per split for each dataset. In the main experiments, each seed pool is a random sample of 10 examples (5 Sensitive and 5 Non-sensitive) from the training split. The official validation split selects the RoBERTa checkpoint, and the test split is evaluated only after checkpoint selection. Neither validation

Dataset	Split	Sensitive	Non-sensitive	Total
SPeDaC1	train	3,817	3,872	7,689
	validation	442	404	846
	test	1,087	1,132	2,219
SPY	train	5,000	5,000	10,000
	validation	500	500	1,000
	test	500	500	1,000

Table 1: Number of examples per split for each dataset (by Sensitive / Non-sensitive). SPY is a label-balanced subset extracted from a larger processed set.

nor test labels are fed back into rubric induction or synthesis.

4.2 Baselines

We use five synthesis methods as baselines. *duplicate* does not use an LLM and creates training data by duplicating the sentences in the seed pool. *paraphrase* uses an LLM to generate paraphrases of the seed sentences. *one-shot* and *few-shot* have the LLM directly generate labeled sentences from a small number of examples. *few-shot* provides the entire 10-example seed pool as few-shot prompts. *Self-Instruct-style* is a synthesis baseline for our binary sentence-classification task based on Self-Instruct’s classification pipeline (Wang et al., 2023). It starts from six generic, human-written privacy instructions shared across datasets. At each iteration it prompts with the six seed instructions and up to two accepted model-generated instructions, accepts a new instruction only when its ROUGE-L similarity to every pooled instruction is below 0.7, and returns accepted instructions to the pool. For each sentence, it selects one instruction from the pool, specifies a Sensitive or Non-sensitive target label, and supplies the same original 5+5 labeled seed examples as few-shot demonstrations. Accepted generated sentences are not added to the few-shot demonstrations in later sentence-generation prompts. Unlike our method, it generates each labeled sentence separately rather than as a contrastive pair. We match the 2,000-item balanced target, generator, sampling settings, LLM verification procedure, normalized exact match, and instance-level ROUGE-L threshold of 0.95. Because these, like our method, train RoBERTa on synthetic data, they are the main comparison targets.

As reference conditions, we also report *official train*, which trains RoBERTa on the official training split, and LLM-as-a-Judge, in which an LLM directly classifies the test set. In LLM-as-a-Judge,

we provide one-shot or few-shot labeled examples and have the LLM directly output the label of each test sentence without training RoBERTa. Because this is a different setting from training on synthetic data, we report it separately as a reference condition in Table 2.

4.3 Evaluation metrics

The main metrics are Macro F1 and Matthews correlation coefficient (MCC). All main results are reported as the mean and standard deviation over three runs that use distinct seed pools and evaluate on the same test split. For paired comparisons, we use 10,000 two-level bootstrap replicates: test items are resampled within gold-label strata and the three observed seed pools are resampled while preserving pairing between methods. We report percentile 95% confidence intervals and apply Holm correction within each comparison family. For each main-table run, we target 2,000 synthetic examples with a Sensitive : Non-sensitive ratio of 1:1. We fixed this common count before the main experiments to balance generation time against the amount of classifier-training data and to keep all methods, datasets, and generators comparable; it was not selected as a per-dataset optimum. Even if the target count is not reached within the trial limit, we train on all obtained examples as long as the completion rate is at least 0.95. The realized range in the twelve proposed runs is 1,966–2,000 examples.

4.4 Ablation study

To examine the effect of each component of our method, we conduct several comparisons. All use the same RoBERTa training settings as the main experiments and, unless otherwise noted, are evaluated with Qwen3.5-9B and $n = 1000$ synthetic examples. First, for the generation structure, we compare (a) single-stage generation that directly produces Sensitive / Non-sensitive pairs without rubrics, (b) a condition that induces rubrics and then generates single sentences individually, and (c) our proposed structure that generates contrastive pairs based on rubrics. For each condition, the number of rubrics is set so as to reach $n = 1000$ ((b) uses 100 and (c) uses 10). Second, for the number of rubrics, we compare 1, 5, 25, and 50. This comparison uses $n = 1000$ and fixes rejection to normalized exact match only. Third, for the seed pool size, we use Qwen3.5-9B and Gemma and compare sizes of 4, 10, 20, and 40. The main com-

Training data / setting	Generator	Classifier	SPeDaC1	SPY
Official train	–	RoBERTa	97.49±0.34/.950±.007	98.80±0.46/.976±.009
Duplicate	–	RoBERTa	73.61±5.26/.481±.091	67.72±5.90/.374±.101
Paraphrase	Qwen	RoBERTa	73.73±4.17/.481±.084	67.34±7.36/.375±.118
Paraphrase	Gemma	RoBERTa	73.45±4.37/.477±.080	67.92±8.18/.387±.132
One-shot	Qwen	RoBERTa	68.14±0.91/.373±.030	69.21±3.10/.436±.045
One-shot	Gemma	RoBERTa	76.23±2.89/.535±.058	71.77±2.50/.494±.061
Few-shot	Qwen	RoBERTa	75.24±5.94/.568±.069	70.32±7.66/.468±.127
Few-shot	Gemma	RoBERTa	71.49±3.50/.504±.052	60.78±10.58/.364±.075
Self-Instruct-style	Qwen	RoBERTa	82.73±3.64/.657±.072	78.55±3.69/.592±.073
Self-Instruct-style	Gemma	RoBERTa	83.23±2.47/.688±.034	78.69±2.20/.604±.020
Ours	Qwen	RoBERTa	83.15±4.69/.672±.089	84.19±2.24/.692±.043
Ours	Gemma	RoBERTa	88.79±0.62/.778±.014	85.95±2.60/.721±.053
LLM-as-a-Judge one-shot	Qwen	Qwen	50.83±1.42/ – –	70.55±0.96/ – –
LLM-as-a-Judge one-shot	Gemma	Gemma	58.72±0.50/ – –	81.05±0.56/ – –
LLM-as-a-Judge few-shot	Qwen	Qwen	54.65±3.14/ – –	74.12±4.10/ – –
LLM-as-a-Judge few-shot	Gemma	Gemma	65.56±5.25/ – –	84.10±1.11/ – –

Table 2: Macro F1 (percentage) / MCC (decimal), reported as mean±sample standard deviation over three distinct seed pools. Self-Instruct-style is the baseline described in Appendix G.2. LLM-as-a-Judge directly classifies test sentences, so only Macro F1 is shown as a reference.

parison fixes the size to 10. Fourth, for the number of synthetic examples, we generate 5000 synthetic examples for each dataset and seed pool and compare subsets containing the first 500, 1000, 2000, 3000, 4000, and 5000 examples in generation order. This comparison fixes the number of rubrics to 25.

5 Results

5.1 Main results

Table 2 shows the main results. Our method outperformed the four baselines in all four settings. All 16 paired Macro-F1 comparisons against these four baselines remained significant after Holm correction; Table 9 reports the corresponding intervals and MCC differences. Against the Self-Instruct-style baseline, the Qwen/SPeDaC1, Qwen/SPY, Gemma/SPeDaC1, and Gemma/SPY Macro-F1 differences were +0.43, +5.64, +5.56, and +7.25 points. Their 95% CIs were [−1.60, 2.24], [3.07, 8.04], [3.05, 8.44], and [2.21, 11.92], respectively. Thus, three settings support an improvement, while Qwen/SPeDaC1 does not establish a difference. The corresponding MCC means for our method were 0.672, 0.692, 0.778, and 0.721. LLM-as-a-Judge showed high values for SPY / Gemma but dropped substantially for SPeDaC1. This indicates that prompts for direct classification alone may not stably apply dataset-specific labeling criteria. For reference, training RoBERTa on the official training split yielded Macro F1/MCC of 0.975/0.950 and 0.988/0.976 for SPeDaC1 and SPY, respectively.

Generation structure	SPeDaC1	SPY
Single-stage	63.72±8.99	72.11±6.95
Rubric → single	79.85±1.43	72.80±6.22
Rubric → pair (ours)	82.02±5.83	83.25±2.13

Table 3: Author-designed generation-structure ablations (Qwen3.5-9B, $n=1000$, Macro F1). Single-stage pair tests pairing without rubrics, rubric-to-single tests rubrics without paired generation, and rubric-to-pair combines both. Rubric counts differ (100 for single sentences and 10 for contrastive pairs), so the latter comparison is diagnostic rather than a pairing-only estimate.

5.2 Ablation study

Generation structure Table 3 shows the ablation of the generation structure. The conditions are author-designed controlled ablations: single-stage pair retains pairing but removes rubric guidance; rubric-to-single retains rubric guidance but removes paired generation; and rubric-to-pair combines both components. Single-stage generation is the lowest, generating single sentences via rubrics improves it, and generating contrastive pairs—the proposed structure—is the highest. Rubric-to-pair minus single-stage pair was +18.30 points for SPeDaC1 (95% CI [13.22, 24.10], Holm-adjusted $p = .0104$) and +11.14 for SPY ([3.53, 17.11], $p = .0620$). Rubric-to-pair minus rubric-to-single was +2.16 ([−4.34, 6.59], $p = 1.0000$) and +10.45 ([3.44, 17.76], $p = .0272$), respectively; because rubric count also differs between these conditions, this comparison does not isolate pair-

Number of rubrics	SPeDaC1	SPY
1	82.43±1.06	76.25±6.15
5	80.14±4.90	81.94±1.49
25	83.21±4.68	84.54±0.58
50	84.12±4.65	79.45±3.82

Table 4: Ablation of the number of rubrics (Qwen3.5-9B, $n=1000$, rejection by normalized exact match, Macro F1).

ing alone. The generation-structure ablation indicates that, rather than directly generating synthetic text, it is important for performance to first create rubrics as an intermediate representation and then generate Sensitive / Non-sensitive contrastive pairs based on those rubrics.

Number of rubrics Table 4 shows the ablation of the number of rubrics. The relation is non-monotonic: 50 rubrics have the highest SPeDaC1 mean, whereas SPY peaks at 25 and then drops. The 50-minus-25 difference is +0.90 points for SPeDaC1 (95% CI $[-1.45, 3.74]$) and -5.08 for SPY ($[-9.21, -1.68]$, Holm-adjusted $p = .021$). We therefore use 25 as a common fixed setting rather than claiming it as an optimum; adding rubrics may broaden the criteria represented while reducing the examples allocated to each criterion, and this experiment does not isolate those effects.

Seed pool size Table 5 shows the effect of the seed pool size. This comparison uses $n = 1000$, which differs from the main table in the number of generated examples. The 4-example condition has high variance and is unstable, whereas at 10 or more examples our method exceeds duplicate in many conditions. The 10-example seed pool of the main condition is a reasonable balance between the small number of required labeled examples and performance stability.

Number of synthetic examples Table 6 shows the effect of the number of synthetic examples. For Qwen3.5-9B, SPeDaC1 attains its highest mean at 500 examples and does not improve monotonically as the number increases, whereas SPY remains near its 2,000-example score through 4,000 examples. Relative to 2,000 examples, the 1,000-example differences were +1.48 points for SPeDaC1 (95% CI $[-4.17, 6.37]$) and -0.62 for SPY ($[-4.22, 2.57]$). At 3,000 examples, SPeDaC1 declined by 4.26 points ($[-6.38, -1.99]$, Holm-adjusted $p = .0148$), while SPY changed by +0.40 ($[-1.61, 2.88]$). The effect of the number of syn-

Size	Qwen	Gemma	duplicate
SPeDaC1			
4	72.92±17.39	79.16±11.10	62.25±4.27
10	85.96±2.65	88.12±5.04	70.69±4.77
20	81.07±2.16	90.84±1.90	82.86±0.64
40	90.87±0.30	92.49±0.68	86.29±3.14
SPY			
4	63.94±14.65	75.84±9.80	65.47±11.67
10	83.04±0.98	85.06±1.87	67.76±4.79
20	82.71±5.46	84.14±5.21	70.75±4.30
40	85.29±3.07	86.75±0.90	81.30±3.99

Table 5: Macro F1 for varying seed pool size ($n=1000$). Values are the mean and standard deviation over three independent runs.

Size	SPeDaC1	SPY
500	85.17±4.05	80.63±3.78
1000	84.63±2.70	83.57±1.18
2000	83.15±4.69	84.19±2.24
3000	78.89±3.80	84.59±1.89
4000	79.12±5.84	84.81±2.82
5000	80.65±1.96	83.07±2.31

Table 6: Macro F1 by number of synthetic examples with Qwen3.5-9B. Each value is the mean and standard deviation over three seed pools. Each size consists of the first indicated number of examples in generation order from the same 5000 generated examples.

thetic examples was therefore dataset-dependent. For SPeDaC1, increasing the subset from the first 2,000 to the first 3,000 generated examples reduced test Sensitive recall from .758 to .635 while increasing Non-sensitive recall from .905 to .950; validation recall changed in the same directions (.728 to .608 and .904 to .946), despite a 1:1 label balance. These asymmetric recall changes indicate that the classifier predicted Non-sensitive more often. They are not attributable to the class ratio of the synthetic training data, which remained 1:1 at both sizes. The additional 500 pairs were unevenly distributed across rubrics (6–43 per rubric) because the subsets were not constructed to contain equal numbers from each rubric. Examples generated later and retained after duplicate filtering may also contribute less new information or contain different proportions of expressions associated with each label. These are hypotheses rather than isolated causal explanations, and SPY did not show the same 3,000-example decline.

6 Rubric Analysis

6.1 Dataset-specific label criteria

Examining the rubrics generated by our method, the main basis for separating Sensitive from Non-sensitive differed between SPeDaC1 and SPY. In SPeDaC1, rather than formal identifiers such as email addresses or phone numbers, what repeatedly appeared as the basis for separating labels was whether the text concretely discloses the writer’s own or a family member’s diagnosis, treatment history, physical condition, disability, occupation, income, or living situation. The corresponding Non-sensitive side, even when it included nearby topics such as healthcare or politics, consisted of general commentary, public policy, product descriptions, or article-style writing that does not disclose a specific individual’s private state.

In SPY, by contrast, the main basis was to whom identifiers or contact details—such as email addresses, phone numbers, addresses, usernames, and insurance IDs—were presented as belonging. Even seemingly identical contact details or names were treated as Sensitive when tied to the writer themselves or to a private individual who is the subject of disclosure, and as Non-sensitive when tied to a recipient, customer, organization, work-related contact, or a general business context.

This difference is consistent with prior work showing that PII is not adequately characterized by fixed attribute lists and that privacy-oriented labeling depends on context and the person to be protected; in our results, the operative criteria depended on the dataset’s construction policy and intended use (Narayanan and Shmatikov, 2010; Pilán et al., 2022). Appendix A provides representative induced rubrics for both datasets.

6.2 Human evaluation

Two authors independently rated 60 rubrics and 96 pairs that passed LLM verification, sampled evenly from the 12 dataset–generator–seed-pool settings. Most rubrics were judged to explain why the displayed seed examples received their labels (rater A/B *yes*-rate ranges: .800–1.000/.733–.933), and most sampled accepted sentences were judged to have valid labels (.792–1.000/.905–1.000) and to be natural (.750–.958/1.000). General applicability was rater-dependent for Qwen (A=.400–.467, B=.867; AC1=.486/.219). AC1 is Gwet’s chance-corrected agreement coefficient (Gwet, 2008). Among pairs that passed LLM verifica-

tion, non-abstaining raters judged both labels valid at A/B=.915/.957 on SPeDaC1 and .745/.909 on SPY; because rejected candidates were not retained, these values estimate the precision of LLM verification but not its recall. Gemma’s sampled rubrics and sentences received higher rates for general applicability and naturalness than Qwen’s in these settings. Across the 12 dataset–generator–seed-pool settings, the average human naturalness rate and downstream Macro F1 had Spearman $\rho = .649$; this association based on 12 settings is descriptive, not causal. Appendix E gives the complete protocol, rating criteria, per-cell ratings, and agreement coefficients.

7 Discussion

Regarding practical applicability, the deployed detector uses roberta-base (approximately 125M parameters), which is substantially smaller than the Qwen3.5-9B and Gemma-4-26B-A4B-NVFP4 generators and screens text without per-sentence LLM calls; using a smaller downstream model for inference is also supported in recent low-resource classification work (Chen et al., 2024; Pecher et al., 2026). Because PII criteria can depend on context, our method uses about ten labeled examples to synthesize training data for a domain-specific classifier rather than requiring organizations to adopt a context-independent definition unchanged (Narayanan and Shmatikov, 2010).

8 Conclusion

We developed rubric-guided contrastive data synthesis for training a lightweight PII text detector from a small labeled seed set. Across SPeDaC1 and SPY, it improved classification performance over four baselines in all four Qwen/Gemma settings and over the Self-Instruct-style baseline in three of four settings.

Limitations

This work has several limitations. First, the evaluation covers English sentence-level classification on only SPeDaC1 and SPY; token-level extraction and multilingual generalization remain untested. Second, although our method outperforms the evaluated synthesis baselines, it does not reach a RoBERTa trained on the official training split. Third, our method depends on the performance of the LLM and does not guarantee the complete correctness of the induced rubrics or synthetic labels.

Our human evaluation used only two unblinded author raters and small samples per setting, and perceived rubric general applicability remained rater-dependent, especially for Qwen. Fourth, the parameters of our method are not optimized per dataset: 25 rubrics and 2,000 synthetic examples are common fixed settings, and both ablations were non-monotonic. Gemma also produced more highly similar rubric pairs than Qwen, so the induced rubric set is not guaranteed to be semantically non-redundant. Finally, this work does not guarantee compliance with legal definitions of PII or with each organization’s policy for handling personal information.

Ethical Considerations

This work addresses the generation of training data for privacy-related text classifiers. We do not use non-public PII such as private user logs, customer data, medical records, or internal documents. For evaluation, we use public benchmarks and synthetic data. When synthetic data is released or shared in real-world operation, additional human review, near-duplicate inspection, and confirmation of each organization’s policy for handling personal information are necessary.

References

- Jiaao Chen, Derek Tam, Colin Raffel, Mohit Bansal, and Diyi Yang. 2023. [An Empirical Survey of Data Augmentation for Limited Data Learning in NLP](#). *Transactions of the Association for Computational Linguistics*, 11:191–211.
- Lingjiao Chen, Matei Zaharia, and James Zou. 2024. [FrugalGPT: How to use large language models while reducing cost and improving performance](#). *Transactions on Machine Learning Research*, 2024.
- Daniele Faraglia. 2012. [Faker](#).
- Gaia Gambarelli, Aldo Gangemi, and Rocco Tripodi. 2023. [Is Your Model Sensitive? SPEDAC: A New Resource for the Automatic Classification of Sensitive Personal Data](#). *IEEE Access*, 11:10864–10880.
- Kilem Li Gwet. 2008. [Computing inter-rater reliability and its variance in the presence of high agreement](#). *British Journal of Mathematical and Statistical Psychology*, 61(1):29–48.
- Thomas Hartvigsen, Saadia Gabriel, Hamid Palangi, Maarten Sap, Dipankar Ray, and Ece Kamar. 2022. [ToxiGen: A Large-Scale Machine-Generated Dataset for Adversarial and Implicit Hate Speech Detection](#). In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 3309–3326, Dublin, Ireland. Association for Computational Linguistics.
- Jie Huang, Hanyin Shao, and Kevin Chen-Chuan Chang. 2022. [Are Large Pre-Trained Language Models Leaking Your Personal Information?](#) In *Findings of the Association for Computational Linguistics: EMNLP 2022*, pages 2038–2047, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.
- Ruibo Liu, Jerry Wei, Fangyu Liu, Chenglei Si, Yanzhe Zhang, Jinmeng Rao, Steven Zheng, Daiyi Peng, Diyi Yang, Denny Zhou, and Andrew M. Dai. 2024. [Best Practices and Lessons Learned on Synthetic Data](#). *Preprint*, arXiv:2404.07503.
- Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. [RoBERTa: A Robustly Optimized BERT Pretraining Approach](#). *Preprint*, arXiv:1907.11692.
- Nils Lukas, Ahmed Salem, Robert Sim, Shruti Tople, Lukas Wutschitz, and Santiago Zanella-Béguelin. 2023. [Analyzing Leakage of Personally Identifiable Information in Language Models](#). In *2023 IEEE Symposium on Security and Privacy (SP)*, pages 346–363. IEEE Computer Society.
- Rei Minamoto, Yusuke Oda, and Daisuke Kawahara. 2026. [Detecting Sensitive Personal Information in Japanese Pre-Training Corpora for Large Language Models](#). In *Findings of the Association for Computational Linguistics: ACL 2026*, pages 13341–13354, San Diego, California, United States. Association for Computational Linguistics.
- Kushagra Mishra, Harsh Pagare, Ranjeet Bidwe, and Sashikala Mishra. 2024. [Synthetic Dataset for PII Detection and Anonymization in Financial Documents](#).
- Arvind Narayanan and Vitaly Shmatikov. 2010. [Myths and fallacies of “personally identifiable information”](#). *Communications of the ACM*, 53(6):24–26.
- Bekelu Negash, Alan Katz, Christine J. Neilson, Moniruzzaman Moni, Marcello Nesca, Alexander Singer, and Jennifer E. Enns. 2023. [De-identification of Free Text Data containing Personal Health Information: A Scoping Review of Reviews](#). *International Journal of Population Data Science*, 8(1).
- Branislav Pecher, Jan Cegin, Robert Belanec, Ivan Srba, Jakub Simko, and Maria Bielikova. 2026. [Better as generators than classifiers: Leveraging LLMs and synthetic data for low-resource multilingual classification](#). In *Findings of the Association for Computational Linguistics: EACL 2026*, pages 2840–2857, Rabat, Morocco. Association for Computational Linguistics.
- Ildikó Pilán, Pierre Lison, Lilja Øvrelid, Anthi Papadopoulou, David Sánchez, and Montserrat Batet. 2022. [The Text Anonymization Benchmark \(TAB\): A Dedicated Corpus and Evaluation Framework for](#)

[Text Anonymization](#). *Computational Linguistics*, 48(4):1053–1101.

Joan Santoso, Patrick Sutanto, Billy Cahyadi, and Esther Setiawan. 2024. [Pushing the Limits of Low-Resource NER Using LLM Artificial Data Generation](#). In *Findings of the Association for Computational Linguistics: ACL 2024*, pages 9652–9667, Bangkok, Thailand. Association for Computational Linguistics.

Maksim Savkin, Timur Ionov, and Vasily Kononov. 2025. [SPY: Enhancing Privacy with Synthetic PII Detection Dataset](#). In *Proceedings of the 2025 Conference of the Nations of the Americas Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 4: Student Research Workshop)*, pages 236–246, Albuquerque, USA. Association for Computational Linguistics.

Yizhong Wang, Yeganeh Kordi, Swaroop Mishra, Alisa Liu, Noah A. Smith, Daniel Khashabi, and Hannaneh Hajishirzi. 2023. [Self-Instruct: Aligning Language Models with Self-Generated Instructions](#). In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 13484–13508, Toronto, Canada. Association for Computational Linguistics.

Jiacheng Ye, Jiahui Gao, Qintong Li, Hang Xu, Jiangtao Feng, Zhiyong Wu, Tao Yu, and Lingpeng Kong. 2022. [ZeroGen: Efficient Zero-shot Learning via Dataset Generation](#). In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 11653–11669, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

A Examples of induced rubrics

Table 7 shows representative examples of rubrics actually induced under the final setting. These are English rubrics generated at run time and are shown without translation. Each row presents a Sensitive condition and a contrasting Non-sensitive condition from the same induced rubric.

B Additional baseline checks

To test dependence on RoBERTa, we trained a fixed TF-IDF + linear SVM on each synthetic dataset. Table 8 shows the result. Our data had higher mean Macro F1 than the Self-Instruct-style baseline in two of four settings (Qwen/SPY and Gemma/SPeDaC1), and both paired differences excluded zero. The other two confidence intervals included zero, so the SVM results did not show an advantage in all four settings. Training the same SVM on the full official split yielded Macro F1 of 0.958 on SPeDaC1 and 0.932 on SPY.

As an additional SPY comparison, a fixed text-only regex baseline covering email addresses, URLs, IP addresses, phone numbers, SSNs, Luhn-valid payment cards, long alphanumeric IDs, and @handles obtained Macro F1 0.719 and MCC 0.449. It used no PII-type metadata. This is below our RoBERTa results (0.842/0.692 with Qwen

and 0.860/0.721 with Gemma, Macro F1/MCC), which is consistent with the task requiring more than surface identifier formats. An off-the-shelf Microsoft Presidio detector with a spaCy backend, configured to predict Sensitive when it detected at least one entity at confidence $\geq .5$, obtained Macro F1/MCC of .510/.025 on SPeDaC1 and .702/.408 on SPY. It read only the sentence text, not SPY’s PII-type metadata. The low SPeDaC1 score is consistent with that dataset requiring disclosure judgments beyond the presence of a recognized entity type; Presidio also lacks a username recognizer, one of SPY’s seven annotated types.¹

C Statistical inference

For each method comparison, we estimated uncertainty using 10,000 two-level paired bootstrap samples. At the first level, we resampled test sentences with replacement within each gold-label class. At the second level, we resampled the three seed-pool runs with replacement. We applied the same sampled sentences and runs to both methods, thereby preserving the pairing. We report percentile 95% confidence intervals for between-method differences in Macro F1 and MCC. For Macro F1, we applied Holm correction separately to the 16 main-

¹<https://github.com/microsoft/presidio>

Dataset	Criterion	Sensitive condition	Non-sensitive condition
SPeDaC1	V001_C1	The text explicitly discloses specific details regarding the speaker’s medical diagnosis (AML), treatment history (chemo), and personal life (disability status, relationship status), where the speaker is the direct subject of this confidential information.	The text discusses general public matters such as urban planning regulations, societal myths about relationship dynamics, or broad political issues (Brexit) without revealing the speaker’s private medical history, earnings, or specific personal relationships.
SPeDaC1	V002_C1	The text contains specific disclosures of personally identifiable information (such as a father’s medical diagnosis and age, the speaker’s specific disability, occupation, earnings, savings, and personal relationships) or reveals the speaker’s active stance against the collection of such private data, establishing a direct relationship of ownership or confidentiality over these details.	The text discusses general topics, medical proverbs, or public administrative events (such as building height proposals or abstract observations about disease diagnosis) where the information is either publicly available, lacks specific personal identifiers, or is framed as general commentary by the speaker rather than a revelation of private facts about themselves or others.
SPeDaC1	V003_C1	The text discloses direct, verifiable details regarding a specific individual’s private health status (diagnosis, medication dosage, or surgical procedures), positioning the speaker as the patient directly affected by or describing the event.	The text discusses abstract concepts, general medical statistics, or public policy matters without revealing a specific individual’s identifiable health data, positioning the speaker as an observer, commentator, or official entity discussing general issues.
SPY	V001_C1	The text contains specific, personally identifiable information (PII) such as a mobile app username, a full name, and a phone number, where the speaker identifies themselves as the direct user or owner of that data/identity.	The text discusses general business or organizational issues involving names or identifiers that are either job-related titles, generic references, or external entities (clients/vendors) with whom the speaker has a professional relationship rather than personal ownership of the data.
SPY	V002_C1	The text reveals direct identifiers (specific email addresses or usernames) owned by the speaker themselves, creating a link between the personal identity and the shared data or contact method.	The text contains email addresses or numbers that are not owned by the speaker (e.g., a client’s decision, a union representative, or a symptom report) or lacks direct personal identifiers of the speaker, thus removing the self-referential link to sensitive identity.
SPY	V003_C1	The text contains the speaker’s personally identifiable health insurance identifier (a specific 9-digit number) or precise residential address, directly implicating the speaker’s identity and private living situation.	The text discusses abstract business outcomes (client decisions), general medical symptoms (bloating, diarrhea), or project management issues (deadlines, delays) without revealing the speaker’s specific identity, exact location, or unique private identifiers.

Table 7: Representative examples of rubrics induced under the final setting. The condition statements in the table are the English rubrics actually generated in the final run of Qwen3.5-9B, shown verbatim.

table comparisons and the 52 ablation comparisons. We did not use validation or test labels to select a synthesis configuration.

Table 9 reports all main comparisons. Tables 10 and 11 report Macro F1 and MCC for all 46 ablation conditions. Tables 12–14 report all 52 paired ablation comparisons.

Generator Dataset	Self-Instruct-style		Ours	
	Macro F1	MCC	Macro F1	MCC
Qwen SPeDaC1	.735±.021	.487±.030	.729±.041	.517±.057
Qwen SPY	.619±.032	.265±.078	.663±.036	.361±.065
Gemma SPeDaC1	.710±.030	.464±.032	.770±.019	.571±.026
Gemma SPY	.661±.044	.328±.091	.643±.031	.350±.032

Table 8: Fixed TF-IDF + linear-SVM results over three seed pools. The classifier uses word 1–2-grams, character-within-word 3–5-grams, $C=1$, and balanced class weights without validation or test tuning. Ours minus Self-Instruct-style baseline Macro-F1 confidence intervals are $[-.045, .026]$, $[.016, .072]$, $[.027, .098]$, and $[-.051, .014]$ in row order. Two of the four intervals exclude zero.

D Pair-generation ablation with matched rubric counts

The main generation-structure ablation changes the rubric count because a single generated sentence contributes one example whereas a contrastive call contributes two. To isolate paired generation more directly, we additionally fixed both methods to 25 rubrics and 1,000 examples. Table 12(c) shows that paired generation improves SPY on both metrics; on SPeDaC1 the MCC interval excludes zero but the Macro-F1 interval includes zero. These results indicate that pair generation contributes to performance, although the effect was not supported by both metrics on both datasets.

E Human evaluation protocol and results

From each of the 12 dataset-generator-seed-pool settings, we sampled five induced rubrics and eight Sensitive/Non-sensitive pairs that passed LLM verification, yielding 60 rubrics and 96 pairs. Two authors rated all items independently in an unblinded interface. Disagreements were not adjudicated. Rubrics and generated pairs were rated in separate tasks. For a rubric, raters viewed the

Generator	Dataset	Baseline	Δ Macro F1 [95% CI]	Holm p	Δ MCC [95% CI]
Qwen	SPeDaC1	Duplicate	+9.54 [+2.46, +14.31]	.0032	+.191 [+0.059, +.283]
Qwen	SPeDaC1	Paraphrase	+9.42 [+2.84, +14.69]	.0032	+.191 [+0.056, +.295]
Qwen	SPeDaC1	One-shot	+15.01 [+10.56, +20.98]	.0032	+.299 [+0.210, +.422]
Qwen	SPeDaC1	Few-shot	+7.91 [+4.57, +11.40]	.0032	+.104 [+0.059, +.145]
Qwen	SPY	Duplicate	+16.47 [+12.43, +21.29]	.0032	+.318 [+0.248, +.400]
Qwen	SPY	Paraphrase	+16.85 [+11.22, +23.54]	.0032	+.316 [+0.231, +.416]
Qwen	SPY	One-shot	+14.98 [+12.37, +17.87]	.0032	+.256 [+0.198, +.312]
Qwen	SPY	Few-shot	+13.87 [+8.15, +19.14]	.0032	+.223 [+0.128, +.315]
Gemma	SPeDaC1	Duplicate	+15.18 [+11.23, +19.93]	.0032	+.297 [+0.228, +.376]
Gemma	SPeDaC1	Paraphrase	+15.34 [+11.58, +18.88]	.0032	+.300 [+0.232, +.362]
Gemma	SPeDaC1	One-shot	+12.56 [+9.91, +15.16]	.0032	+.242 [+0.191, +.293]
Gemma	SPeDaC1	Few-shot	+17.30 [+13.67, +20.66]	.0032	+.273 [+0.217, +.325]
Gemma	SPY	Duplicate	+18.23 [+10.55, +26.43]	.0032	+.348 [+0.209, +.495]
Gemma	SPY	Paraphrase	+18.02 [+9.06, +28.80]	.0032	+.334 [+0.170, +.512]
Gemma	SPY	One-shot	+14.17 [+10.09, +18.02]	.0032	+.227 [+0.122, +.321]
Gemma	SPY	Few-shot	+25.16 [+15.35, +39.08]	.0032	+.357 [+0.232, +.479]

Table 9: All 16 paired comparisons of our method against the four submitted baselines. Macro-F1 differences are percentage points and MCC differences are decimals. Holm p denotes the p value adjusted across the 16 Macro-F1 comparisons.

(a) Generation structure (Qwen)				(b) Rubric count (Qwen)				(c) Pair vs. single, 25 rubrics (Qwen)			
Dataset	Condition	Macro F1	MCC	Dataset	Rubrics	Macro F1	MCC	Dataset	Mode	Macro F1	MCC
SPeDaC1	Single pair	63.72 ± 8.99	.320 ± .182	SPeDaC1	1	82.43 ± 1.06	.664 ± .009	SPeDaC1	Pair	83.21 ± 4.68	.687 ± .074
SPeDaC1	Rubric→single	79.85 ± 1.43	.608 ± .026	SPeDaC1	5	80.14 ± 4.90	.637 ± .070	SPeDaC1	Single	79.81 ± 1.71	.600 ± .033
SPeDaC1	Rubric→pair	82.02 ± 5.83	.669 ± .087	SPeDaC1	25	83.21 ± 4.68	.687 ± .074	SPY	Pair	84.54 ± 0.58	.705 ± .009
SPY	Single pair	72.11 ± 6.95	.455 ± .135	SPeDaC1	50	84.12 ± 4.65	.705 ± .075	SPY	Single	74.21 ± 4.92	.499 ± .083
SPY	Rubric→single	72.80 ± 6.22	.463 ± .122	SPY	1	76.25 ± 6.15	.545 ± .107				
SPY	Rubric→pair	83.25 ± 2.13	.672 ± .033	SPY	5	81.94 ± 1.49	.648 ± .034				
				SPY	25	84.54 ± 0.58	.705 ± .009				
				SPY	50	79.45 ± 3.82	.593 ± .078				

Table 10: Complete metrics for the generation-structure and rubric-count ablations and the pair-versus-single comparison with 25 rubrics in both conditions. Values are mean±sample standard deviation over three seed pools; Macro F1 is a percentage.

(a) Seed-pool size (Qwen)				(b) Seed-pool size (Gemma)				(c) Synthetic examples (Qwen)			
Dataset	Pool	Macro F1	MCC	Dataset	Pool	Macro F1	MCC	Dataset	Examples	Macro F1	MCC
SPeDaC1	4	72.92 ± 17.39	.529 ± .243	SPeDaC1	4	79.16 ± 11.10	.620 ± .189	SPeDaC1	500	85.17 ± 4.05	.713 ± .070
SPeDaC1	10	85.96 ± 2.65	.725 ± .047	SPeDaC1	10	88.12 ± 5.04	.766 ± .098	SPeDaC1	1,000	84.63 ± 2.70	.707 ± .046
SPeDaC1	20	81.07 ± 2.16	.639 ± .040	SPeDaC1	20	90.84 ± 1.90	.817 ± .038	SPeDaC1	2,000	83.15 ± 4.69	.672 ± .089
SPeDaC1	40	90.87 ± 0.30	.818 ± .006	SPeDaC1	40	92.49 ± 0.68	.850 ± .014	SPeDaC1	3,000	78.89 ± 3.80	.620 ± .056
SPY	4	63.94 ± 14.65	.340 ± .205	SPY	4	75.84 ± 9.80	.529 ± .178	SPeDaC1	4,000	79.12 ± 5.84	.623 ± .097
SPY	10	83.04 ± 0.98	.666 ± .014	SPY	10	85.06 ± 1.87	.702 ± .037	SPeDaC1	5,000	80.65 ± 1.96	.630 ± .034
SPY	20	82.71 ± 5.46	.663 ± .110	SPY	20	84.14 ± 5.21	.692 ± .101	SPY	500	80.63 ± 3.78	.616 ± .079
SPY	40	85.29 ± 3.07	.711 ± .058	SPY	40	86.75 ± 0.90	.737 ± .018	SPY	1,000	83.57 ± 1.18	.681 ± .022
								SPY	2,000	84.19 ± 2.24	.692 ± .043
								SPY	3,000	84.59 ± 1.89	.696 ± .035
								SPY	4,000	84.81 ± 2.82	.698 ± .055
								SPY	5,000	83.07 ± 2.31	.664 ± .042

Table 11: Complete metrics for the seed-pool-size and synthetic-training-size ablations. Values are mean±sample standard deviation over three seed pools; Macro F1 is a percentage.

Dataset	Comparison (left—right)	Δ Macro F1 [95% CI]	Holm p	Δ MCC [95% CI]
(a) Generation structure (Qwen)				
SPeDaC1	Rubric→pair — single-stage pair	+18.30 [+13.22, +24.10]	.0104	+.349 [+0.228, +.489]
SPeDaC1	Rubric→pair — rubric→single	+2.16 [−4.34, +6.59]	1.0000	+.061 [−.043, +.137]
SPY	Rubric→pair — single-stage pair	+11.14 [+3.53, +17.11]	.0620	+.218 [+0.062, +.335]
SPY	Rubric→pair — rubric→single	+10.45 [+3.44, +17.76]	.0272	+.210 [+0.069, +.353]
(b) Rubric count (Qwen)				
SPeDaC1	25 — 1 rubrics	+0.78 [−3.54, +6.98]	1.0000	+.023 [−.040, +.110]
SPeDaC1	5 — 1 rubrics	−2.30 [−6.84, +4.17]	1.0000	−.027 [−.089, +.056]
SPeDaC1	25 — 5 rubrics	+3.08 [+2.12, +4.07]	.0104	+.050 [+0.031, +.069]
SPeDaC1	50 — 25 rubrics	+0.90 [−1.45, +3.74]	1.0000	+.018 [−.019, +.063]
SPY	25 — 1 rubrics	+8.29 [+2.25, +13.41]	.0528	+.160 [+0.055, +.250]
SPY	5 — 1 rubrics	+5.69 [−0.36, +11.27]	1.0000	+.103 [−.003, +.204]
SPY	25 — 5 rubrics	+2.60 [+0.98, +4.49]	.1044	+.058 [+0.020, +.100]
SPY	50 — 25 rubrics	−5.08 [−9.21, −1.68]	.0210	−.112 [−.195, −.044]
(c) Pair versus single with 25 rubrics (Qwen)				
SPeDaC1	Pair — single	+3.40 [−0.20, +7.11]	1.0000	+.087 [+0.033, +.142]
SPY	Pair — single	+10.32 [+4.78, +15.64]	.0104	+.206 [+0.110, +.296]

Table 12: Complete paired comparisons for the generation-structure and rubric-count ablations and the pair-versus-single comparison with 25 rubrics in both conditions. Macro-F1 differences are percentage points and MCC differences are decimals. Holm p is adjusted across all 52 ablation Macro-F1 comparisons.

(a) Qwen					(b) Gemma				
Data	Comparison	Δ F1 [95% CI]	Holm p	Δ MCC [95% CI]	Data	Comparison	Δ F1 [95% CI]	Holm p	Δ MCC [95% CI]
<i>Seed-pool size (left—right)</i>					<i>Seed-pool size (left—right)</i>				
SPeD Pool 4—10	−13.05 [−31.63, +0.34]	1.0000	−.196 [−.444, −.008]		SPeD Pool 4—10	−8.96 [−15.10, −0.04]	1.0000	−.146 [−.256, −.002]	
SPeD Pool 20—10	−4.90 [−9.79, −1.48]	.0148	−.086 [−.170, −.021]		SPeD Pool 20—10	+2.72 [−2.29, +9.26]	1.0000	+.051 [−.048, +.179]	
SPeD Pool 40—10	+4.91 [+2.45, +7.17]	.0104	+.093 [+0.050, +.132]		SPeD Pool 40—10	+4.37 [+0.63, +10.38]	.0528	+.085 [+0.012, +.201]	
SPY Pool 4—10	−19.11 [−34.45, −8.46]	.0104	−.326 [−.528, −.167]		SPY Pool 4—10	−9.22 [−21.36, −0.63]	.6263	−.173 [−.394, −.012]	
SPY Pool 20—10	−0.33 [−5.91, +3.90]	1.0000	−.004 [−.115, +.080]		SPY Pool 20—10	−0.92 [−5.84, +3.49]	1.0000	−.010 [−.106, +.079]	
SPY Pool 40—10	+2.25 [−0.19, +4.78]	1.0000	+.044 [−.008, +.095]		SPY Pool 40—10	+1.69 [−0.25, +3.53]	1.0000	+.035 [−.004, +.072]	
<i>Proposed—duplicate at each pool size</i>					<i>Proposed—duplicate at each pool size</i>				
SPeD Pool 4	+10.67 [−5.88, +21.18]	1.0000	+.232 [+0.043, +.354]		SPeD Pool 4	+16.91 [+10.19, +24.01]	.0104	+.324 [+0.158, +.473]	
SPeD Pool 10	+15.28 [+8.37, +22.34]	.0104	+.294 [+0.176, +.404]		SPeD Pool 10	+17.43 [+14.76, +20.63]	.0104	+.334 [+0.273, +.410]	
SPeD Pool 20	−1.80 [−4.75, +0.81]	1.0000	−.020 [−.072, +.029]		SPeD Pool 20	+7.97 [+6.10, +9.74]	.0104	+.158 [+0.119, +.194]	
SPeD Pool 40	+4.58 [+1.13, +7.37]	.2016	+.089 [+0.016, +.147]		SPeD Pool 40	+6.20 [+2.90, +8.78]	.0104	+.121 [+0.053, +.173]	
SPY Pool 4	−1.53 [−5.64, +2.97]	1.0000	+.029 [−.050, +.103]		SPY Pool 4	+10.38 [+6.74, +14.04]	.0104	+.218 [+0.132, +.303]	
SPY Pool 10	+15.29 [+10.79, +20.44]	.0104	+.300 [+0.217, +.391]		SPY Pool 10	+17.31 [+11.95, +24.28]	.0104	+.335 [+0.234, +.465]	
SPY Pool 20	+11.97 [+6.93, +18.50]	.0104	+.239 [+0.138, +.370]		SPY Pool 20	+13.39 [+8.95, +18.51]	.0104	+.268 [+0.185, +.368]	
SPY Pool 40	+3.99 [−1.37, +10.20]	1.0000	+.082 [−.020, +.203]		SPY Pool 40	+5.45 [+1.83, +8.80]	.0780	+.108 [+0.037, +.175]	

Table 13: Complete paired comparisons, separated by generator, for the seed-pool-size ablation and for our method versus the duplicate baseline at each seed-pool size. SPeD denotes SPeDaC1. Holm p is adjusted across all 52 ablation Macro-F1 comparisons.

labeled examples used to induce it and answered two questions. The first asked whether the rubric explained why the displayed examples received their labels. The second asked whether the rubric stated a general rule applicable beyond those examples, considering specificity, non-circularity, substance, and scope. The seed examples remained visible, so this was an unblinded human assessment

rather than a measurement of held-out classification accuracy. Responses were *yes/partial/no*, with *uncertain* for an unjudgeable item. For generated pairs, raters viewed the rubric and answered two *yes/no* questions for each sentence: whether its label was valid under the original dataset definition and whether the English sentence was natural. Machine-generated Japanese translations

Examples	Δ Macro F1 [95% CI]	Holm p	Δ MCC [95% CI]
(a) SPeDaC1 (Qwen)			
500	+2.02 [-0.11, +4.70]	1.0000	+0.041 [+0.002, +0.086]
1,000	+1.48 [-4.17, +6.37]	1.0000	+0.035 [-0.071, +0.121]
3,000	-4.26 [-6.38, -1.99]	.0148	-.052 [-0.094, -.015]
4,000	-4.03 [-8.00, +1.44]	1.0000	-.050 [-0.111, +0.038]
5,000	-2.50 [-5.49, +0.11]	1.0000	-.042 [-0.107, +0.019]
(b) SPY (Qwen)			
500	-3.56 [-7.06, +0.30]	1.0000	-.076 [-0.153, +0.010]
1,000	-0.62 [-4.22, +2.57]	1.0000	-.011 [-0.082, +0.051]
3,000	+0.40 [-1.61, +2.88]	1.0000	+0.005 [-0.036, +0.055]
4,000	+0.62 [-1.68, +3.24]	1.0000	+0.007 [-0.045, +0.063]
5,000	-1.12 [-3.01, +1.03]	1.0000	-.028 [-0.067, +0.017]

Table 14: Complete synthetic-training-size comparisons. Each difference is the score at the displayed training size minus the score at 2,000 examples. Holm p denotes the p value adjusted across all 52 ablation Macro-F1 comparisons.

Evaluation question	Qwen	Gemma
(a) SPeDaC1		
Explains labels of displayed		
seed examples	.867/.933 (.780)	1.000/.933 (.931)
General applicability	.467/.867 (.486)	.867/1.000 (.858)
Sensitive label validity	1.000/1.000 (1.000)	.958/1.000 (.955)
Non-sensitive label validity	.917/.958 (.953)	.958/.958 (1.000)
Sensitive naturalness	.833/1.000 (.803)	.958/1.000 (.957)
Non-sensitive naturalness	.750/1.000 (.680)	.875/1.000 (.858)
(b) SPY		
Explains labels of displayed		
seed examples	.933/.867 (.780)	.800/.733 (.594)
General applicability	.400/.867 (.219)	.733/.733 (.669)
Sensitive label validity	.875/1.000 (.851)	.875/1.000 (.905)
Non-sensitive label validity	.792/.913 (.824)	.957/.905 (.942)
Sensitive naturalness	.750/1.000 (.680)	.958/1.000 (.957)
Non-sensitive naturalness	.833/1.000 (.803)	.833/1.000 (.803)

Table 15: Human evaluation by dataset and generator. For each dataset-generator combination, rubric questions cover 15 rubrics. Each sentence question covers 24 Sensitive or Non-sensitive sentences, one from each of 24 generated pairs. Values are rater A/B *yes* rates after excluding that rater’s uncertain responses; Gwet AC1 is in parentheses.

(nvidia/Qwen3.6-35B-A3B-NVFP4, temperature 0) were displayed only as reading aids; ratings used the authoritative English text. Raters could again select *uncertain*.

Table 15 reports each rater’s *yes* rate after excluding that rater’s uncertain responses. Gwet’s first-order agreement coefficient (AC1) measures agreement between the two raters after accounting for agreement expected by chance (Gwet, 2008). A value of 1 indicates perfect agreement and 0 indicates chance-level agreement. We calculated AC1 using the three non-uncertain rubric responses and the two non-uncertain sentence responses. Items marked *uncertain* by either rater were excluded.

Most sampled rubrics were judged to explain why the displayed seed examples received their labels. Most sentences in pairs that passed LLM

verification were judged to have valid labels and to be natural. Gemma received higher general-applicability and naturalness ratings than Qwen and also achieved higher downstream Macro F1 on both datasets. Across the 12 dataset-generator-seed-pool settings, mean human naturalness ratings and downstream Macro F1 had Spearman $\rho = .649$. This is a descriptive association and does not establish that naturalness causes better classification performance. Among accepted pairs, the proportions for which both sentence labels were judged valid were A/B=.915/.957 on SPeDaC1 and .745/.909 on SPY. Because candidates rejected by LLM verification were not sampled, these proportions estimate the precision of LLM verification but not its recall; they do not measure how many valid candidates were rejected.

F Examples generated per rubric and similarity between rubrics

The target allocation was 1,000 pairs over 25 rubrics, or 40 pairs per rubric. Qwen reached 40 per rubric in five of six runs; one run induced 24 rubrics and allocated 41–42 pairs. Gemma reached 40 per rubric in four of six runs; two SPY runs stopped at 983 and 993 pairs, with per-rubric ranges of 23–40 and 33–40. There were no normalized exact duplicate rubrics. For each run, we embedded the rubric text with all-MiniLM-L6-v2 and compared all 300 pairs of rubrics (276 in the 24-rubric run). At cosine similarity $\geq .95$, Qwen had 1–15 pairs per run and Gemma 36–166; at $\geq .98$, Qwen had none and Gemma 1–25. Thus, generation was distributed mostly evenly across rubrics. Qwen produced fewer highly similar rubric pairs than Gemma at both thresholds.

G Prompts used

In the main condition, we used a 3+3 subset of support examples for rubric induction and the entire 5+5 seed pool for contrastive pair generation and verification. Rubric induction was run 25 times, with a maximum of one output per run. Below we show the prompt templates actually used in the final condition. In the implementation, the intermediate representation is called a *contrast card*, which we describe as a rubric in the main text.

G.1 Rubric induction prompt

You are preparing synthetic training data for a binary privacy text classification task.

Use only the labeled support examples below. Do not use dataset names, benchmark descriptions, external privacy policies, or prior assumptions.

Sensitive examples:
{positive_examples}

Non-sensitive examples:
{negative_examples}

Task:
Induce up to {k} contrast cards. Each contrast card describes one label-defining difference that is supported by the labeled examples.

A valid contrast card must:

- cite at least one Sensitive support example and one Non-sensitive support example;
- describe the Sensitive side by stating what information in the cited example supports the Sensitive label, whose or what information it is, and how the text's speaker or source is related to that information;
- describe the comparable Non-sensitive side using the same three points: what information in the cited example is closest to the Sensitive-side evidence or is absent, whose or what it is about, and how the text's speaker or source relation differs.

Return only JSON in this format:

```
[
  {
    "card_id": "C1",
    "sensitive_condition": "...",
    "nonsensitive_condition": "...",
    "support_sensitive": ["S1", "S3"],
    "support_nonsensitive": ["N2", "N5"]
  }
]
```

G.2 Self-Instruct-style baseline

The Self-Instruct-style baseline starts with the following six generic human-written instructions, identical for both datasets: (1) whether a sentence discloses private information about the speaker; (2) whether a reader would learn something the speaker would ordinarily keep private; (3) whether a sentence is about the speaker or someone else; (4) whether it describes a specific person or makes a general statement; (5) whether it is safe to publish without the speaker's consent; and (6) whether it carries private information or only mentions a topic to which private information could belong.

At each iteration, the prompt contains the six seed instructions and up to two previously accepted model-generated instructions sampled from the instruction pool. The model proposes two new instructions. We add an instruction to the pool only if its ROUGE-L F1 is below .7 against every instruction already in the pool. Accepted instructions may then be sampled in later iterations.

You are extending a list of instructions for a binary sentence-level privacy classification task.

Each instruction names one aspect of deciding whether an English sentence is Sensitive or Non-sensitive.

Existing instructions:
{six seed and up to two accepted model-generated instructions}

Write 2 new instructions for aspects that the list does not already cover.

Requirements:

- Each instruction must start with "Decide whether a sentence".
- Name a concrete aspect. Do not restate an existing instruction in other words.
- Describe what to look for, not how to phrase a sentence.
- Do not mention datasets, benchmarks, prompts, or label names other than Sensitive and Non-sensitive.

Return only JSON: {"instructions": ["...", "..."]}

For each sentence, one instruction is selected from the instruction pool and a Sensitive or Non-sensitive target label is specified. The same original 5+5 labeled seed examples as the proposed method are supplied as few-shot demonstrations. Accepted generated sentences are not added to the few-shot demonstrations in later sentence-generation prompts. Each labeled sentence is generated separately rather than as part of a contrastive pair.

You are preparing synthetic training data for a binary sentence-level privacy classification task.

Use only the labeled demonstrations below to infer the distinction between Sensitive and Non-sensitive.

Sensitive demonstrations:
{five Sensitive seeds}

Non-sensitive demonstrations:
{five Non-sensitive seeds}

Instruction for this item:
{sampled instruction}

The correct label for the sentence you write is fixed in advance: {label}

Write one new English sentence whose correct label is {label} and which is relevant to the instruction above. Write a natural sentence, do not copy or closely paraphrase a demonstration, and do not state or explain the label.

Return only JSON: {"text": "..."}.

During LLM verification, the model receives the original 5+5 seed examples, the generated sentence, and its target label. It checks label correctness, naturalness, and whether the sentence copies a seed. Accepted sentences must also pass normalized exact-match filtering and have ROUGE-L < .95 against all seed and previously accepted sentences. The target is 1,000 accepted sentences per label.

G.3 Contrastive pair generation prompt

You are generating synthetic training data for a binary privacy text classification task.

Use only the labeled support examples and the contrast card below.

Sensitive examples:
{positive_examples}

Non-sensitive examples:
{negative_examples}

Contrast card:
{contrast_card}

Task:

```

Generate one matched pair of new texts:
- one text whose correct label is Sensitive;
- one text whose correct label is Non-sensitive.

Requirements:
- Keep the two texts as similar as possible in topic, style,
  length, and surface cue.
- Change only the minimum relation, role, source, or context
  needed to move from one label to the other under the
  contrast card.
- Make both texts natural and compatible with the support
  examples.
- Do not copy any support example.
- Do not mention dataset names, benchmark names, labels, or
  explanations inside the generated texts.

Return only JSON:
{
  "sensitive_text": "...",
  "nonsensitive_text": "..."
}

```

G.4 Contrastive pair verification prompt

```

You are verifying synthetic data for a binary privacy text
classification task.

Use only the labeled support examples and the contrast card
below.

Sensitive examples:
{positive_examples}

Non-sensitive examples:
{negative_examples}

Contrast card:
{contrast_card}

Generated pair:
Sensitive candidate:
<<<{sensitive_text}>>>

Non-sensitive candidate:
<<<{nonsensitive_text}>>>

Check the pair.

Return only JSON:
{
  "sensitive_label_correct": true,
  "nonsensitive_label_correct": true,
  "matched_pair": true,
  "copied_from_support": false,
  "accept": true,
  "reason": "short reason"
}

```

H Scope of support examples

In the main condition, we use a 3+3 subset of support examples for rubric induction and the entire 5+5 seed pool for contrastive pair generation and verification. Using the same 3+3 for generation and verification yielded, for Qwen with $n = 1000$, Macro F1 of SPeDaC1 84.65 ± 2.85 and SPY 83.81 ± 2.93 . The corresponding main-condition values were 84.63 ± 2.70 and 83.57 ± 1.18 . Seed variance was large, and one SPY run terminated at 986/1000 examples. We therefore retained the main design: inducing local rubrics from 3+3 and using the full 5+5 seed pool for contrastive pair generation and verification.