Writerslogic at PAN 2026: Process over Content for Robust Detection under Domain Shift

Notebook for the PAN Lab at CLEF 2026

David Condrey[1]

[1]WritersLogic Inc, San Diego, CA, USA

Abstract

We describe the Writerslogic systems for three PAN@CLEF 2026 shared tasks—Reasoning Trajectory Detection, Voight-Kampff Generative AI Detection, and Multi-Author Writing Style Analysis—unified by a shared analytical framework: feature robustness under distribution shift is governed by *support overlap* between training and test distributions, not by training-set effect size. This yields a taxonomy (domain-anchored, domain-portable, domain-invariant) that explains why generator-specific features die under domain shift while vocabulary fingerprints (hapax ratio, Yule's K, Heaps' exponent), compression measures, and character n-grams survive. On Reasoning Trajectory Detection, where training was 100% mathematics and 84% of test was unseen domains, the framework guided system design to 1st place in source detection (0.85 macro F1 via Opus–Sonnet agreement) and 3rd place in safety classification (0.66 macro F1 via query-refusal decomposition). For Voight-Kampff, we built a calibrated ensemble of DeBERTa-v2 (ONNX), multi-seed LightGBM with 44 domain-portable stylometric features, and SVM on n-gram TF-IDF, combined via learned stacking with isotonic calibration; the best configuration achieved 0.891 on the PAN 2026 test set with exceptionally balanced sub-metrics (0.853–0.902 across all evaluation dimensions), confirming that domain-portable feature design transfers across genres. For Multi-Author Writing Style Analysis, we describe a system fusing spectral clustering over character n-gram similarity graphs, normalized compression distance for local boundary detection, and SmolLM-135M perplexity for neural change-point detection; a platform mix-up meant our run never reached the official evaluation, so we report the design and its a priori predictions rather than measured results. Across all three tasks, features measuring *generation process* properties are designed to outperform features measuring *generated content* properties under domain shift. Code: https://github.com/dcondrey/voight-kampff-clef2026, https://github.com/dcondrey/trajectory-detection-clef2026.

Keywords

AI-generated text detection, domain shift, feature robustness, vocabulary fingerprints, safety classification, ensemble methods

1. Introduction

The standard recipe for a text classification shared task is: extract features, train a model, optimize on validation, submit. We followed this recipe on the PAN@CLEF 2026 Reasoning Trajectory Detection task [1, 2]. A LightGBM [3] classifier achieved 0.97 macro F1 on validation but 0.69 on test. An XLM-RoBERTa [4] safety classifier achieved 0.83 on validation but 0.45 on test.

These failures follow from a principle this paper develops: *feature robustness under distribution shift is governed by support overlap—whether the feature has non-trivial values in the test distribution—not by training-set effect size.* This yields a taxonomy (Section 3) and two consequences: the *validation illusion*, where IID validation actively selects non-transferable features; and the *granularity trap*, where fine-grained estimators degrade hierarchical performance until they exceed a computable accuracy threshold.

We participated in three PAN@CLEF 2026 tasks, applying this framework to guide system design:

1. **Reasoning Trajectory Detection** (Section 4): Source detection (1st place, 0.85 F1) and safety classification (3rd place, 0.66 F1). The framework was developed retrospectively on this task, where generator-specific features with 0% test fire rate explained the validation-test gap.

CLEF 2026 Working Notes, 21 – 24 September 2026, Jena, Germany

david@writerslogic.com (D. Condrey)

0009-0003-1849-2963 (D. Condrey)

2. **Voight-Kampff Generative AI Detection** (Section 5): Cross-genre AI text detection using a calibrated ensemble of DeBERTa-v2, LightGBM, and SVM (0.891 on PAN 2026 test, 0.979 on PAN 2025 test). Feature engineering was informed by the framework: all 44 features were designed to be domain-portable across genres.
3. **Multi-Author Writing Style Analysis** (Section 6): Paragraph-level style-change detection under adversarial mimicry, fusing spectral clustering over character n-gram graphs, normalized compression distance, and SmolLM-135M perplexity. Our run did not reach the official evaluation platform in time, so we hold no official results; we present the design and register a priori predictions as an untested application of the framework, not a result.

The unifying thesis is that features measuring *generation process* properties are more robust under domain shift than features measuring *generated content* properties.

2. Related Work

AI-generated text detection. Detection approaches fall into three families. Probability-based methods analyze statistical signatures: DetectGPT [5] uses log-probability curvature, GLTR [6] visualizes token-level rank distributions, and Binoculars [7] compares perplexity across models. Stylometric methods extract surface features and train classifiers [8, 9]. Fine-tuned transformer methods (RoBERTa [10], DeBERTa [11]) achieve high in-distribution accuracy but degrade under domain shift [12, 13, 14, 15]. Tavan and Najafi [16] demonstrated that heterogeneous ensembles outperform individual detectors. Our feature death taxonomy offers one explanation: mixing features from different robustness categories means surviving features compensate when others die.

Cross-domain generalization. Park et al. [12] propose domain generalization techniques for machine-generated text detection. Liu et al. [17] use spectral alignment for cross-domain transfer. Wang et al. [18] integrate lexical pair dispersion. Mobin and Islam [19] demonstrate inverse perplexity-weighted ensembles. Ramponi and Plank [20] observe that decomposition into shift-invariant subproblems can outperform direct domain adaptation, a principle our safety classification system instantiates.

Vocabulary richness as detection signal. Yule's K [21] measures vocabulary repetitiveness. Heaps' law [22, 23] captures vocabulary growth rate. Baayen [24] provides theoretical foundations; Tweedie and Baayen [25] establish within-author stability across genres. Faltýnek and Matlach [26] show hapax legomena exhibit within-author regularity. Gambetta et al. [27] confirm that hapax counts degrade under recursive LLM generation. We extend these to the human/AI distinction across multiple tasks.

LLM safety classification. Young [28] finds top-performing guardrails drop from 91% to 34% on novel attacks. Liu et al. [29] show guardrail models are poorly calibrated. Deng et al. [30] demonstrate multilingual queries bypass safety filters. Our 22-language refusal detector addresses the complementary problem of recognizing refusals regardless of language.

Ensemble methods and calibration. Stacking [31] trains a meta-learner on base classifier outputs. Isotonic regression [32] and Platt scaling [33] ensure calibrated probabilities. Wang et al. [34] propose Mixture-of-Agents for LLM collaboration; our multi-LLM approach uses simpler agreement-based filtering.

3. Feature Robustness Framework

We define *fire rate* as the fraction of test samples where a feature produces a non-default value. Fire rate measures *support overlap*: whether the feature's non-trivial region intersects the test distribution.

Table 1
Feature fire rates and effect sizes from Reasoning Trajectory Detection. Generator-specific features had strong effect sizes ($d > 1.5$) but 0% test fire rate; vocabulary fingerprints have moderate effect sizes but 100% fire rate.

Feature	Train	Test	d	Category
`has_final_answer`	93.6%	0.0%	2.12	Anchored
`reasoning_tag`	99.3%	0.0%	1.89	Anchored
`opens_with_the`	70.0%	∼5%	1.54	Anchored
`has_boxed_answer`	99.0%	29.3%	0.91	Portable
`latex_density`	95%+	16%	0.78	Portable
`hapax_ratio`	100%	100%	−.89	Invariant
`yules_k`	100%	100%	.614	Invariant
`heaps_exponent`	100%	100%	.523	Invariant
`compression_ratio`	100%	100%	.412	Invariant
`sentence_length_cv`	98%	97%	.750	Invariant

A feature with 0% fire rate is *dead*: zero contribution regardless of training-set effect size.

We propose three categories based on our Reasoning Trajectory experiments (Table 1):

Domain-Anchored (zero support overlap). Features encoding generator- or topic-specific patterns. `has_final_answer` detects DeepSeek-R1's "**Final Answer:**" marker (93.6% fire rate in training, 0% in test). These features detect *specific generators writing about specific topics*, not AI-generated text in general.

Domain-Portable (support overlap, degraded separability). Features that fire across domains but lose class separation. `has_boxed_answer` detects LaTeX `\boxed{}`, present in 99% of AI math solutions but only 29% of non-math test data.

Domain-Invariant (stable support, stable separability). Features measuring *generation process* properties. Hapax ratio, Yule's K, and Heaps' exponent differ systematically between human and AI text because LLM next-token sampling produces different tail-frequency distributions than human compositional writing [35, 26]. Their 100% fire rate confirms domain independence.

The a priori criterion. A feature should be classified as likely anchored if its definition references a specific generator, output format, or topic. This requires no test data, only introspection about the feature's construction. We applied this criterion when designing the Voight-Kampff feature set.

The validation illusion. IID validation selects features with high training-distribution support, including anchored features that die under shift. On Reasoning Trajectory Subtask 2, the model with the highest validation F1 (XLM-R, 0.834) produced the lowest test F1 (0.448), a gap of −0.386. The diagnostic: when shift is known in advance, validation gains that do not transfer signal feature death, not feature improvement.

4. Task 1: Reasoning Trajectory Detection

The PAN@CLEF 2026 Reasoning Trajectory Detection task [2, 36] defines two subtasks. **Subtask 1 (Source Detection)**: classify a solution as human-written or AI-generated (macro F1). **Subtask 2 (Safety Classification)**: given a query and AI reasoning trace, produce a trace-level binary label and per-step safety scores.

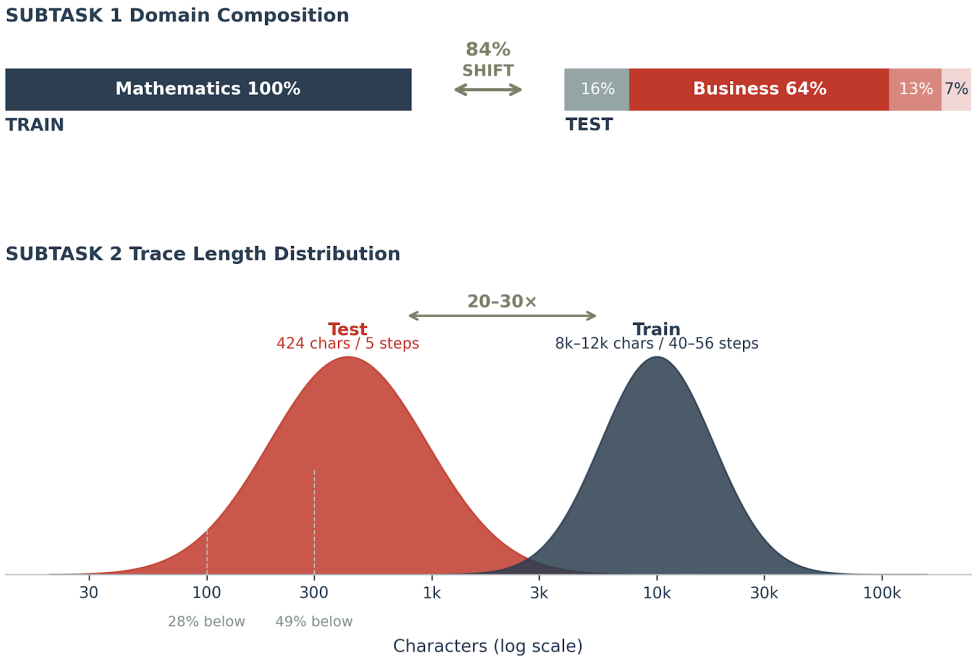

Figure 1: Subtask 1 (top): training was 100% mathematics; 84% of test is unseen domains. Subtask 2 (bottom): test traces average 424 characters across 15+ languages vs. 8,000–12,000 in training.

4.1. Distribution Shift

Training was 100% mathematics; 84% of the test set comprised unseen domains (business 64%, finance 13%, coding 7%). For Subtask 2, training traces averaged 40–56 steps in a single language; test traces averaged 5 steps across 15+ languages and 4 script families (Figure 1).

4.2. Subtask 1: Source Detection (1st Place, 0.85 F1)

Feature-based classifier. After removing 5 dead generator-specific features and adding 7 vocabulary fingerprints, our LightGBM ensemble (5 seeds, 30 features) achieved val F1 0.886, test F1 0.690. Adding vocabulary fingerprints improved validation by +0.172 but test by only +0.009, illustrating that even domain-invariant features transfer imperfectly.

Multi-LLM ensemble. Llama 3.3 70B and Qwen3-235B classify each solution independently via zero-shot prompts. Test F1: 0.720.

Opus–Sonnet agreement (final system). Claude Opus 4.6 and Claude Sonnet 4.6 [37] classify each sample independently. When both agree, their shared label is used; disagreements are broken by LightGBM. Test F1: **0.850**. The largest gain (+0.10) came from capability-tier agreement, not feature engineering.

Table 2
Subtask 1 ablation. The largest gain (+0.10) came from Opus–Sonnet agreement.

System	Test F1	Δ
LightGBM (30 features)	0.690	—
Multi-LLM (Llama + Qwen3)	0.720	+.030
Calibrated combination	0.750	+.030
Opus–Sonnet agreement	**0.850**	+.100

Table 3
The validation illusion on Subtask 2. Supervised methods show inverse val/test correlation.

Approach	Val	Test	Δ
LightGBM (28 feat.)	.502	.362	−.14
Refusal heuristic	.624	.350	−.27
TF-IDF + LR	.793	.500	−.29
XLM-RoBERTa	.834	.448	−.39
5-model LLM ensemble	–	.551	–
Query-refusal decomp.	–	.640	–
Multi-signal union	–	**.660**	–

4.3. Subtask 2: Safety Classification (3rd Place, 0.66 F1)

All supervised approaches collapsed under distribution shift (Table 3): XLM-RoBERTa achieved 0.834 val but 0.448 test. This is the *validation illusion*: IID validation selects for anchored features that die under shift.

Two-stage query-refusal decomposition. Rather than interpreting ambiguous trace content, we decomposed the problem into two distribution-robust sub-questions: (1) *Is the query harmful?* (Claude Sonnet via Batch API, aggressive prompt, 58% classified harmful); (2) *Does the trace contain a refusal?* (100+ compiled regex patterns across 22 languages in 7 categories: direct refusals, apology-led refusals, policy citations, redirections, indirect refusals, safety meta-reasoning, and model self-identification). Combined rule: safe if refusal detected or query benign; unsafe otherwise. Test F1: 0.640.

Multi-signal union (final system). Three Claude Sonnet batches with different prompt strategies combined with query-refusal and LLM ensemble via union (unsafe if any signal votes unsafe). Per-step labels assigned uniformly. Test macro F1: **0.660**. Soft F1: **0.270** (2nd place overall). We emphasize that the uniform per-step assignment is *not* unsafe-step localization: it copies the trace-level label onto every step and scores well only because the test traces are largely label-coherent (Section 8). Its soft-F1 reflects the benchmark's step-label distribution, not any ability of our system to locate the unsafe step.

The granularity trap. Post-hoc step-level experiments (GPT-4o adversarial extraction, XLM-RoBERTa fine-tuned on 3,000 synthetic multilingual traces) uniformly degraded performance. Uniform labels outperformed every fine-grained alternative because a step-level estimator must exceed accuracy $1 - \epsilon_{\text{trace}}$ to beat uniform propagation; neither GPT-4o nor XLM-R cleared this threshold. This threshold is a property of the benchmark—specifically its trace-level label coherence—not of our system; we therefore report the uniform-label outcome as a benchmark observation and make no claim to step-localization capability.

Reproducibility and model versions. All LLM components used fixed model snapshots with greedy decoding (temperature 0). Subtask 1 queried Claude Opus 4.6 (`claude-opus-4-6`) and Claude Sonnet 4.6 (`claude-sonnet-4-6`) with zero-shot prompts capped at 10 output tokens, plus the open models Llama 3.3 70B and Qwen3-235B (`qwen3-235b-a22b`) via hosted inference (OpenRouter/Together.ai). Subtask 2 used Claude Sonnet 4.6 through the Anthropic Message Batches API under three prompt strategies (10 output tokens, except a 200-token rationale variant); GPT-4o and GPT-4o-mini were used only for the post-hoc step-extraction experiments of the granularity-trap analysis. On Opus–Sonnet disagreement the ensemble defers to LightGBM; no separate API-failure fallback was triggered for the batch runs. Experiments ran February–May 2026 on a total compute budget of roughly $43 (Together.ai) and $5 (Modal) plus Claude Batch API usage, which we did not meter per experiment. Exact prompts and configurations are released in the linked repositories.

Error analysis. The failure modes are consistent across subtasks. On Subtask 1, LightGBM errs on the shifted (business/finance/coding) majority of the test set, where its math-calibrated features fall to their default values; the Opus–Sonnet ensemble instead errs on short, low-signal traces where both models default to "AI", which is also where the two agree yet are jointly wrong—so agreement removes the easy errors but not these correlated ones. On Subtask 2, the query–refusal rule misclassifies two categories: traces that comply with a harmful query without emitting an explicit refusal token (predicted safe, actually unsafe), and benign queries phrased as adversarial hypotheticals (predicted unsafe, actually safe). Both are failures of the decomposition's assumption that harm is legible at the query or refusal level rather than in the trace body—the case we deliberately declined to model.

4.4. Leaderboard

Table 4 places our systems on the official PAN 2026 leaderboards [36].[1] On Subtask 1 our Opus–Sonnet agreement system ranks 1st of nine teams (macro F1 0.85, ahead of the next entry at 0.83). On Subtask 2 we rank 3rd of seven on macro F1 (0.66) but 2nd on soft F1 (0.27), behind only the top entry (0.31): the union system sacrifices a little overall accuracy for better-calibrated step scores. The gap between our 1st-place source detection and mid-table safety classification mirrors the difficulty gap between the two subtasks—source detection rewards a high-precision cross-domain signal (LLM agreement), whereas safety classification under shift resists every approach we tried.

Table 4
Leaderboard results (top 5 each subtask).

	Team	F1	Soft F1	Rank (F1)	Rank (Soft)
S1	**davidcondrey**	**.85**	–	**1**	–
	srikarkashyap	.83	–	2	–
	skyduan	.80	–	3	–
	affliction	.75	–	4	–
	23020073	.72	–	5	–
S2	ioanabrr	.75	.31	1	1
	srikarkashyap	.70	.17	2	3
	davidcondrey	**.66**	**.27**	**3**	**2**
	skyduan	.65	.16	4	4
	threshdsnmj	.56	.09	5	5

5. Task 2: Voight-Kampff Generative AI Detection

The PAN@CLEF 2026 Voight-Kampff task [2, 38] requires classifying text as human-written or AI-generated across multiple genres (essays, news, fiction), outputting a continuous probability score in $[0, 1]$. Our feature engineering was directly informed by the Reasoning Trajectory experience: the fire-rate analysis there (Section 3) showed that generator-specific features were *dead* out-of-domain (0% fire rate, despite $d > 1.5$ in training) while vocabulary fingerprints and compression measures retained 100% fire rate and within-author stability across domains. We therefore carried only that evidence forward: every Voight-Kampff feature was selected to be domain-portable, and no feature keyed to a specific generator or genre was included—trading training-set effect size for support overlap with unseen test genres.

[1] Our official runs appear on the PAN leaderboards under two submission identities—`David Condrey` (Reasoning Trajectory Detection) and `writerslogic-inc` (Voight-Kampff); both denote this team.

Table 5
The 44 domain-portable features for Voight-Kampff, organized by category. All features produce meaningful values regardless of genre.

Category	Count	Representative Features
Document structure	7	word/char/sentence counts, paragraph count
Vocabulary richness	8	type-token ratio, hapax ratio, Yule's K, Heaps' exponent, MATTR [39], Zipf coeff.
Sentence structure	6	avg. sent. length, sent. length CV, sent. start diversity, conjunction start ratio
Readability	4	Flesch-Kincaid, Coleman-Liau, avg. syllables
Compression/entropy	4	zlib ratio, char entropy, repetition ratio
Style markers	9	punct ratio, quote density, contraction ratio
Discourse	4	transition diversity, connective formality
Distributional	2	burstiness, intrinsic dimensionality [40]
Perplexity (GPT-2)	4	log-perplexity, burstiness, rank-1 accuracy, binoculars ratio [7]
Total	**44+4**	

5.1. Feature Engineering

We extract 44 domain-portable features organized into eight groups (Table 5), deliberately avoiding genre-specific signals.

The vocabulary fingerprints (hapax ratio, Yule's K, Heaps' exponent, MATTR) are the same features that proved domain-invariant on Reasoning Trajectory (Section 3). They measure properties of the generation process—how vocabulary is sampled and reused—rather than the generated content. The compression features (zlib ratio, character entropy) serve the same role: AI-generated text tends to compress better than human text because it is more predictable.

5.2. System Architecture

Three complementary classifiers are combined via learned stacking:

DeBERTa-v2 transformer. Fine-tuned DeBERTa-v2 [11, 41] exported to ONNX [42] with INT8 dynamic quantization for CPU inference. 3 epochs, LR 2×10^{-5}, AdamW, max 512 tokens.

LightGBM with domain-portable features. Multi-seed LightGBM [3] operating on the 44 stylometric features concatenated with truncated SVD projections of character n-gram (3–6) and word n-gram (1–2) TF-IDF [43], plus GPT-2 perplexity features when available.

Calibrated SVM. Linear SVM with Platt scaling [33] via scikit-learn's `CalibratedClassifierCV` [44], operating on raw sparse character and word n-gram TF-IDF features.

Ensemble stacking. Component probabilities P_{lgb}, P_{svm}, P_{deb} are combined with interaction features ($\max$, $\min$, std, range) via logistic regression stacker, followed by isotonic regression calibration [32]. Borderline predictions receive a small bias toward the human class; predictions within a narrow margin of 0.5 abstain.

5.3. Results

We submitted 11 software configurations to the TIRA evaluation platform [45]; the official results are compiled in the task overview [38]. The configurations share the same trained base models and differ only in how those components are composed at inference, along five axes exposed by the runtime configuration: (i) which base classifiers enter the ensemble (LightGBM only, LightGBM + SVM, or all three with DeBERTa); (ii) the fusion rule (learned logistic stacker over component probabilities

and their interaction features, versus a fixed weighted average); (iii) whether isotonic calibration is applied to the fused score; (iv) the size of the multi-seed LightGBM bag; and (v) the abstention margin around the 0.5 decision boundary. Table 6 summarizes representative configurations. The best system (`large-rank`)—all three classifiers, learned stacker, isotonic calibration—achieved **0.891** ROC-AUC on the PAN 2026 test set with exceptionally balanced sub-metrics: 0.891 (ROC-AUC), 0.853 (C@1), 0.902 (F1), 0.899 (F0.5u), and 0.887 (mean) across evaluation dimensions [38], placing 7th by mean score. On the PAN 2025 backward-compatibility test set, the same configuration scored 0.979.

Table 6
Voight-Kampff results across key configurations. Metric 1 is the primary evaluation score. `large-rank` achieves the best cross-metric balance; `brown-velocity` peaks on Metric 1 but collapses on secondary metrics, exhibiting the threshold-collapse pattern predicted by the feature robustness framework.

Configuration	PAN-26 M1	M2	M3	M4	M5	M6
`large-rank`	**0.891**	**0.891**	**0.853**	**0.902**	**0.899**	**0.887**
`brown-velocity`	0.880	0.519	0.475	0.463	0.678	0.603
`frozen-author`	0.708	0.636	0.493	0.492	0.699	0.606
`lively-fixed`	0.683	0.528	0.501	0.460	0.673	0.569

The contrast between `large-rank` and `brown-velocity` is instructive: `brown-velocity` achieves a comparable primary score (0.880) but its secondary metrics collapse (0.519, 0.475, 0.463), indicating optimization for a single decision boundary at the expense of calibration. `large-rank`'s metric symmetry validates the domain-portable feature design and isotonic calibration pipeline.

6. Task 3: Multi-Author Writing Style Analysis

The PAN@CLEF 2026 Multi-Author Writing Style Analysis (MAWSA) task [2] requires detecting style changes (author switches) at paragraph boundaries, with an "easy" subset and a "hard" adversarial-mimicry subset. Signal selection was guided by the feature robustness framework: under mimicry, surface-level features (vocabulary, sentence length) become unreliable because authors consciously adapt them, making them effectively domain-anchored *relative to the mimicry condition.* We therefore fuse three signals chosen for their resistance to conscious control—the same principle that governs adversarial attacks on other authorship channels, where a verification signal is only as robust as it is hard to manipulate (e.g., timing-forgery against keystroke-based detection [46]). **We did not complete this task's submission on the official platform within the evaluation window, so no official results exist;** the predictions below are a priori registrations of the framework, not measured outcomes.[2]

6.1. Signal 1: Spectral Eigen-Cut

Each paragraph is represented as a TF-IDF vector over character n-grams (bigrams through 4-grams) [9, 43], capturing sub-word patterns that are difficult to consciously control. We build an affinity matrix $A_{ij} = \mathbf{x}_i^\top \mathbf{x}_j$ and apply spectral clustering [47] with $k = 2$. The partition encodes global topological structure: even when adjacent paragraphs are stylistically similar under mimicry, the spectral embedding can separate them if the overall graph supports a two-cluster normalized cut. A boundary is marked where adjacent paragraphs receive different cluster labels.

6.2. Signal 2: Normalized Compression Distance

For each adjacent pair (p_i, p_{i+1}) we compute $\mathrm{NCD}(p_i, p_{i+1}) = \frac{C(p_i p_{i+1}) - \min(C(p_i), C(p_{i+1}))}{\max(C(p_i), C(p_{i+1}))}$, where $C(\cdot)$ is the zlib compressed length. NCD approximates the normalized information distance [48, 49], requires

[2] Our style-change pipeline was fully operational, but a mix-up between the CodaBench and TIRA submission endpoints meant the MAWSA run was never submitted to TIRA before the deadline.

no training data, and is thus domain-invariant by construction; it is most effective when paragraphs exceed roughly 200 characters. We threshold dynamically at $\tau = \mu_{\text{NCD}} + 1.5\,\sigma_{\text{NCD}}$.

6.3. Signal 3: SmolLM-135M Perplexity

Each paragraph is passed through SmolLM-135M [50] to extract token-level surprisal; we derive paragraph-level features (mean, variance, trend) and apply kernel change-point detection with an RBF kernel [51]. SmolLM-135M was chosen for practical reasons: at ∼270 MB it fits within the containerized TIRA evaluation constraints.

6.4. Ensemble Fusion and Predictions

For documents with $n \geq 5$ paragraphs we take the relaxed disjunction of the spectral and NCD signals, with SmolLM perplexity as a secondary cross-check; for short documents ($n < 5$) we fall back to NCD-only detection with a conservative threshold. Because we obtained no official evaluation for this task, we register two a priori predictions from the framework: (1) character n-gram TF-IDF similarity and NCD retain >80% of their easy-subset detection rate on the hard subset, because sub-word typing patterns and compression profiles are hard to mimic; (2) word-level TF-IDF cosine similarity degrades on the hard subset (<30% boundary detection), because mimicking authors adopt similar vocabulary. These are untested predictions, stated so they remain falsifiable against the official results, not claimed findings.

7. Cross-Task Analysis

7.1. Feature Reuse

The pattern of feature reuse is consistent with the framework: vocabulary fingerprints and compression features were shared across tasks, while content-anchored features (generator markers, used only on Trajectory until the fire-rate analysis removed them) were task-specific. Reuse alone is not evidence of transfer, though—only Reasoning Trajectory and Voight-Kampff were officially evaluated, so any MAWSA overlap records design intent, not a measured result.

7.2. Shared Architectural Patterns

Two architectural patterns recur across tasks:

Ensemble diversity. Both systems combine heterogeneous classifiers: DeBERTa + LightGBM + SVM (Voight-Kampff), Opus + Sonnet + LightGBM (Trajectory S1). The diversity principle—different classifiers fail on different inputs—is the practical consequence of mixing features from different robustness categories.

Calibration and fusion. Voight-Kampff uses isotonic regression and learned stacking. Trajectory S2 uses union voting. Both prioritize combining signals over optimizing individual components.

Process over content. Vocabulary fingerprints, compression features, and character n-grams are process-level features that measure *how* text is produced rather than *what* it says. These features are the common thread connecting the evaluated systems.

8. Discussion

Decomposition as invariance factorization. The query-refusal decomposition on Trajectory S2 succeeded by factoring the label into components with different invariance properties: query

harmfulness (invariant to trace length), refusal detection (invariant to query language), and trace interpretation (invariant to neither). The first two are individually shift-tractable; the third was avoided entirely [52]. This principle—decompose along shift dimensions—may apply to other tasks where holistic classification is brittle.

When agreement beats features. On Trajectory S1, two frontier LLMs agreeing on a classification (0.85 F1) outperformed 30 engineered features (0.69 F1) by a wide margin. LLMs *read* text rather than measuring surface statistics; their agreement provides a high-precision cross-domain signal. The calibrated combination (0.75) shows the signals are complementary: LightGBM is confident and correct on extreme cases but unreliable in the middle; LLMs are uniformly decent but occasionally wrong on easy cases.

Implications for shared task design. Future tasks with intentional distribution shift should consider providing a shift-representative development set or noting that IID validation may be actively misleading. Our results suggest that task designers who want to test generalization should expect participants' validation rankings to invert on test.

What uniform labels reveal. Our naive uniform step labels (all 1.0 or all 0.0) placed 2nd on soft F1, implying the test set's step labels are largely trace-coherent. This is a benchmark property, not a system property.

8.1. Limitations

Single participant. Both submissions are from the same researcher using the same analytical lens. The framework was developed retrospectively on Reasoning Trajectory; Voight-Kampff results confirm the framework's predictions (domain-portable features achieved balanced cross-genre performance).

API dependence. Our Trajectory systems depend on proprietary LLM APIs (Claude, GPT-4o), introducing cost, latency, and reproducibility challenges.

Ensemble mechanism unknown. We do not know whether the Opus–Sonnet ensemble succeeds due to capability-tier asymmetry, same-family agreement, or simply Opus being a better classifier.

Framework granularity. The three categories are deliberately coarse. Features exist on a continuum of robustness. Feature-level robustness does not guarantee system-level generalization.

9. Conclusion

We presented the Writerslogic systems for three PAN@CLEF 2026 shared tasks, unified by a feature robustness framework: robustness under distribution shift is governed by support overlap, not training-set effect size. This yields a practical diagnostic (compute fire rates on out-of-domain text), a taxonomy (anchored, portable, invariant), and two consequences (the validation illusion, the granularity trap). We caution that the framework was developed retrospectively on Reasoning Trajectory and only partly corroborated on Voight-Kampff; its status is a hypothesis supported by two evaluated tasks, not an established law.

The framework guided system design across tasks: vocabulary fingerprints and compression features for both Voight-Kampff and Reasoning Trajectory; query-refusal decomposition for Trajectory safety classification. The distinction between *what* was generated and *how* it was generated proved a useful axis for feature selection.

For practitioners: before deploying a detection system under expected shift, audit features by the a priori criterion, distrust validation gains that exceed test gains, and propagate parent-level estimates rather than introducing noisy child estimators. These principles guided our systems to 1st place on source detection (0.85 F1) and 2nd on safety soft F1 (0.27).

Declaration on Generative AI

During the preparation of this work, the author used Claude Code (Anthropic) to assist with code development, experiment orchestration, and literature search. The author used GPT-4o and GPT-4o-mini (OpenAI) to generate synthetic training data and classify test samples for step-level experiments. After using these tools, the author reviewed and edited the content as needed and takes full responsibility for the publication's content.

References

[1] J. Bevendorff, D. Dementieva, M. Fröbe, B. Gipp, A. Greiner-Petter, J. Karlgren, M. Mayerl, P. Nakov, A. Panchenko, M. Potthast, A. Shelmanov, E. Stamatatos, B. Stein, Y. Wang, M. Wiegmann, E. Zangerle, Overview of PAN 2025: Generative AI Authorship Verification, Multi-Author Writing Style Analysis, Multilingual Text Detoxification, and Generative Plagiarism Detection, in: Experimental IR Meets Multilinguality, Multimodality, and Interaction. Proceedings of the Fourteenth International Conference of the CLEF Association (CLEF 2025), Lecture Notes in Computer Science, Springer, 2025.

[2] J. Bevendorff, M. Fröbe, A. Greiner-Petter, A. Jakoby, M. Mayerl, P. Nakov, H. Plutz, M. Potthast, B. Stein, M. N. Ta, Y. Wang, E. Zangerle, Overview of PAN 2026: Voight-Kampff Generative AI Detection, Text Watermarking, Multi-Author Writing Style Analysis, Generative Plagiarism Detection, and Reasoning Trajectory Detection, in: M. Hagen, M. Potthast, B. Stein, P. Schaer, E. Zangerle, A. Barrón-Cedeño, A. G. S. de Herrera, E. S. Salido, S. MacAvaney, J. M. Struß (Eds.), Experimental IR Meets Multilinguality, Multimodality, and Interaction. Proceedings of the Seventeenth International Conference of the CLEF Association (CLEF 2026), Lecture Notes in Computer Science, Springer, Berlin Heidelberg New York, 2026.

[3] G. Ke, Q. Meng, T. Finley, T. Wang, W. Chen, W. Ma, Q. Ye, T.-Y. Liu, LightGBM: A highly efficient gradient boosting decision tree, in: NeurIPS, 2017, pp. 3146–3154.

[4] A. Conneau, K. Khandelwal, et al., Unsupervised cross-lingual representation learning at scale, in: Proceedings of ACL, 2020, pp. 8440–8451. doi:10.18653/v1/2020.acl-main.747.

[5] E. Mitchell, Y. Lee, A. Khazatsky, C. D. Manning, C. Finn, DetectGPT: Zero-shot machine-generated text detection using probability curvature, in: Proceedings of ICML, 2023, pp. 24950–24962.

[6] S. Gehrmann, H. Strobelt, A. M. Rush, GLTR: Statistical detection and visualization of generated text, in: Proceedings of ACL: System Demonstrations, 2019, pp. 111–116. doi:10.18653/v1/P19-3019.

[7] A. Hans, A. Schwarzschild, V. Cherepanova, H. Kazemi, A. Saha, M. Goldblum, J. Geiping, T. Goldstein, Spotting LLMs with binoculars: Zero-shot detection of machine-generated text, in: Proceedings of ICML, 2024, pp. 17519–17537.

[8] S. K. Aityan, W. Claster, K. S. Emani, S. Rais, T. Tran, A lightweight approach to detection of AI-generated texts using stylometric features, arXiv preprint arXiv:2511.21744 (2025). doi:10.48550/arXiv.2511.21744.

[9] E. Stamatatos, A survey of modern authorship attribution methods, JASIST 60 (2009) 538–556. doi:10.1002/asi.21001.

[10] Y. Liu, M. Ott, N. Goyal, et al., RoBERTa: A robustly optimized BERT pretraining approach, arXiv preprint arXiv:1907.11692 (2019).

[11] P. He, X. Liu, J. Gao, W. Chen, DeBERTa: Decoding-enhanced BERT with disentangled attention, in: Proceedings of ICLR, 2021.

[12] S. Park, S. Han, M. Cha, Enhancing domain generalization for robust machine-generated text detection, IEEE TKDE 37 (2025) 7140–7153. doi:10.1109/TKDE.2025.3581694.

[13] Y. Wang, J. Mansurov, P. Ivanov, J. Su, A. Shelmanov, A. Tsvigun, C. Whitehouse, O. M. Afzal, T. Mahmoud, T. Sasaki, T. Arnold, A. F. Aji, N. Habash, I. Gurevych, P. Nakov, M4: Multi-generator,

multi-domain, and multi-lingual black-box machine-generated text detection, in: Proceedings of EACL, 2024, pp. 1369–1407. doi:10.18653/v1/2024.eacl-long.83.

[14] Y. Li, Q. Li, L. Cui, W. Bi, Z. Wang, L. Wang, L. Yang, S. Shi, Y. Zhang, MAGE: Machine-generated text detection in the wild, in: Proceedings of ACL, 2024, pp. 36–53. doi:10.18653/v1/2024.acl-long.3.

[15] H. Xu, J. Ren, P. He, S. Zeng, Y. Cui, A. Liu, H. Liu, J. Tang, On the generalization of training-based ChatGPT detection methods, in: Findings of the Association for Computational Linguistics: EMNLP 2024, 2024, pp. 7223–7243. doi:10.18653/v1/2024.findings-emnlp.424.

[16] E. Tavan, M. Najafi, MarSan at PAN: BinocularsLLM, fusing binoculars' insight with the proficiency of large language models for machine-generated text detection, in: Working Notes of CLEF 2024, 2024.

[17] S. Liu, X. Liu, C. Li, Z. Zhang, G. Ma, Y. Lan, S. Xiao, MGT-Prism: Enhancing domain generalization for machine-generated text detection via spectral alignment, in: Proceedings of AAAI, 2026, pp. 32132–32140. doi:10.1609/aaai.v40i38.40485.

[18] J. Wang, T. Xiao, H. Du, C. Zhang, P. Liu, Cross-domain detection of AI-generated text: Integrating linguistic richness and lexical pair dispersion via deep learning, Pattern Recognition Letters 200 (2026) 123–128. doi:10.1016/j.patrec.2025.12.010.

[19] M. K. Mobin, M. S. Islam, LuxVeri at GenAI detection task 3: Cross-domain detection of AI-generated text using inverse perplexity-weighted ensemble of fine-tuned transformer models, in: Proceedings of the COLING 2025 Workshop on Detecting AI-Generated Content, 2025, pp. 352–357.

[20] A. Ramponi, B. Plank, Neural unsupervised domain adaptation in NLP: A survey, in: Proceedings of COLING, 2020, pp. 6838–6855. doi:10.18653/v1/2020.coling-main.603.

[21] G. U. Yule, The Statistical Study of Literary Vocabulary, Cambridge University Press, 1944.

[22] G. Herdan, Type-Token Mathematics: A Textbook of Mathematical Linguistics, Mouton, 1960.

[23] H. S. Heaps, Information Retrieval: Computational and Theoretical Aspects, Academic Press, 1978.

[24] R. H. Baayen, Word Frequency Distributions, Kluwer Academic Publishers, 2001.

[25] F. J. Tweedie, R. H. Baayen, How variable may a constant be? measures of lexical richness in perspective, Computers and the Humanities 32 (1998) 323–352. doi:10.1023/a:1001749303137.

[26] D. Faltýnek, V. Matlach, Hapax remains: Regularity of low-frequency words in authorial texts, Digital Scholarship in the Humanities 37 (2022) 693–715. doi:10.1093/llc/fqab077.

[27] D. Gambetta, G. Gezici, F. Giannotti, D. Pedreschi, A. Knott, L. Pappalardo, A linguistic analysis of undesirable outcomes in the era of generative AI, arXiv preprint arXiv:2410.12341 (2024).

[28] R. J. Young, Evaluating the robustness of large language model safety guardrails against adversarial attacks, arXiv preprint arXiv:2511.22047 (2025).

[29] H. Liu, H. Huang, X. Gu, H. Wang, Y. Wang, On calibration of LLM-based guard models for reliable content moderation, in: Proceedings of ICLR, 2025.

[30] Y. Deng, W. Zhang, S. J. Pan, L. Bing, Multilingual jailbreak challenges in large language models, in: Proceedings of ICLR, 2024.

[31] D. H. Wolpert, Stacked generalization, Neural Networks 5 (1992) 241–259. doi:10.1016/S0893-6080(05)80023-1.

[32] B. Zadrozny, C. Elkan, Transforming classifier scores into accurate multiclass probability estimates, in: Proceedings of ACM SIGKDD, 2002, pp. 694–699. doi:10.1145/775047.775151.

[33] J. C. Platt, Probabilistic outputs for support vector machines and comparisons to regularized likelihood methods, in: Advances in Large Margin Classifiers, MIT Press, 1999, pp. 61–74.

[34] J. Wang, J. Wang, B. Athiwaratkun, C. Zhang, J. Zou, Mixture-of-agents enhances large language model capabilities, in: Proceedings of ICLR, 2025.

[35] G. K. Zipf, Human Behavior and the Principle of Least Effort, Addison-Wesley, 1949.

[36] M. N. Ta, K. Wan, Y. Lin, Y. Wang, P. Nakov, Overview of the first Reasoning Trajectory Detection Task at PAN 2026, in: A. Barrón-Cedeño, A. G. S. de Herrera, E. S. Salido, S. MacAvaney, J. M. Struß (Eds.), Working Notes of CLEF 2026 – Conference and Labs of the Evaluation Forum, CEUR Workshop Proceedings, CEUR-WS.org, 2026.

[37] Anthropic, Claude model family, 2025. URL: https://www.anthropic.com/claude.

[38] J. Bevendorff, R. R. Gunti, J. Karlgren, M. Fröbe, M. Potthast, B. Stein, Overview of the third "Voight-Kampff" Generative AI / LLM Detection Task at PAN and ELOQUENT 2026, in: A. Barrón-Cedeño, A. G. S. de Herrera, E. S. Salido, S. MacAvaney, J. M. Struß (Eds.), Working Notes of CLEF 2026 – Conference and Labs of the Evaluation Forum, CEUR Workshop Proceedings, CEUR-WS.org, 2026.

[39] M. A. Covington, J. D. McFall, Cutting the Gordian knot: The moving-average type-token ratio (MATTR), Journal of Quantitative Linguistics 17 (2010) 94–100. doi:10.1080/09296171003643098.

[40] E. Levina, P. J. Bickel, Maximum likelihood estimation of intrinsic dimension, in: NeurIPS, 2004, pp. 777–784.

[41] P. He, J. Gao, W. Chen, DeBERTaV3: Improving DeBERTa using ELECTRA-style pre-training with gradient-disentangled embedding sharing, in: Proceedings of ICLR, 2023.

[42] Microsoft, ONNX Runtime, https://onnxruntime.ai/, 2021.

[43] G. Salton, C. Buckley, Term-weighting approaches in automatic text retrieval, Information Processing & Management 24 (1988) 513–523. doi:10.1016/0306-4573(88)90021-0.

[44] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, Édouard Duchesnay, Scikit-learn: Machine learning in Python, Journal of Machine Learning Research 12 (2011) 2825–2830.

[45] M. Fröbe, M. Wiegmann, N. Kolyada, B. Grahm, T. Elstner, F. Loebe, M. Hagen, B. Stein, M. Potthast, Continuous Integration for Reproducible Shared Tasks with TIRA.io, in: Advances in Information Retrieval. 45th European Conference on IR Research (ECIR 2023), Lecture Notes in Computer Science, Springer, 2023, pp. 236–241. doi:10.1007/978-3-031-28241-6_20.

[46] D. Condrey, On the insecurity of keystroke-based AI authorship detection: Timing-forgery attacks against motor-signal verification, arXiv preprint arXiv:2601.17280 (2026).

[47] U. von Luxburg, A tutorial on spectral clustering, Statistics and Computing 17 (2007) 395–416. doi:10.1007/s11222-007-9033-z.

[48] M. Li, X. Chen, X. Li, B. Ma, P. M. B. Vitányi, The similarity metric, IEEE Transactions on Information Theory 50 (2004) 3250–3264. doi:10.1109/TIT.2004.838101.

[49] R. Cilibrasi, P. M. B. Vitányi, Clustering by compression, IEEE Transactions on Information Theory 51 (2005) 1523–1545. doi:10.1109/TIT.2005.844059.

[50] L. B. Allal, A. Lozhkov, E. Bakouch, G. M. Blázquez, L. Tunstall, A. Piqueres, A. Marafioti, C. Zakka, L. von Werra, T. Wolf, SmolLM: Training small language models with high-quality data, arXiv preprint arXiv:2502.02737 (2025).

[51] C. Truong, L. Oudre, N. Vayer, Selective review of offline change point detection methods, Signal Processing 167 (2020) 107299. doi:10.1016/j.sigpro.2019.107299.

[52] M. Arjovsky, L. Bottou, I. Gulrajani, D. Lopez-Paz, Invariant risk minimization, arXiv preprint arXiv:1907.02893 (2019).