

AI-Monitors at ChildSafeAds 2026: A Routed Hybrid for Commercial Content in Child-Facing YouTube Videos

Pragyananda Sahoo¹, Vishnu Tripathi¹, Prakhar Joshi¹, Gaurav Kumar¹,
Neel Mani¹

School of Science & Technology, SRHU, Dehradun¹

{pragyanand.saho|vishnu.tripathi|prakhar.joshi|gaurav.soni}@aicentre.org
neelmani@srhu.edu.in

Abstract

We describe AI-Monitors, our system for the ChildSafeAds shared task at NLLP@EMNLP 2026. The task labels sponsored segments in YouTube videos aimed at teenagers with their commercial type, product category and compliance-risk flags, and provides the input at four levels of increasing collection cost, from the free transcript to a crawled product page. Our system gives each sub-task its own input level and model: a fine-tuned ModernBERT-large encoder reading all four levels predicts commercial type and product category, the latter averaged with a TF-IDF classifier; a second encoder reading only the transcript predicts the compliance flags, averaged on the two disclosure flags with a QLoRA-tuned Qwen2.5-7B-Instruct that also reads the video metadata, before rules from the task’s taxonomy are applied. The system scored 0.6223 mean macro-F1 on the test set, 11th of 21 teams, and labels the 503 test instances in 25.5 GPU-minutes. We report the experiments behind each design choice: in our runs the crawled page helped product categorisation but not compliance flagging, the taxonomy’s definitions helped a prompted model while the legal provisions added little, and in two cases our validation procedure was misleading.

1 Introduction

ChildSafeAds (Bertaglia et al., 2026) treats the monitoring of influencer marketing as a system-design problem: how much can a regulator detect at each level of data access, and at what cost? The input fields come in four levels ordered by collection cost, from the segment transcript to the crawled product page, and three sub-tasks are scored on the same sponsored segments: commercial type (ST1), product category (ST2) and compliance-risk flags (ST3).

Our system, AI-Monitors, gives each sub-task its own input level and model type (Figure 1, Section 5): a fine-tuned ModernBERT-large encoder

reading all four levels for ST1 and ST2 (the latter averaged with a TF-IDF classifier), a second encoder reading only the transcript for ST3, averaged on the two disclosure flags with a QLoRA-tuned Qwen2.5-7B-Instruct, and thresholds and four rules from the task’s taxonomy applied last. It scored 0.6223 mean macro-F1 on the test set; Section 6 reports the leaderboard result and the cost of running it. Section 7 analyses the design choices behind it through four questions:

1. **Q1: Which data access levels help which sub-task?** (Section 7.1) In our encoder runs the crawled page raised ST2 by 0.128 and lowered ST3 by 0.031.
2. **Q2: Which type of model works best for each sub-task?** (Section 7.2) The encoder for ST1; a TF-IDF blend for ST2 in cross-validation, though it lowered the test score; a fine-tuned 7B LLM for the two disclosure flags of ST3.
3. **Q3: Does the legal material provided with the task help?** (Section 7.3) The taxonomy’s definitions raised a zero-shot 7B model’s ST3 by 0.038; the legal provisions did not help overall.
4. **Q4: Which validation results predicted test performance?** (Section 7.4) Consistency across resampled splits was a better guide than effect size in our experiments, but it missed a rare class and seed variance.

We release the code, the analyses behind the numbers reported here, and the exact predictions we submitted.¹

2 Related Work

Disclosure of sponsored content. Prior work has measured disclosure *practice* at scale and its rela-

¹<https://github.com/aimonitors25/childsafeads-nllp2026>

tion to national law (Bertaglia et al., 2025), alongside disclosure studies on YouTube and Pinterest (Mathur et al., 2018) and supervised detectors for Instagram (Zarei et al., 2020; Kim et al., 2021). Recent encoder and embedding-LLM work studies fine-tuning choices (Singh et al., 2025) and complementary predictions (Tripathi et al., 2026). ChildSafeAds (Bertaglia et al., 2026) instead fixes the annotation and varies the input.

Legal-NLP compliance classification. Gui et al. (2025) study legal explanations of compliance decisions; ChildSafeAds scores classification but not explanation. LexGLUE (Chalkidis et al., 2022) and LegalBench (Guha et al., 2023) classify legal text against legal categories, whereas ChildSafeAds classifies platform content against consumer-law categories. Our analysis connects to annotation-artifact literature (Gururangan et al., 2018): reasoning from legal definitions disagrees sharply with the labels (Section 7.3).

3 Task Description and Data

Instances. Each instance is a crowd-submitted SponsorBlock interval on a channel screened as likely to reach a teenage audience (Bertaglia et al., 2026). The segment is known to be commercial and the channel to be child-facing, so the benchmark evaluates classification, not discovery. The four cumulative levels run from the transcript (L1, free) through video metadata including the paid-promotion field (L2, one API call per video) and channel identity (L3, one lookup: channel ID and name) to the crawled page (L4, a crawl of the outbound link). We did not evaluate L3 as a separate level with our fine-tuned models: ENC-L4 reads the channel name as part of its cumulative input, but no configuration stops at L3, so its contribution is not isolated. This was a design choice based on the hypothesis that channel identity adds little across channel-disjoint splits, although a channel name can still carry transferable meaning (16% of dev channel names share a word with a training channel name); in a linear probe (TF-IDF with logistic regression, train to dev), adding it to L2 changed ST1, ST2 and ST3 by -0.003 , -0.001 and -0.002 .

Splits. 2,353 train, 504 dev and 503 test instances over 632, 154 and 153 channels, channel-disjoint by construction. Brands are deliberately *not* held out (Bertaglia et al., 2026): 67.7% of dev and

65.0% of test instances promote a domain seen in training. The splits therefore test generalisation to new channels, not to new advertisers (Section 7.1).

Labels and metric. ST1 is single-label over five classes, ST2 multi-label over twelve, and ST3 multi-label over eight flags, listed with their legal families and counts in Table 5 (Appendix C); inadequate_disclosure is a disclosure too weak or too late, direct_exhortation is directly urging children to buy, and hfss_food_marketing is food high in fat, salt or sugar. The imbalance is extreme (ST1 other has 2 training instances, ST2 gambling 12, ST3 insufficient_context 15), and each task is scored by macro-F1 with equal label weight, so each ST3 flag is worth 4.2% of the leaderboard mean and rare classes weigh as much as frequent ones: dev holds only 5 hfss_food_marketing, 7 insufficient_context and 14 ST1 none instances (Section 7.4). The organisers’ majority-class baseline scores 0.093 on dev.

Annotation. Labels come from an LLM judge under expert-guided iteration of the taxonomy and prompts, with human validation but no systematic double-annotation (Bertaglia et al., 2026). The taxonomy defines undisclosed_advertising as no disclosure in the transcript, description or platform field, so a positive paid-promotion label rules the flag out by definition; none of the 1,281 training instances with official_disclosure = true carries it. This becomes rule (i) of Section 5.

4 Validation Protocol

Local reimplementations of the metric. We reimplemented the metric so that decisions cost no submissions; a dev-phase upload scored 0.6185 both officially and locally, which also settles how ST1 is macro-averaged (Appendix D).

Run-to-run variance. Re-running the same code with the same seed gave 0.618 and 0.575. Single runs are known to be unreliable (Reimers and Gurevych, 2017; Dodge et al., 2020); here two runs of an identical configuration differed by 0.089 on ST1 and 0.056 on ST3, as much as the single-run differences between access levels (0.091 and 0.009); only on ST2 (0.016) was one run informative. We therefore re-measured every single-run conclusion with several seeds (Table 3 confirms both directions, with a larger ST3 effect of 0.046 at the same 1,024-token budget), and the system averages seeds rather than selecting among them.

Component	Checkpoint / model	Input; trained on
ENC-L4	answerdotai/ ModernBERT-large; 4 epochs, 6 seeds	L1–L4 cumulative, 2,048 tokens; train
ENC-L1	same checkpoint; 6 epochs, 6 seeds	L1 transcript, 1,024 tokens; train
TF-IDF-ST2	TF-IDF, per-label logistic regression	page, URL (L4), video title, transcript; train+dev
QLoRA-ST3	Qwen/ Qwen2.5-7B-Instruct, 4-bit QLoRA; 3 epochs, 3 seeds	L1–L2 + taxonomy rubric, 1,024 tokens; train
Rules	3 thresholds, 4 taxonomy rules	transcript length, paid-promotion field; –

Table 1: Components of the submitted system; roles are shown in Figure 1. Train is 2,353 instances, train+dev 2,857. Dev and the cross-validation of Section 4 were used only for selection; no component saw test labels.

Cross-validation and resampled halves. Because dev holds only 504 instances, we built a channel-grouped 5-fold cross-validation over train+dev (**2,857 out-of-fold instances**); whole components were validated on it, and individual constants (thresholds and rules) on 10 to 60 random channel-grouped halves of dev. The adoption criterion, set after the first test submissions (Section 6) and fixed before each later check, was **consistency rather than effect size**: at least 90% of halves or 4 of 5 folds had to improve. The largest change it selected was 6 epochs instead of 4 for the ST3 encoder (+0.032, sd 0.007, **5/5 folds**), an effect smaller than the ± 0.04 run-to-run noise on dev.

5 System

The system is routed twice, by data level and by model family (Figure 1); each routing decision follows from a measurement (Section 7). Four components make up the final submission, named after model type and access level: **ENC-L4** and **ENC-L1** are the encoder at levels 4 and 1, **TF-IDF-ST2** the classical model on ST2, and **QLoRA-ST3** the fine-tuned 7B on ST3; Table 1 gives each component’s checkpoint, input and training data.

Backbone. ModernBERT-large (Warner et al., 2025) with three heads (softmax for ST1, sigmoid for ST2/ST3), trained jointly with pos-weighted BCE ($\text{clip}(n_{\text{neg}}/n_{\text{pos}}, 1, 20)$) so that the rarest flags are not dominated by the most frequent (Appendix C). The two encoder components differ only in input and training length (Table 1; the 6 epochs

follow Section 4), each a six-seed probability average trained on the 2,353 training instances only (hyperparameters in Appendix A). We chose this encoder for its context length: at L4, 98% of inputs exceed 512 tokens (median about 1,300).

TF-IDF-ST2. TF-IDF over three views: word n -grams (1–2) on the page title and text, word n -grams (1–2) on the video title and transcript, and character n -grams (3–5, `char_wb`) on the outbound domain and URL, each fed into per-label logistic regression (Pedregosa et al., 2011), blended 50/50 with ENC-L4 on ST2; it is refit on train+dev for its test predictions.

QLoRA-ST3 (disclosure flags only). Qwen2.5-7B-Instruct (Qwen Team, 2024) under 4-bit QLoRA (Detrmers et al., 2023; Hu et al., 2022), three seeds, fine-tuned on the 2,353 training instances as an **eight-way sequence classifier rather than a generator**, so that its probabilities can be averaged with the encoder’s. Its input carries a compressed rubric of the taxonomy’s written tests (Appendix B) and, unlike ENC-L1, the video title, description and paid-promotion field, so it is the only ST3 component that reads level 2.

Constraint layer. ST3 probabilities are thresholded at 0.5 (inadequate_disclosure at 0.40), then four rules apply in order: (i) the paid-promotion flag rules out undisclosed_advertising (Section 3); (ii) the disclosure flags are mutually exclusive, undisclosed_advertising surviving; (iii) a near-empty transcript (≤ 10 words) forces insufficient_context; (iv) otherwise no_flag at $p \geq 0.6$ stands alone, rules (iii)–(iv) excluding everything else. ST2 falls back to its highest-margin label if nothing clears 0.5. Every constant was fixed by Section 4, and the pipeline rebuilt from this description reproduces our submitted file on all 503 test predictions.

6 Results

Five submissions were permitted; Table 2 gives their test scores and the dev score of each configuration. The final system scored **0.6223** mean macro-F1, ranking **11th of 21 teams** (winning mean 0.7347), at coverage 1.0 (a prediction for every test instance on every sub-task). The family-level ST3 score, the organisers’ secondary score over the three legal families (Table 5), rose from 0.5798 to 0.6501.

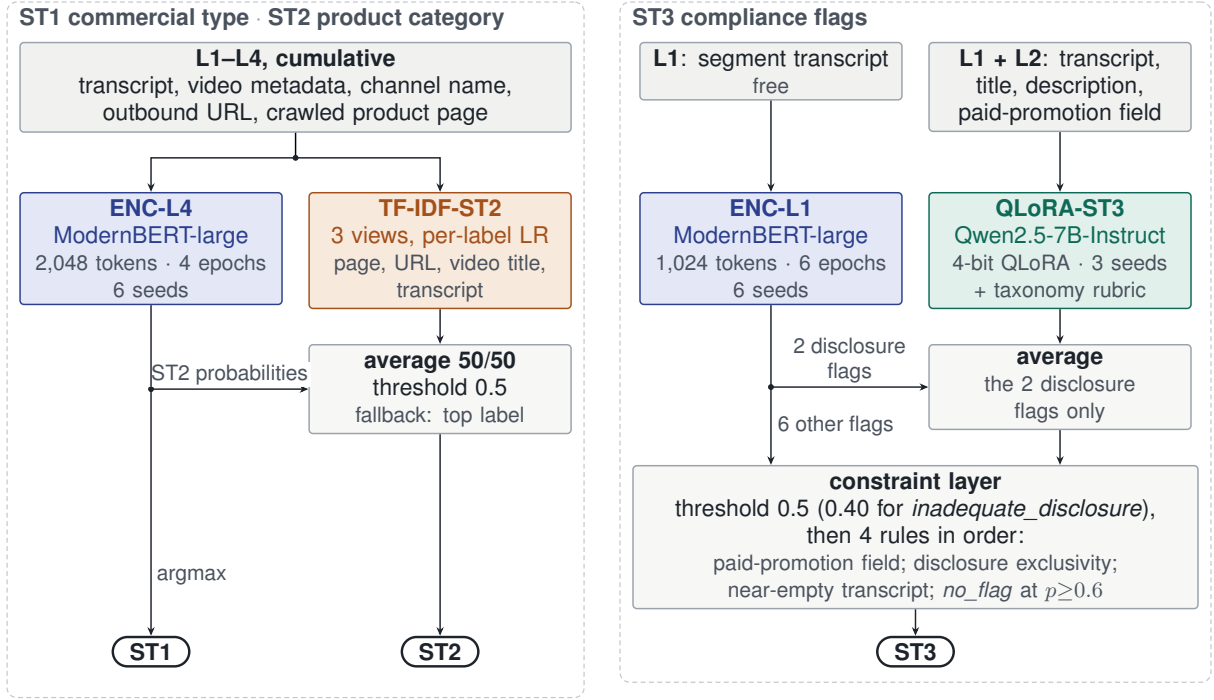


Figure 1: The submitted system (details in Table 1). Left: ENC-L4 produces ST1, and its ST2 probabilities are averaged with TF-IDF-ST2’s. Right: ENC-L1 produces six ST3 flags; its probabilities for the two disclosure flags are averaged with QLoRA-ST3’s before the constraint layer. Colours mark the model type: encoder (blue), classical (orange), LLM (green).

Cost on the 503 test instances. On the training card (16 GB RTX PRO 2000), GPU inference for the whole test set took 25.5 minutes: 5.6 for ENC-L4 and 1.4 for ENC-L1 (six seeds each) and 18.5 for QLoRA-ST3 (three seeds, 9 GB peak memory); TF-IDF-ST2 fits on train+dev in 33 CPU-seconds and, like the constraint layer, predicts the test set in under a second. The inputs are 503 transcripts (L1), 503 metadata calls (L2), 153 channel lookups (L3) and 373 distinct product-page crawls (L4). Training is a separate one-off cost of about 22 GPU-hours (Appendix A); no paid API was used.

Dev and test scores. Without the ST2 blend, the submitted configuration scores 0.6259 on dev and 0.6321 on test, a gap of 0.006 (never uploaded as one file, but each of its sub-task predictions was scored in another submission). The blend has no dev estimate, because TF-IDF-ST2 was refit on train+dev. The agreement holds for the mean only: test is 0.108 below dev on ST1 and 0.078 and 0.049 above it on ST2 and ST3. Across the five configurations, dev was within 0.007 of test for four but overestimated submission 3 by 0.047, and ranked submissions 2 and 3 above submission 1 while test ranked both below it.

#	Configuration	ST1	ST2	ST3	Mean	Dev
<i>evaluation (test) set</i>						
1	routed ensemble	.5906	.7272	.4707	.5962	.5925
2	1 + LLM for exhortation	.5906	.7272	.4657	.5945	.6011
3	1 + ST1 adj., <i>no_flag</i>	.5501	.7272	.4913	.5895	.6360
4	1 + <i>no_flag</i> rule	.5906	.7272	.4913	.6030	.6017
5	4 + ST3, ST2 changes	.5906	.6979	.5784	.6223	—
—	5, no ST2 blend	.5906	.7272	.5784	.6321	.6259
—	dev: ours, no ST2 blend	.6990	.6490	.5296	.6259	

Table 2: Test scores of the five submissions and the dev mean of each configuration; rows 2–4 change submission 1, row 5 changes submission 4. *LLM for exhortation*: a prompted model’s decision replaces the encoder’s; *ST1 adj.*: logit adjustment, $\tau=0.3$; *ST3, ST2 changes*: the 6-epoch encoder, QLoRA-ST3 on the disclosure flags and new thresholds (Section 5), plus the TF-IDF blend (no dev score, as TF-IDF-ST2 was refit on train+dev). Provenance notes: Appendix D.

Use of the five submissions. Submissions 1–4 were diagnostic, and their test scores shaped the final system. Submission 2 replaced the encoder’s *direct_exhortation* decision with a prompted LLM’s, a change never checked on held-out halves of dev; it lowered ST3 by 0.005 and was dropped. Submission 3 bundled the ST1 logit adjustment (Menon et al., 2021) ($\tau=0.3$; 7/10 held-out halves

Config	n	ST1	ST2	ST3
L1 transcript	9	.604±.013	.621±.014	.427±.021
L2 + metadata	1	.594	.637	.416
L4 page @1024	3	.638±.027	.719±.004	.381±.006
L4 page @2048	8	.686±.059	.749±.014	.396±.027
L1 → L4		+.083	+.128	−.031
Welch p		.005	10^{-11}	.018

Table 3: Dev macro-F1, mean $\pm$ sd over independent runs; Welch’s t -test (Welch, 1947) for L1 ($n=9$) against L4@2048 ($n=8$). Cohen’s d for ST2 is 9.0, and the ST3 family level agrees (-0.025 , $p=0.063$).

improved, sd 0.070) with the no_flag rule ($p \geq 0.6$; 10/10, sd 0.010); the per-sub-task scores showed that the first lowered ST1 by 0.041 and the second raised ST3 by 0.021, so submission 4 kept only the rule. These results led us to require consistency for adoption (Section 4), which the logit adjustment would have failed. The submission-5 components were built after submission 4 and selected under that criterion; none was scored on test before the final upload, and the 0.6321 without the blend was identified only afterwards.

7 Analysis

7.1 Q1: Which data access levels help which sub-task?

We fine-tuned the same architecture at L1, L2 and L4 over 21 runs (Table 3); the L4 runs include the L3 channel name, since levels are cumulative, and L3 alone was not run (Section 3). Doubling the page window from 1,024 to 2,048 tokens raises ST2 but leaves ST3 within run-to-run noise, so for ST3 the page appears to act as a distraction.

Implications for monitoring. ST3, the sub-task most directly concerned with compliance risk, does not need the crawled page: transcript-only input scored highest on it (Table 3), and the submitted ST3 components read only L1 and L2. Compliance triage can therefore run without the crawl, deferring it to segments already flagged. In a simulation on dev (Appendix E), crawling 20% of segments, compliance-flagged first, retained 89% of the gain from crawling all of them (0.590 against 0.596, from 0.540 with no crawl), and catching 80% of undisclosed_advertising cost 301 segments per 1,000 screened at precision 0.39. The splits leak brands (Section 3): on the 163 dev instances whose outbound domain does not occur in training, ENC-L4 loses 0.123 on ST1 and 0.076 on

ST2, while the submitted ST3 components score 0.164 higher; every encoder-based ST3 variant we scored gains on these instances ($+0.085$ to $+0.164$), but QLoRA-ST3 alone does not (-0.015 to -0.036). The two parts differ in label mix, so this is an observation rather than a controlled comparison (Appendix E), but it suggests that product identification partly memorises advertisers while the encoder-based compliance flags do not. These figures use the system before its final ST3 changes, and test labels are withheld, so we could not check whether they transfer.

7.2 Q2: Which type of model works best for each sub-task?

The best type of model also differs by sub-task. We trained a simple classical model (TF-IDF over the product page, the video title and transcript, and the outbound URL) on the cross-validation of Section 4, with the same folds and decision rule (means over five folds). On ST1 the encoder is best (0.614), and both the classical model (0.595) and the 50/50 blend (0.603) are lower, the blend winning only 2 folds of 5; on ST2 the classical model beats the encoder (0.686 against 0.661) and the blend reaches 0.731, **winning 5 folds of 5**, between half and four fifths of it recalibration of the encoder’s own thresholds (Appendix D). ST2 asks *what category of product is promoted*, which character n -grams over the URL often identify from the brand; ST1 asks *what kind of commercial relationship it is*, which the brand name does not determine. This reading of ST2 as largely lexical rests on our two model types, an encoder that truncates the page at 2,048 tokens and a TF-IDF model that matches only words: another team’s open-weights LLM reached 0.8243 on ST2 against our best of 0.7272, so ST2 leaves room for models that read the page more fully.

QLoRA-ST3 on the disclosure flags. QLoRA-ST3 is worse than the encoder overall but better on the two disclosure flags (Table 7, Appendix D): it loses on six flags of eight and by 0.051 on the ST3 macro, but wins by 0.240 on undisclosed_advertising and by 0.113 on inadequate_disclosure. The comparison is confounded: QLoRA-ST3 also reads the level-2 fields and a rubric of the taxonomy’s definitions, and Table 7 averages three of its seeds against six encoder seeds. With three encoder seeds the gaps are 0.245 and 0.081, so seed count does not explain them; nor, apparently, does the input alone, since one encoder run

on level 2 scored 0.52 on `undisclosed_advertising`, within the range of single transcript-only runs (0.49–0.61), against 0.79–0.87 for single QLoRA-ST3 runs. We cannot separate the model from its rubric. Averaging QLoRA-ST3 into all eight flags scores higher on full dev than the disclosure-only blend (0.541 against 0.530) but improves only 16 of 20 channel-grouped halves against 20 of 20, so only the latter met the adoption criterion and was shipped (Appendix D).

7.3 Q3: Does the legal material provided with the task help?

The organisers pose this as an unscored question. We ran Qwen2.5-7B-Instruct zero-shot on ST3 in three conditions differing *only* in the legal grounding in the prompt (per-flag scores in Table 9, Appendix D). In single greedy runs on dev, the taxonomy’s plain-language definitions raise the ST3 macro from 0.254 with bare flag names to 0.292, and the legal provisions on top give 0.290; the encoder scores 0.427. The exception is `direct_exhortation`, the one flag defined as an explicit three-part test rather than a category: with provisions it scores 0.45, our best score on that flag in this paper, against 0.41 with definitions alone and 0.34 for the encoder, in a single run on 77 positive instances. Conversely, on `misleading_claim` definitions barely help the prompted model (0.09 against 0.76 for the encoder) while eight demonstrations reach 0.70, suggesting that the encoder learns how the annotation applies the flag rather than performing legal analysis (Appendix B).

7.4 Q4: Which validation results predicted test performance?

Two sets of changes were made in the final submission, both validated by the same criterion. The ST3 changes (Table 2), adopted on 20 resampled halves of dev (+0.072, 20/20), delivered **+0.087**; the ST2 blend, adopted on five folds of the 2,857-instance cross-validation (+0.070, 5/5), delivered **−0.029**. The larger and more expensive validation procedure was the one whose estimate did not hold, for a reason it could not detect (Appendix D): on test the blend cut the predictions of every ST2 class, most for the rare ones, and took gambling from 3 predictions to **0**, which alone costs up to 0.083 of a twelve-class macro-F1; with 17 instances in train+dev, about three per fold, gambling was too rare for five folds to disagree. A second experiment, an auxiliary span-tagging head trained on

the taxonomy’s evidence quotes, showed the complementary blind spot: it changed ST3 by -0.013 over ten matched pairs (sd 0.028, winning 4 of 10), yet seed 0 alone looked like an improvement at 4 of 5 folds, and changing only the seed moved the baseline by 0.036, four times the effect measured. We now require that no multi-label change takes any class below half its predictions or to zero, and that no ST3 component is adopted on single-seed evidence.

8 Conclusion

In our experiments the three sub-tasks of Child-SafeAds were best served by different inputs and models: on dev, routing ST3 to a transcript-only encoder scored 0.006 above using a single L4 encoder for all three sub-tasks, a difference within our run-to-run noise. The value of the routing lies in cost rather than score: ST3, the sub-task most directly concerned with compliance risk, did not need the crawled page and its encoder-based components scored no worse on advertisers not seen in training (Section 7.1), so compliance triage can run on the transcript alone, with the crawl reserved for segments already flagged. The main lesson concerns validation (Q4): two sets of changes adopted under the same consistency criterion had opposite effects on test, and in our setting neither a larger nor a resampled validation set by itself revealed the rare-class and seed-variance problems behind that.

Limitations

The L2 configuration ran once, so we claim no ordering among the cheap levels, only that the crawl lowered ST3, by 0.031. Because a video entered the corpus only if its page could be crawled, our measured value of the crawl is an upper bound. Labels come from an annotation pipeline, so scores measure agreement with it rather than with law. The corrected 0.6321 without the ST2 blend was never uploaded as one file (Section 6). Our reading of ST2 as *lexical* rests on two model types and is qualified by another team’s LLM result of 0.8243 (Section 7.2); all our LLM results rest on one 7B at 4-bit precision.

Acknowledgments

We thank the ChildSafeAds organizers for the shared task, dataset and evaluation framework. We are grateful to Piyush Arora (American Express) for his guidance throughout this work.

References

- Thales Bertaglia, Catalina Goanta, Gerasimos Spanakis, and Gunes Acar. 2026. [ChildSafeAds shared task 2026: Commercial content in child-facing YouTube videos](#). *Preprint*, arXiv:2608.19165.
- Thales Bertaglia, Catalina Goanta, Gerasimos Spanakis, and Adriana Iamnitchi. 2025. [Influencer self-disclosure practices on Instagram: A multi-country longitudinal study](#). *Online Social Networks and Media*, 45:100298.
- Ilias Chalkidis, Abhik Jana, Dirk Hartung, Michael Bommarito, Ion Androutsopoulos, Daniel Martin Katz, and Nikolaos Aletras. 2022. [LexGLUE: A benchmark dataset for legal language understanding in English](#). In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 4310–4330. Association for Computational Linguistics.
- Tim Dettmers, Artidoro Pagnoni, Ari Holtzman, and Luke Zettlemoyer. 2023. [QLoRA: Efficient finetuning of quantized LLMs](#). In *Advances in Neural Information Processing Systems 36*, pages 10088–10115.
- Jesse Dodge, Gabriel Ilharco, Roy Schwartz, Ali Farhadi, Hannaneh Hajishirzi, and Noah Smith. 2020. [Fine-tuning pretrained language models: Weight initializations, data orders, and early stopping](#). *Preprint*, arXiv:2002.06305.
- Neel Guha, Julian Nyarko, Daniel E. Ho, Christopher Ré, et al. 2023. [LegalBench: A collaboratively built benchmark for measuring legal reasoning in large language models](#). In *Advances in Neural Information Processing Systems 36, Datasets and Benchmarks Track*, pages 44123–44279.
- Haoyang Gui, Thales Bertaglia, Taylor Annabell, Catalina Goanta, Tjomme Dooper, and Gerasimos Spanakis. 2025. [Evaluating LLM-generated legal explanations for regulatory compliance in social media influencer marketing](#). In *Proceedings of the Natural Legal Language Processing Workshop 2025*, pages 157–171, Suzhou, China. Association for Computational Linguistics.
- Suchin Gururangan, Swabha Swayamdipta, Omer Levy, Roy Schwartz, Samuel R. Bowman, and Noah A. Smith. 2018. [Annotation artifacts in natural language inference data](#). In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 2 (Short Papers)*, pages 107–112, New Orleans, Louisiana. Association for Computational Linguistics.
- Edward J. Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. 2022. [LoRA: Low-rank adaptation of large language models](#). In *International Conference on Learning Representations*.
- Seungbae Kim, Jyun-Yu Jiang, and Wei Wang. 2021. [Discovering undisclosed paid partnership on social media via aspect-attentive sponsored post learning](#). In *Proceedings of the 14th ACM International Conference on Web Search and Data Mining (WSDM)*, pages 319–327. ACM.
- Zachary C. Lipton, Charles Elkan, and Balakrishnan Naryanaswamy. 2014. [Optimal thresholding of classifiers to maximize F1 measure](#). In *Machine Learning and Knowledge Discovery in Databases, Lecture Notes in Computer Science*, pages 225–239. Springer.
- Jiachang Liu, Dinghan Shen, Yizhe Zhang, Bill Dolan, Lawrence Carin, and Weizhu Chen. 2022. [What makes good in-context examples for GPT-3?](#) In *Proceedings of Deep Learning Inside Out (DeeLIO 2022): The 3rd Workshop on Knowledge Extraction and Integration for Deep Learning Architectures*, pages 100–114. Association for Computational Linguistics.
- Arunesh Mathur, Arvind Narayanan, and Marshini Chetty. 2018. [Endorsements on social media: An empirical study of affiliate marketing disclosures on YouTube and Pinterest](#). *Proceedings of the ACM on Human-Computer Interaction*, 2(CSCW):119:1–119:26.
- Aditya Krishna Menon, Sadeep Jayasumana, Ankit Singh Rawat, Himanshu Jain, Andreas Veit, and Sanjiv Kumar. 2021. [Long-tail learning via logit adjustment](#). In *International Conference on Learning Representations*.
- Fabian Pedregosa, Gaël Varoquaux, Alexandre Gramfort, Vincent Michel, Bertrand Thirion, Olivier Grisel, Mathieu Blondel, Peter Prettenhofer, Ron Weiss, and Edouard Duchesnay. 2011. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830.
- Qwen Team. 2024. [Qwen2.5 technical report](#). *Preprint*, arXiv:2412.15115.
- Nils Reimers and Iryna Gurevych. 2017. [Reporting score distributions makes a difference: Performance study of LSTM-networks for sequence tagging](#). In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 338–348, Copenhagen, Denmark. Association for Computational Linguistics.
- Azad Singh, Vishnu Tripathi, Ravindra Kumar Pandey, Pragyanand Saho, Prakhar Joshi, Neel Mani, Richa Alagh, Pallaw Mishra, and Piyush Arora. 2025. [AI-monitors at GenAI detection task 1: Fast and scalable machine generated text detection](#). In *Proceedings of the 1st Workshop on GenAI Content Detection (GenAIDetect)*, pages 230–235, Abu Dhabi, UAE. International Conference on Computational Linguistics.
- Vishnu Tripathi, Prakhar Joshi, Azad Singh, Pragyanand Saho, Gaurav Kumar, Neel Mani, and Piyush Arora.

2026. [AI-monitors at SemEval-2026 task 4: A hybrid embedding and LLM ensemble for narrative similarity](#). In *Proceedings of the 20th International Workshop on Semantic Evaluation (2026)*, pages 2139–2148, San Diego, California, USA. Association for Computational Linguistics.

Benjamin Warner, Antoine Chaffin, Benjamin Clavié, Orion Weller, Oskar Hallström, Said Taghadouini, Alexis Gallagher, Raja Biswas, Faisal Ladhak, Tom Aarsen, Griffin Thomas Adams, Jeremy Howard, and Iacopo Poli. 2025. [Smarter, better, faster, longer: A modern bidirectional encoder for fast, memory efficient, and long context finetuning and inference](#). In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2526–2547. Association for Computational Linguistics.

B. L. Welch. 1947. [The generalization of ‘Student’s’ problem when several different population variances are involved](#). *Biometrika*, 34(1–2):28–35.

Koosha Zarei, Damilola Ibosiola, Reza Farahbakhsh, Zafar Gilani, Kiran Garimella, Noel Crespi, and Gareth Tyson. 2020. [Characterising and detecting sponsored influencer posts on Instagram](#). In *2020 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining (ASONAM)*, pages 327–331. IEEE.

Appendix

A System architecture and configuration

Encoders. ModernBERT-large in both encoder components: learning rate 2×10^{-5} , AdamW with weight decay 0.01, batch 4 with gradient accumulation 4 (effective 16), bf16 autocast, 6% cosine warmup, gradients clipped at 1.0. Token budgets, epochs and seed counts are per component in Figure 1, and seed probabilities are averaged. **TF-IDF-ST2’s** three views are word 1–2 grams on the page title and text, word 1–2 grams on the video title and transcript, and character 3–5 grams with char_wb on the outbound domain and URL.

QLoRA-ST3. Qwen2.5-7B-Instruct trained as an eight-way sequence classifier. LoRA $r=16$, $\alpha=32$, dropout 0.05, no bias, on q_proj, k_proj, v_proj, o_proj and the MLP projections; NF4, 3 epochs, lr 1×10^{-4} , weight decay 0.01, max length 1024. Trained on the 2,353 training instances only; dev was never seen.

Cost. ≈ 22 GPU-hours on one 16 GB card (RTX PRO 2000), all one-off training: six L4@2048 runs (5.2 h), six L1@1024 6-epoch runs (7.8 h), three QLoRA runs (9.0 h) and 33 CPU-seconds to fit the classical model on train+dev, so the 7B is

41% of training cost. Inference on the 503 test instances, measured, takes 25.5 GPU-minutes, 73% of it the 7B (Section 6); the fine-tuned weights were not saved, so each model was timed from its base checkpoint with the same architecture, inputs, batch size and precision, on which speed does not depend. The wider study cost ≈ 48 GPU-hours over 76 successful fine-tuning runs: 27 ModernBERT configurations on dev (the 21 of Table 3 among them), 40 cross-validation folds (20 of them for the span-supervision test of Section 7.4), 3 QLoRA seeds, and 6 models retrained on train+dev. Those six were built but never submitted, because a model trained on dev cannot be validated on dev. The final submission rests on 15 of the runs (six ENC-L4, six ENC-L1, three QLoRA-ST3), all trained on train only, and each of our five uploaded files is reproduced exactly when rebuilt from train-only models.²

B Prompt templates

Every prompting condition in Section 7.3 shares one system-prompt **head** and one **tail**, and differs *only* in the block inserted between them (Table 4). That is what makes the comparison controlled: model, decoding, instances, parsing and constraint layer are held fixed.

Condition	Block inserted between head and tail
BARE	nothing
+DEFS	FLAG DEFINITIONS: the taxonomy’s plain-language entry for each flag
+PROV	the same definitions, plus LEGAL GROUNDING: instrument, provision, severity and note per flag, from the shipped legal_provisions.json

Table 4: The three legal-grounding conditions; only the inserted block changes.

Decoding and parsing. Qwen2.5-7B-Instruct, 4-bit NF4 with double quantisation and bfloat16 compute, batch 8, greedy (do_sample=False), max_new_tokens=48. We take the first JSON array; if the model ignores the format we fall back to bare flag mentions. **The constraint layer the encoder receives is then applied unchanged:** the model does not honour the disclosure exclusivity rule on its own, so the comparison is of models, not of post-processing.

²Wall-clock was not logged per training run, so training GPU-hours are estimates from observed run times; inference times are measured, and run counts are exact from surviving artefacts.

System prompt. The head states the two facts the release gives and forbids re-assessing them; the tail fixes the output format.

System prompt, head (all conditions)

You audit sponsored segments from YouTube videos that reach a child and teenage audience, under EU consumer law. Two facts are given and must NOT be re-assessed: the channel is child-facing, and the segment IS commercial.

Decide which compliance flags apply.

[INSERT: condition-dependent block, Table 4]

System prompt, tail (all conditions)

Answer with a JSON array of flag strings and nothing else. Use [] if the segment appears compliant. Emit every flag that applies. Valid flags:
undisclosed_advertising
inadequate_disclosure
direct_exhortation
misleading_claim
age_restricted_or_prohibited_product
hfss_food_marketing

User turn. Identical in every condition. Only this turn carries instance data, and it is level 2: transcript plus title, paid-promotion field and description.

User turn (all conditions)

TRANSCRIPT OF SPONSORED SEGMENT:
{transcript, truncated to 3500 chars}

VIDEO TITLE: {title}
PLATFORM PAID-PROMOTION LABEL:
{true | false | unknown}
VIDEO DESCRIPTION:
{description, truncated to 1200 chars}

The full definition of direct_exhortation. We reproduce direct_exhortation because it is the only flag the taxonomy states as an explicit *test* rather than a category, and Section 7.3 turns on exactly that distinction.

Taxonomy definition inserted for direct_exhortation

direct_exhortation
A direct appeal to children to buy the product, or to persuade their parents to buy it for them.

COUNTS
explicit purchase appeal in the imperative ("go and buy this", "ask your parents to order it"); or wording that would be a plain instruction but targets the young audience with personal, hyped or pressuring language -- parasocial appeals ("if you love us, please download it"), pleading, repetition, urgency, child-directed slang.

DOES NOT COUNT
basic transactional instructions, even imperative -- "download the app from the link below", "use my code for 15% off". Stating where to obtain the product, or that a discount exists, is not an exhortation.

BOUNDARY
the test is pressure on the child to make the purchase happen, not the presence of an imperative verb. If genuinely ambiguous, do not flag.

Few-shot variants. Eight demonstrations are prepended as alternating user/assistant turns, each a shortened instance (transcript to 1100 chars, title, paid-promotion label) answered with the gold flag array: random uses one fixed set for every instance, knn the nearest neighbours by TF-IDF over transcripts, following the standard finding that semantically similar demonstrations outperform random ones (Liu et al., 2022). Only the six substantive flags are demonstrated; no_flag and insufficient_context come from rules. Retrieved demonstrations move misleading_claim from 0.09 to 0.70 and ST3 macro from 0.292 to 0.368, while direct_exhortation falls from 0.41 to 0.33.

C Label distributions

Flag	Family	Severity	Train	Dev
misleading_claim	content	cond.	1,277	260
inadequate_disc.	disclosure	cond.	611	118
no_flag	—	—	529	127
undisclosed_adv.	disclosure	<i>per se</i>	352	74
direct_exhortation	content	<i>per se</i>	304	77
age_restricted	product	cond.	59	16
hfss_food	product	soft law	40	5
insufficient_ctx.	—	—	15	7

Table 5: The eight ST3 flags with the organisers’ legal-family and severity assignments (Bertaglia et al., 2026).

Class	Train	Dev
<i>ST1, single-label</i>		
physical_goods	1,125	218
digital_content_or_services	1,069	248
physical_services	122	24
none	35	14
other	2	0
<i>ST2, multi-label</i>		
apps	654	171
hardware_electronics	516	120
other	412	87
food	293	49
creator_community	269	47
fashion	267	50
health	264	55
education	140	12
financial	135	20
gambling_adjacent	92	17
toys	77	8
gambling	12	5

Table 6: ST1 and ST2 label distributions.

Collection artefacts. Table 6 gives the full ST1 and ST2 distributions. A video enters the dataset only if at least one description link yielded a usable

page, a filter that removed 22.4% of candidates, so dead and bot-blocking pages are underrepresented. For 1,208 of 3,360 instances no link could be confidently matched to the marked segment, so the first usable page stands in and may describe a different promotion; 60 instances have no transcript at all.

D Supporting measurements

How ST1 is macro-averaged. The task does not specify whether ST1 macro-averages over all five labels or only those present in the reference, and because other has 2 training and 0 dev instances the two readings differ by ≈ 0.14 on identical predictions. A dev-phase upload scored 0.6185 officially; our reimplementation scores that file 0.6185 under the reference-present reading and 0.5711 under the all-labels reading, which identifies the convention rather than merely agreeing with it. Constants were validated on resampled channel-grouped halves of dev: 10 draws for no_flag and the ST1 logit adjustment, 20 for the ST3 hybrid, 30 for the ST2 thresholds and 60 for the product gates. Whole components were validated on the 5-fold cross-validation.

Flag	n	Enc.	QLoRA
undisclosed_adv.	74	.610	.850
inadequate_disc.	118	.369	.482
direct_exhortation	77	.353	.324
misleading_claim	260	.769	.685
age_restricted	16	.600	.160
hfss_food	5	.333	.222
no_flag	127	.559	.494
insufficient_ctx.	7	.435	.400
ST3 macro		.503	.452

Table 7: Dev, six-seed encoder against three-seed QLoRA at a flat 0.5, no constraint layer. Averaging seeds is why the encoder column exceeds the single runs of Table 3.

Model	ST1	ST2	ST3	mean
seed 0 baseline	.473	.571	.433	.492
seed 0 + spans	.481	.566	.442	.496
seed 1 baseline	.482	.588	.469	.513
seed 1 + spans	.476	.554	.437	.489
<i>ten matched pairs, treatment – baseline</i>				
ST3	–0.013	sd 0.028	4/10	

Table 8: Span supervision, pooled out-of-fold over 2,857 instances per cell (Section 7.4).

Complementarity or recalibration. On the cross-validation, the ST2 blend lowers the label

rate from 1.91 per instance to 1.35, against 1.32 in the gold labels, so its gain could be a threshold adjustment in disguise. We tested this: the encoder with per-label thresholds fitted within each training fold reaches 0.696 with a threshold grid of step 0.05 and 0.716 with step 0.01 (means over the five folds), so recalibration accounts for 50% to 79% of the blend’s $+0.070$. The rest comes from complementary errors that no adjustment of the encoder’s own thresholds reproduces. We could not separate the two halves on test; the test regression (Section 7.4) coincides with the lower label rate, which removed every gambling prediction.

Flag	Enc.	Bare	+Defs	+Prov.
misleading_claim	.76	.02	.09	.21
undisclosed_adv.	.65	.48	.36	.42
age_restricted	.45	.15	.37	.32
inadequate_disc.	.41	.13	.22	.14
direct_exhortation	.34	.41	.41	.45
no_flag	.22	.35	.42	.37
ST3 macro	.427	.254	.292	.290

Table 9: Zero-shot ST3 per flag under the three grounding conditions (Section 7.3), against a single-run encoder.

What we did not test. We did not sweep the span-supervision weight: with a seed swing of 0.036, tuning it with this validation setup would fit noise. Level 1 was a reasonable first test but not the only option: evidence-quote coverage for the two target flags rises from 61%/69% at L1 to **93%/98% at L4**, so a level-4 span model remains plausible and untested.

Other approaches we rejected. DeBERTa-v3-large as a second architecture diverged to NaN from epoch 1 under bf16 at this learning rate, scoring 0.068, below the majority baseline. Three ST3 threshold schemes were fitted on one channel-grouped half of dev and scored on the other, over 20 halves: per-label thresholds gained 0.055 on the fitting half and lost 0.048 on the other (1 of 20 halves improved), joint coordinate ascent gained 0.076 and lost 0.023 (7/20), and a single global threshold changed little either way ($+0.010$, $+0.001$). None was adopted (on thresholding for F1, see Lipton et al., 2014). An outbound-domain gazetteer was worth $+0.13$ over TF-IDF but $+0.001$ over the encoder, which already learns the same brands; a keyword lexicon built from the taxonomy reached F1 0.12 and got worse at level 4. Deriving ST1

from predicted ST2 cost 0.085 and removed every correct prediction of none.

Blending only the disclosure flags. Averaging QLoRA-ST3 into all eight flags raises ENC-L1’s dev ST3 under the shipped constraint layer from 0.502 to 0.541, and into the two disclosure flags only to 0.530; but across 20 channel-grouped halves of dev the all-flags blend gains +0.011 (sd 0.046, 16 halves improved) and the disclosure-only blend +0.031 (sd 0.007, all 20). Only the second meets the adoption criterion of Section 4, so we shipped it; the all-flags blend was never scored on test, so this shows which change the criterion selects, not that the other would have done worse.

Why the disclosure flags. A possible explanation is that `undisclosed_advertising` is a written test over the input that a model reading the test can apply directly, whereas `age_restricted` and `hfss` are sparse product categories (59 and 40 training instances).

The two validation failures in detail. The following expands Section 7.4.

Failure 1: a rare class that cross-validation could not detect. On test, the blend lowers ST2 from 1.89 labels per instance to 1.43 (1.32 in train+dev), cutting predictions most for the rare classes: `gambling` 3 → 0, `gambling_adjacent` 47 → 24, `health` 72 → 46. A class predicted zero times has F1 exactly 0, and one of twelve ST2 classes is worth up to 0.083, so **a single rare class accounts for the entire regression**. `gambling` has 17 instances in train+dev, about three per fold: enough to fire occasionally, never enough for five folds to disagree, so the cross-validation reported 5/5 while the twelfth class was too rare in every fold to register.

Failure 2: an apparent effect that was seed variance. The release ships the verbatim quote justifying each ST3 flag, and ST3’s weakest classes seem to lack training signal rather than be intrinsically hard (`age_restricted` has 75 positives in train+dev and `hfss_food_marketing` 45, 96% and 98% of them with the triggering phrase). We added an auxiliary token-tagging head to the same encoder; at weight 0.0 no parameters are created and the RNG stream is untouched, so the baseline is bit-identical to a run without it. Over ten matched pairs the effect is **ST3 −0.013, sd 0.028, winning 4 of 10** (Table 8): seed 0 alone looks like

an improvement (+0.009 at 4 of 5 folds) and seed 1 reverses it, and changing only the seed moves the baseline on ST3 from 0.433 to 0.469, **a swing of 0.036, four times the effect measured**, which fold consistency within one seed cannot detect.

Resulting rules. No multi-label change may take any class below half its predictions or to zero, and no ST3 component is adopted on single-seed evidence. In both failures the validation setup had been enlarged, resampled and cross-checked and still missed the problem: once because the deciding quantity was below its resolution, once for lack of a seed control.

E Deployment analysis on dev

All figures here are computed on dev with the six-seed, four-epoch L1 encoder for ST3 and the six-seed ENC-L4 for ST1 and ST2, the system before its final ST3 changes. Test labels are withheld, so whether these operating points transfer to test is unchecked.

Cascade. ST3 is always answered by the transcript-only encoder, and so are ST1 and ST2 except for the segments escalated to a crawl, which ENC-L4 answers. Crawling 20% of segments, compliance-flagged ones first, retains 89% of the gain from crawling all of them (0.590 against 0.596, from 0.540 with no crawl), and which segments are crawled matters: at a 10% budget, escalating the segments on which the transcript-only tier is least certain scores 0.577 against 0.546 for random choice. The curve is not monotonic, so we claim only that most of the crawl is optional. Reweighting the six substantive flags by the organisers’ severity classes (*per se* 3 for practices banned outright, conditional 2, soft law 1) grows the advantage of the L1 encoder over the L4 encoder from +0.030 to +0.040, mostly because of `undisclosed_advertising` (0.65 against 0.51); on `direct_exhortation`, the other *per se* flag, the two tie at 0.34.

Review workload. For deployment, the relevant measure is the review workload at a fixed recall target rather than macro-F1. Catching 80% of `undisclosed_advertising` costs 301 segments per 1,000 screened at precision 0.39; for `direct_exhortation` precision stays low at every recall level we measured (0.22 at 80% recall). At a 20% crawl budget the pipeline issues 200 crawls and 738 analyst items per 1,000 (59% flagged, 15% on

which it abstains). Flagging this many segments is expected, since the reference marks misleading_claim on 52% of dev, so the order of the queue is what makes it useful: ranked by a noisy-or (the chance at least one fires) over the two *per se* prohibitions, the top 10% surfaces **28% of all segments labelled with a *per se* flag, at 76% precision**, against a 26.8% base rate. Compute costs are small in comparison: ≈ 22 GPU-hours of training and no paid API, 41% of it QLoRA-ST3 (Appendix A).

Advertisers not seen in training. The splits leak brands (Section 3). Splitting dev by whether an instance’s outbound domain occurs in training (341 seen, 163 unseen), ENC-L4 loses 0.123 on ST1 and 0.076 on ST2 on the unseen part, while the submitted ST3 components score 0.164 higher; every encoder-based ST3 variant gains there (+0.085 to +0.164), but QLoRA-ST3 alone does not (−0.015 to −0.036). The two parts differ in label mix (undisclosed_advertising 24.5% against 10.0%, age_restricted 9.2% against 0.3%), so this is an observation rather than a controlled comparison; it suggests that product identification partly memorises advertisers while the encoder-based compliance flags do not.