QUALITY IS NOT SINGULAR: AUDITING, DISTILLING, & BLENDING LLM-JUDGE ANNOTATIONS FOR PRE-TRAINING DATA CURATION

Anonymous authors
Paper under double-blind review

ABSTRACT

Curation of data in language model training commits the corpus to particular notions of 'quality'. We study that commitment in three parts: judging different notions of quality, optimising them in a mix on a probability simplex, and training and evaluating on the filtered corpora. An LLM-judge scores a fixed one-million-document English Common Crawl sample using 22 unique prompts, and we distil each set of scores into lightweight classifiers. We train three classifier types (fastText, a frozen embedding with an MLP head, and fine-tuned BERT) for a total of 192 models. The scores share a dominant content factor (first principal component 48.1% of variance) but can also be very distinct, e.g., conversationality is largely uncorrelated with the other tested axes. Disagreement between scores is largest among the documents different classifiers would label 'high quality'. Within-document spread across standardised scores rises from 0.39 in the bottom decile of mean score to 0.77 in the top, and classifiers with similar aggregate metrics keep different top-25% sets (Jaccard 0.47–0.61). We optimise a weighted mix of five of these scores on a probability simplex. We show a weighted sum of the raw scores performs worse than a corpus selected only with the Dolma 3 quality filter. However, putting each score on a shared rank scale and keeping documents above a Dolma 3 classifier cutoff lets those dimensions combine more effectively. Training on this latter mix, the Dolma 3 classifier filtered data, and a random crawl sample in a 0.4B dense model and a 7.7B-A0.65B mixture-of-experts model shows the the raw weighted sum does not beat Dolma 3 classified, while the rank-scaled mix does and also has some positive transference to performance outside of tested English language capabilities (including coding, mathematics, and German language).

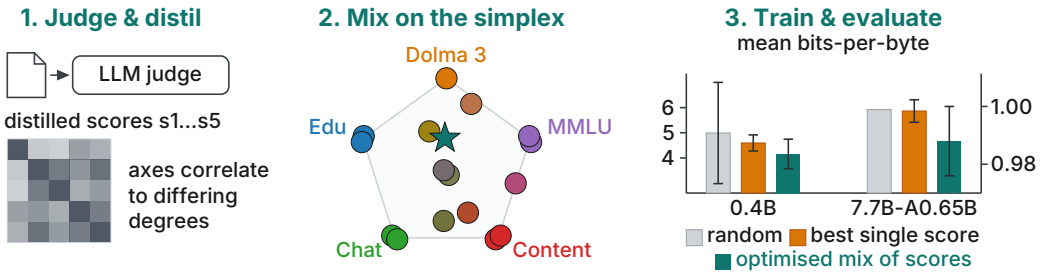

Figure 1: **Optimising a mixture of complementary quality scores.** *(1)* An LLM judge scores each web document on several axes, each distilled into a classifier The axes correlate to differing degrees, so each is its own target. *(2)* Five scores are the corners of a simplex (Dolma 3, Edu, Chat, Content, MMLU). Each interior point is one mixture, coloured by the blend of weights, and the star is the mixture we train on. *(3)* Mean bits-per-byte (lower better) for a random crawl sample, the best single score (Dolma 3), and the optimised mix. Each group has its own vertical scale Whiskers are 95% intervals over training seeds. Fig. 4 plots each benchmark.

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

1 INTRODUCTION

The performance of a pre-trained language model depends heavily on the composition of its training data, and 'quality' filtering, i.e., selecting a high-value subset of a noisy web crawl, has historically been an effective tool (Brown et al., 2020; Raffel et al., 2020; Penedo et al., 2024; Li et al., 2024). The dominant paradigm trains a lightweight classifier to separate a curated positively-labelled corpus (Wikipedia, books, chat-like data, or LLM-labelled 'educational' text) from random web documents, then keeps documents the classifier scores highly. More recent work replaces the classifier with a fine-tuned language model that rates subtler aspects of quality (Sachdeva et al., 2024; Wettig et al., 2024).

Across these methods, quality is operationalised in mutually inconsistent ways, e.g., 'educational value' vs. 'chat-likeness', and the field largely treats them as proxies for an implicit underlying property. Yet each positive set encodes an abductive commitment about which ultimate model capabilities ought to matter.

As a motivation and starting point, we report in Appendix A on an experiment where we trained matched 7.7B-A0.65B MoE models on the top and bottom score quartile of scores from 16 unique English quality classifiers trained to separate a curated corpus from random web documents. Among these classifiers, we include the Dolma 3 fastText quality classifier[1], which forms part of one of the best-performing pre-training data curation pipelines for English that is publicly known at the time of writing (OLMo Team, 2025). We find that, among the other 15 classifiers we trained, Dolma 3 is tied for the best general aggregate and distinctly outperforms all others on coding benchmarks. The larger differences observed for other capabilities are capability swaps along a 'content' vs. 'conversational' axis. Beating a well-performing single classifier therefore does not appear trivial or to be a matter of swapping in another one-dimensional classifier. This experiment also shows us that specific individual 'quality' dimensions can have higher fidelity than a 'positive' vs. random split, which motivates us in this work to attempt to measure these dimensions in more nuanced ways and to optimise an explicit mixture of scores in those dimensions.

We organise the paper around the three stages of Fig. 1: *judge and distil* the 'quality'-relevant constructs for scoring documents, *mix* and optimise the mix of scores on the probability simplex, and *train and evaluate* the resulting filters on 0.4B dense a 7.7B-A0.65B mixture-of-experts (MoE) models. Our central claim is that to achieve better performance on language modelling evaluations, 'quality' should be treated as a sophisticated mixture of partly-orthogonal, partly-redundant constructs.

Judge and distil. Quality is operationalised inconsistently across the dominant classifier-based filters: an educational-value classifier distilled from Llama-3-70B annotations (Penedo et al., 2024), a 'chat-like' vs. random fastText classifier (Li et al., 2024), earlier selection against curated positive sets (Brown et al., 2020; Wenzek et al., 2020; Gao et al., 2020), and fine-tuned LLM raters that score subtler aspects (Sachdeva et al., 2024; Wettig et al., 2024). Whether these constructs agree at the document level is rarely measured, and judge reliability and bias are documented (Zheng et al., 2023; Wang et al., 2024). We score a fixed one-million-document English Common Crawl sample with an LLM judge (Qwen3-32B) under 22 prompts and distil each axis into 192 lightweight classifiers across three backends (§2). The judged space is low-dimensional but plural. Score disagreements concentrate in the filtered top tail, and high score agreements between backends or constructs does not necessarily translate to high agreements on which documents are ultimately retained and trained on. Prior fairness concerns (Gururangan et al., 2022; Longpre et al., 2024) reappear at the document level as a register preference rather than an English-variety preference, which we report on.

Mix on the simplex. Combining several complementary signals rather than thresholding a single score scalar is increasingly the practice in large-scale pipelines: Nemotron-CC ensembles classifiers (Su et al., 2025), Aleph-Alpha-GermanWeb combines heuristic and model-based filtering (Burns et al., 2026), and propella-1 annotates eighteen properties (Idahl et al., 2026). Contrastingly, target-aware methods instead reweight data towards an explicit target (Xie et al., 2023b;a). In this work, we combine five per-document scores with weights on a probability simplex (§3). Combining naïvely doesn't work since the scores aren't directly commensurable, which causes a raw convex sum to fail, while a rank-normalised, gated blend to succeed.

[1] https://huggingface.co/allenai/dolma3-fasttext-quality-classifier

Table 1: The seven prompt families, with the descriptive names used throughout the paper. 22 prompts in total (21 document-level across these seven families, plus one pairwise contrastive prompt); the general rubric alone contributes twelve 1–5 axes.

Family (name used here)	Notion of 'quality'
General rubric	12 near-independent 1–5 axes (content, register, style, ...)
Estimated provenance	which positive corpus would rate this document highly?
Sociolinguistic features	English variety, register tags, prestige, cultural markers
Classifier anchors (educational, chat-like, Wikipedia)	predicted score of an anchored classifier
Content / format score	content-focussed score; format-focussed score
Binary screens (prior work)	informative yes/no; conversational 0/1
Benchmark usefulness	1–5 usefulness/relevance/style/format per task

Train and evaluate. We materialise the mixture filters and train two architectures, a 0.4B dense and a 7.7B-A0.65B MoE model, evaluating on a bits-per-byte (BPB) suite (§4). The raw weighted sum of scores fails to beat Dolma 3, the single classifier the background experiment sweep also could not beat (Appendix A), whereas a 'a gate $\times$ rater' combination does.

Contributions. (i) Judge and distil (§2): a document-level analysis of 22 quality constructs distilled into 192 classifiers, establishing a low-but-plural latent dimensionality, an asymmetric disagreement structure that concentrates precisely where top-k filtering operates, a register (not English-variety) fairness signal, and that metric agreement between backends is a poor proxy for the corpus they select. **(ii) Mix on the simplex** (§3): a gate $\times$ rater bilinear combiner on the probability simplex, a geometric account of why a raw weighted sum of skewed scores fails (Appendix B), and a rank-normalised gated fix. **(iii) Train and evaluate** (§4): our ablations at the two scales show that a raw weighted sum does not beat Dolma 3, but the 'a gate $\times$ rater' combination does. The full judge prompts and schemas are provided in Appendix C.

2 JUDGE AND DISTIL

Annotation and distillation setup. Corpus and judge. We use a fixed one-million-row English sample of substring-deduplicated Common Crawl, preprocessed with the OLMo 3 recipe (extraction, heuristic and language filtering, and three rounds of deduplication; Appendix D). The judge throughout is Qwen3-32B, with a context limit of $40{,}960$ tokens and a per-prompt output budget from 64 tokens (binary screens) to $1{,}024$ (pairwise document comparisons). Each of the 22 prompts operationalises a different notion of 'quality' and returns structured JSON. We enforce that structure with constrained decoding using PydanticAI. Table 1 groups the prompts into seven families; full prompts and schemas are included in Appendix C. Throughout, 'quality' should be read with scare quotes: a high metric value means the judge, and by extension a classifier, more consistently reproduces a particular conceptual commitment or target to be optimised towards during training.

Distilling the judge. To test which judgements are detectable by lightweight classifiers, which is required to apply them at corpus scale, we distil each scalar axis into a binary task: positive $=$ top-25% of judge scores, negative $=$ bottom-25%, discarding the middle two quartiles. This gives a roughly 450k-row balanced training set per task (after excluding a $1/9$ validation split). For every task we train three backends: a fastText classifier (Joulin et al., 2017); a frozen Luxical-one embedding (DatologyAI Team, 2025b) with an MLP head; and a fine-tuned `bert-base-uncased` (Devlin et al., 2019). Including a set of categorical tasks (provenance, English variety) this yields 192 models. We report positive-class F1 at threshold 0.5 and AUC for binary tasks, and macro-F1/accuracy for categorical tasks. All classifier-side analysis applies the trained models to a new random one-million-document sample of Common Crawl, independent of the sample used for LLM-judging. A high metric means the classifier can reproduce the judge. Full training configurations and final validation metrics for all three backends on a representative construct are in Appendix E.

The axes correlate in blocks. Document-level Spearman correlations (Fig. 2, left; $n = 99{,}623$) group the judge axes into a content-and-anchor block, an isolated conversational axis, an anti-correlated non-native-English block, and a tight code/maths block. The space is low-dimensional but plural: a content factor dominates, and several axes barely covary with it, indicating choosing one over another makes large difference in which documents are selected for training (even if they

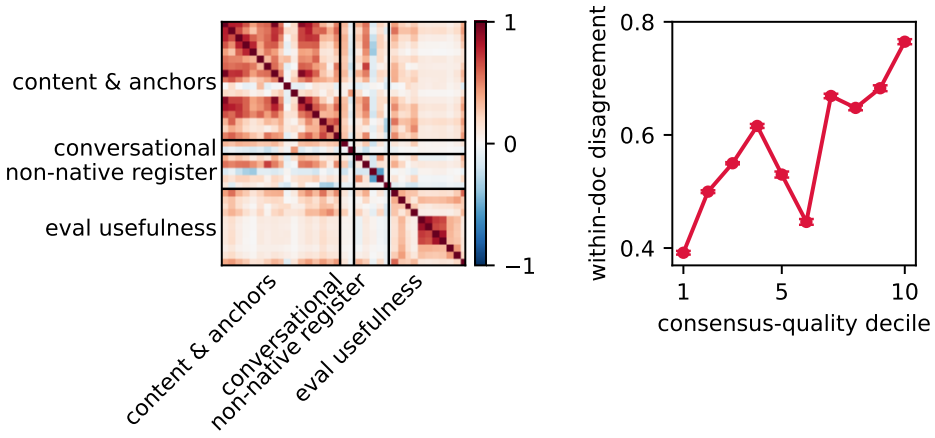

Figure 2: **Quality is low-dimensional but not one-dimensional, and it is contested exactly where filtering acts. Left:** document-level Spearman correlations across the judge axes ($n = 99{,}623$) show clear block structure: a large content-and-anchor block, an isolated `conversational` axis, an anti-correlated non-native-English block, and a tight code/maths (eval-usefulness) block. **Right:** mean within-document disagreement (standard deviation of nine z-scored quality signals) by consensus-quality decile, with 95% credible-interval error bars, is asymmetric rather than U-shaped, rising from 0.39 at the bottom decile to 0.77 at the top. Documents fail diverse positive constructs jointly but satisfy different ones.

perform similarly). The educational, Wikipedia, and chat-like anchors are the sharpest case of these differences. Academic documents score high on the educational and Wikipedia anchors and low on the chat-like anchor, and chat-like text reverses that ordering (Appendix F, Table 8). Appendix F records the marginal statistics and a principal components analysis behind this picture, including the moderate agreement of one judge re-rating a common construct as measured by similar prompts.

Disagreement peaks where filtering acts. Across constructs, disagreement is asymmetric (Fig. 2, right): 0.39 in the lowest consensus decile and 0.77 in the highest. Documents that fail many positive constructs fail them together; documents that satisfy some tend to satisfy only a few different ones. Top-k retention therefore decides in the region where the constructs disagree most.

Distilling the judge. Of the 60 binary distillation tasks, only 21 show learnable signal (Luxical AUC ≥ 0.60); the other 39 sit around chance across all three backends (Table 2). The unlearnable tasks are those which had near-constant scores from the judging stage, including every code/maths benchmark-usefulness task and every `format_match` dimension. Where a signal exists, the ordering of how well the backends reproduce the full judge is BERT $>$ Luxical $>$ fastText. The distilled scores keep the same plurality: backends trained on one learnable axis agree in rank and still retain different top-25% sets (Appendix F).

Further audits. Anchored quality tracks prestige register, with little difference by English variety, and roughly a third of judge-score variance is predictable from surface features such as length, a floor the distilled classifiers clear (Appendix F). The same anchor split is visible inside the trained models: academic and casual vocabulary push the educational and chat-like classifiers in opposite directions, and the BERT signal forms in the last encoder layers (Appendix G).

3 Mix on the simplex

A convex combination of scores. If several quality constructs are partly-orthogonal, a natural hypothesis is that a weighted sum of them beats any single one. The single one to beat is Dolma 3: in

Table 2: Distillation metrics for representative binary tasks on a new random sample of Common Crawl. Each row names the judge axis being distilled (top-25% vs bottom-25% binarisation). The upper block appears learnable. The lower block carries no learnable signal because its labels are near-constant or heavily skewed. Backend order where signal exists is BERT > Luxical > fastText. In the learnable block, the best backend per row (by F1) is in **bold** and the second best is underlined.

Binary task	fastText F1	Luxical F1 / AUC	BERT F1 / AUC
Educational anchor	0.865	0.913 / 0.971	**0.945** / 0.987
Wikipedia anchor	0.830	0.861 / 0.933	**0.886** / 0.954
Chat-like anchor	0.801	0.830 / 0.914	**0.880** / 0.953
Content-focussed score	0.835	0.877 / 0.949	**0.904** / 0.970
Format-focussed score	0.748	0.803 / 0.884	**0.863** / 0.943
MMLU usefulness	0.805	0.854 / 0.932	**0.881** / 0.952
HellaSwag usefulness	0.767	0.806 / 0.897	**0.860** / 0.942
TriviaQA usefulness	0.781	0.829 / 0.911	**0.869** / 0.942
PIQA usefulness	0.524	**0.620** / 0.709	**0.620** / 0.719
MMLU-Pro usefulness	0.554	0.454 / 0.599	0.447 / 0.598
HumanEval usefulness	0.496	0.422 / 0.534	0.224 / 0.532
MATH (Minerva) usefulness	0.496	0.460 / 0.517	0.664 / 0.518
Benchmark format match (all)	∼0.50	∼0.50	∼0.50

the background sweep of Appendix A its top quartile is tied for the best English general aggregate and has the lowest code bits-per-byte of the sixteen classifiers, so another off-the-shelf dimension is not a route past it. We test H1: *there is a weighted sum of classifier scores, including Dolma 3-high, that outperforms Dolma 3-high alone.* We take five per-document scores in $[0, 1]$: Dolma 3 $P(\text{high_quality})$ and four Luxical heads (educational anchor, chat-like anchor, content-focussed score, MMLU usefulness), chosen to bias towards the most distillable axes, cover apparently independent latent factors, and stay comparable to DCLM- and FineWeb-Edu-style classifiers. We sample 21 interior weight vectors from a $\mathrm{Dirichlet}(\alpha{=}1)$ over the five-simplex plus the four Luxical one-hot vertices (25 vectors total; the Dolma 3 vertex is available separately as Dolma 3-high), combine the *raw* scores by a null-aware convex weighted mean (Eq. equation 2), and keep the top-25% by cl100k token mass (only $\sim$14–17% of rows, since kept documents are longer). We train one 400M dense model per vector (Chinchilla $1\times$, 8.2B tokens, one seed each; the dense and 7.7B-A0.65B MoE architectures are specified in Appendix H and the evaluation suites and metric in Appendix I), fit a regression surrogate from weights to English BPB, and materialise the predicted optimum.

The surrogate's interior optimum placed all weight on the two low-noise Luxical raters, $0.45 \cdot$ content-focussed $+ 0.55 \cdot$ MMLU-usefulness (cutoff 0.858307, 72,158,178 rows retained). We call this the 'Luxical blend'. Appendix B plots the twenty-five arms in the layout of Fig. 3; the blend's downstream outcome, which does not beat Dolma 3-high, is in §4 and Fig. 10.

Distribution-aware, rank-normalised gated blending. A raw weighted sum does not give the five scores equal say. Dolma 3 is piled near zero while the Luxical heads are not, so equal weight on the simplex is not equal influence on the selected set (Appendix B). Rank-normalisation removes that mismatch: it puts every axis on one scale before the scores are combined. We rank-normalise each basis column and add a Dolma 3 gate. The combined score is

$$S(d) \;=\; \underset{a,\,b}{\mathrm{combine}}\; W_{a,b} \cdot \mathrm{rank}\big(\,\mathrm{gate}_a(d) \cdot \mathrm{rater}_b(d)\,\big), \tag{1}$$

where combine is a null-aware weighted mean, the gates a are the identity and a hard Dolma 3 gate $\mathbf{1}\{\text{Dolma 3} > \tau\}$, the raters b are the four Luxical heads and Dolma 3 itself (so $2 \times 5 = 10$ basis columns), and $\mathrm{rank}(\cdot)$ is the per-column mid-rank empirical CDF. The mid-rank makes gating safe: $\mathrm{rank}(0)$ is not 0 but ≈ 0.43, so the $\sim 86\%$ of documents tied at a gated-out 0 collapse to a single shared mid-rank and impose no spurious ordering among rejected documents; ordering among them comes only from the identity-gated columns. We freeze one recipe, the **gated rank-blend**: Dolma 3 gate threshold $\tau = 0.0216$ (the corpus-median Dolma 3 score), 65,536 rank bins, and ten fitted weights whose largest is the identity$\times$Dolma 3 term at 0.3351. The recipe is the optimum of an 18-point bilinear fleet, each arm cut to a fixed 8B-token budget, selected by a GP surrogate over the completed arms (Fig. 3).

From score to filter. The transform keeps the combined score multi-source: plotted against each raw input over a 1M-document sample it correlates with no single source above Pearson 0.69, and the Dolma 3 gate is visible as a sharp lower envelope (Appendix J). Rank-normalisation collapses the five raw supports onto a common uniform scale. We show the payoff of this in §4.

4 Train and Evaluate

The fleet behind the gated rank-blend. The weights frozen in §3.2 are the surrogate's selected optimum on the fleet in Fig. 3, fit from every other completed arm. Each arm is a point on the ten-dimensional gate × rater simplex and trains a 400M dense model on a fixed 8B-token slice of the same English crawl. The surrogate that selected the optimum averaged sixteen English benchmarks after dividing each by the identity × Dolma 3 reference. The axis in Fig. 3 is the mean of those same benchmarks in bits-per-byte, without that division, and lower is better.

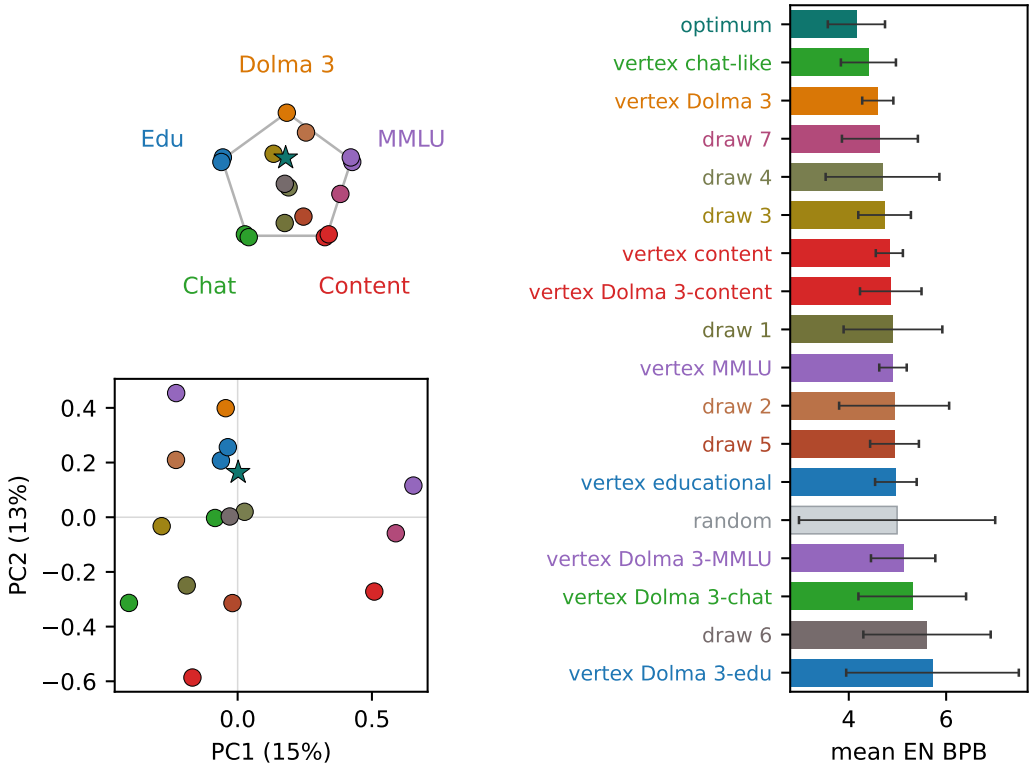

Figure 3: **The 400M gate × rater fleet.** *Top left:* each arm on a regular pentagon whose corners are the five raters (Dolma 3 at the top). An arm's weight on a rater is the sum of its identity-gate and Dolma 3-gate columns, so the point is that arm's convex combination of the corners. Identity-gate and Dolma 3-gate one-hots of one rater share a corner, a colour, and are drawn side by side. Colours are the five scores of Fig. 1: an interior arm is the blend of those colours, and the optimum is the teal star. *Bottom left:* the same arms in the leading two principal components of the ten-dimensional blend weights, with the optimum again the teal star. *Right:* mean English bits-per-byte over those sixteen benchmarks, ordered by increasing value from top to bottom, with the same colour as that arm's point. Whiskers are 95% Student-t credible intervals over the completed seeds. *random* is unfiltered Common Crawl scored the same way; it is not a point on the simplex. The Dirichlet sample is numbered draw 1–6. *optimum* is the surrogate-selected gated rank-blend. Draw 7 is the raw $0.45 \cdot$ content-focussed $+ 0.55 \cdot$ MMLU-usefulness blend. One-hot arms are labelled by their vertex; a Dolma 3- prefix marks the hard Dolma 3 gate. The Dolma 3-gate × Dolma 3 vertex is omitted, because it retains the same documents as the identity × Dolma 3 reference.

Panel 3 of Fig. 1 is one English mean per arm: the sixteen benchmarks at 400M, and those same benchmarks at the MoE with WinoGrande left out. Fig. 4 plots each benchmark in that mean. On the mean the optimised mix is the lowest of the three arms at both scales: 4.157 against Dolma 3 at 4.596 and a random crawl at 4.992 (400M, four seeds each), and 0.988 against 0.998 and 0.999 at the MoE (Dolma 3 and the mix, two seeds; random, one). The pairwise posteriors of Appendix I.5, with a region of practical equivalence of $\pm 1\%$ of the two means, put the mix ahead of Dolma 3 at $P(\text{better}) = 0.91$ at 400M. At the MoE that comparison is $P(\text{better}) = 0.62$, $P(\text{ROPE}) = 0.36$, $P(\text{worse}) = 0.02$. The mix is ahead of random at 0.84 (400M) and 0.70 (MoE).

We ask whether that English selection transfers to another language (Fig. 5). The filters are fit on English Common Crawl only, so Global-MMLU, PIQA, HumanEval, and MBPP are an out-of-distribution check: a gain on German is positive transfer, and a rise in German bits-per-byte is a cross-lingual regression.

The German aggregate is the unweighted mean of those four benchmarks, over seeds that scored all four. At 400M it is 4.505 for the mix (three seeds; one seed's final eval is missing Global-MMLU), 4.584 for Dolma 3, and 5.220 for random (four seeds each). Both filters lean ahead of random, $P(\text{better}) = 0.75$ for the mix and 0.76 for Dolma 3, and the mix does not separate from Dolma 3: $P(\text{better}) = 0.54$, $P(\text{ROPE}) = 0.10$, $P(\text{worse}) = 0.37$. The lean towards the mix is in the two QA tasks ($P(\text{better than Dolma 3}) = 0.70$). On German code the mix sits behind Dolma 3 ($P(\text{worse}) = 0.70$), while Dolma 3 leans ahead of random ($P(\text{better}) = 0.77$). At the MoE the four-benchmark mean is 0.827 for the mix, 0.852 for Dolma 3, and 0.847 for the single random seed. The mix leans ahead of Dolma 3 ($P(\text{better}) = 0.77$) and of random ($P(\text{better}) = 0.76$). Dolma 3 against that random seed is $P(\text{better}) = 0.26$, $P(\text{ROPE}) = 0.30$, $P(\text{worse}) = 0.45$. The MoE edge is again in QA ($P(\text{mix better than Dolma 3}) = 0.88$); German code is spread across the trichotomy (0.41, 0.31, 0.27). Transfer is partial. The English-selected mix keeps German bits-per-byte at or below the unfiltered crawl, and at the MoE it is the lowest of the three. At 400M the four-benchmark mean does not improve on Dolma 3, because the code tasks go the other way.

Bits-per-byte across matched fleets. We report the downstream results in aggregate bits-per-byte (BPB; lower is better). BPB is comparable within a matched fleet but not across fleets, which differ in tokeniser and evaluation suite.

Raw mixing does not beat the best single score. We confirmed the Luxical blend of §3.1 against Dolma 3-high at 400M dense ($n = 5$) and 7.7B-A0.65B MoE (Appendix B, Fig. 10). It did not beat Dolma 3-high. At 400M dense the blend was weakly worse on EN-overall BPB (4.551 vs 4.410) and on Code (5.248 vs 5.124); at 7.7B-A0.65B MoE it was decisively worse on EN-overall (0.946 vs 0.888), with posterior $P(\text{blend worse})$ 0.98–0.99 on Code, language modelling, and EN-overall. Combining the strongest constructs by a raw weighted sum therefore did not beat committing to Dolma 3, the single classifier the background sweep could not displace (Appendix A), a failure that the score geometry of Appendix B explains.

5 Discussion

Judge and distil. Quality is a vector of partly-orthogonal, partly-redundant constructs. The judged space is low-dimensional (a dominant content factor) but not one-dimensional (conversationality and the chat-like anchor are near-orthogonal to it, and anchors encode independent targets). Disagreement is concentrated in the upper tail, where top-k filtering operates, so the choice of construct matters most exactly where it is hardest, and metric agreement between backends does not imply agreement on the corpus they select.

Mix on the simplex. Combining constructs is beneficial only once their score distributions are made commensurable: a raw weighted sum is partly an optimisation over distribution shape and fails, while a rank-normalised gated blend, differing only in making the axes commensurable, succeeds. The operative choices are which construct to commit to and how to combine them. This multi-dimensional reading of quality is corroborated by concurrent large-scale data pipelines that combine several complementary signals rather than threshold a single score (Su et al., 2025; Burns et al., 2026; Idahl et al., 2026). Our checks add the mechanism (partly-orthogonal constructs that disagree in the filtered tail) and the condition (distributional commensurability) under which such combinations are most beneficial.

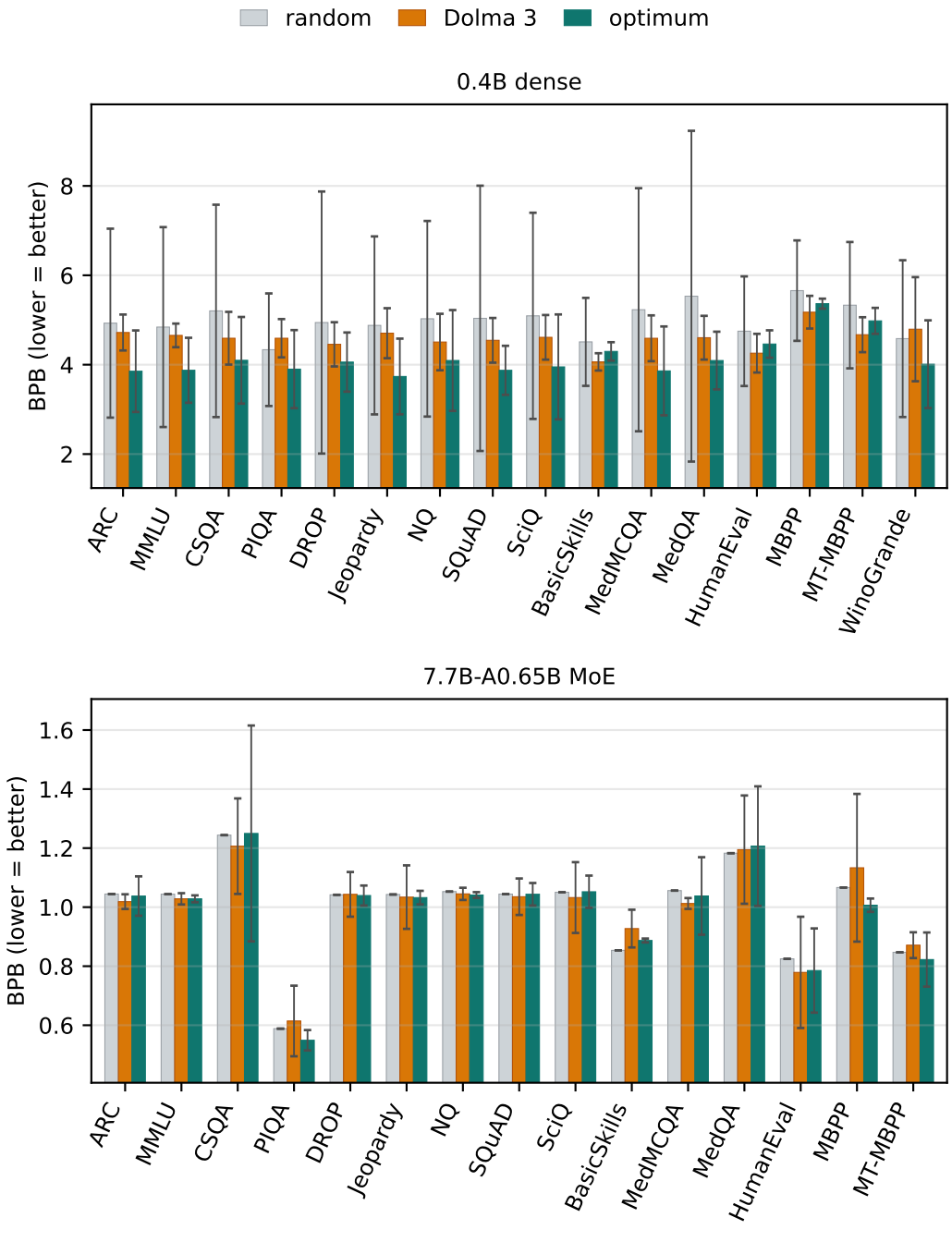

Figure 4: **Each benchmark inside Fig. 1, panel 3.** *Top:* the sixteen English benchmarks of the 400M fleet, in order: the twelve QA tasks (ARC through MedQA), then HumanEval, MBPP, and MT-MBPP, then WinoGrande. *Bottom:* the fifteen benchmarks of that MoE mean, in the same order as the top panel, with WinoGrande left out. Random is one seed; Dolma 3 and the optimum are two. Whiskers are 95% Student-t credible intervals over seeds.

Train and evaluate. The background sweep (Appendix A) is the comparison the mixtures are asked to beat: at 7.7B-A0.65B MoE, a single construct's top quartile reshapes capability along the same 'content' vs. 'conversational' axis the audit isolates (§2.2), and Dolma 3's top quartile is the strongest such arm. A raw weighted sum still fails to beat that arm. The rank-scaled mix is the lowest English

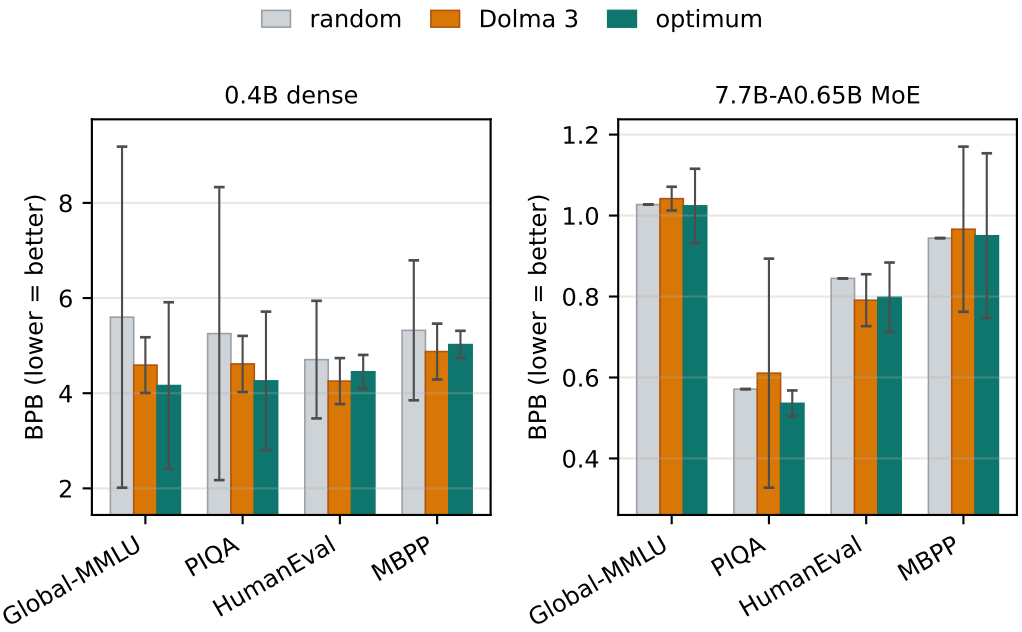

Figure 5: **German benchmarks for the arms in Fig. 4.** An out-of-distribution check of transfer from an English-only filter. Global-MMLU, PIQA, HumanEval, and MBPP. *Left:* the 400M fleet. Random and Dolma 3 are four seeds. The optimum is four seeds on PIQA, HumanEval, and MBPP, and three on Global-MMLU: one seed's final eval is missing that benchmark. *Right:* the 7.7B-A0.65B MoE. Random is one seed; Dolma 3 and the optimum are two. Whiskers are 95% Student-t credible intervals over seeds.

mean of Fig. 1 at both scales. At 400M it is ahead of Dolma 3 ($P(\text{better}) = 0.91$). At the MoE the gap sits on the $\pm 1\%$ band ($P(\text{better}) = 0.62$, $P(\text{ROPE}) = 0.36$, $P(\text{worse}) = 0.02$). The same arms, scored on German, are an out-of-distribution check of transfer (Fig. 5). The mix keeps German bits-per-byte at or below a random crawl. It leads Dolma 3 on the four-benchmark German mean at the MoE ($P(\text{better}) = 0.77$) and does not separate from Dolma 3 at 400M ($P(\text{better}) = 0.54$), where German code favours Dolma 3.

6 Limitations

We used a single judge, a single language, and a single crawl. Judge reliability and bias are documented (Zheng et al., 2023; Wang et al., 2024), and multi-judge replication will be an important aspect of future work. Rather than fidelity to any ground truth, the distillation metrics measure fidelity to the judge. So, a high-AUC classifier has learnt to reproduce the judge. The ablations are small-scale (400M dense and 7.7B-A0.65B MoE) and seed-limited (one seed per weighted-sum arm; $n \leq 5$ elsewhere). The finding that 'high quality is contested' may partly reflect having under-constructed explicit negative-quality prompts. The estimated-provenance and sociolinguistic prompts exhibited schema drift. Finally, the weighted-sum and distribution-aware blends are not a controlled A/B: they differ in gate, normalisation, and token-budget rule, so the distribution-aware blend's success localises the fix to 'commensurability' partly in combination with the geometry argument of Appendix B.

7 Conclusion

We audited what LLM-judged quality filters select, distilled 22 constructs into 192 classifiers, and showed that the operative choices in quality filtering are which construct to commit to and how to combine constructs, given that their score distributions must first be made commensurable.

AI USE STATEMENT

We used generative AI tools as coding assistants for data-processing and plotting scripts, and for copy-editing prose. All such output was reviewed by the authors. The LLM judge (Qwen3-32B) and the distilled classifiers are objects of study, described in § 2.1–2.3 and Appendix C. We take responsibility for the final content of this work.

ETHICS STATEMENT

This work studies data-selection filters, which have documented fairness implications (Gururangan et al., 2022; Longpre et al., 2024). Our §2.3 analysis is itself a positive contribution in the same vein, and we find a register preference rather than an English-variety preference. We recommend register-aware auditing before large training runs.

REPRODUCIBILITY STATEMENT

The corpus, judge, and prompt families are specified in §2.1 and Appendix C. The distillation protocol and held-out evaluation in §2.1 and Appendix F. The ablation model architectures in Appendix H. the evaluation suites, metric, and statistics in Appendix I. The blending recipes, including Eq. equation 1 and the frozen gated rank-blend parameters, in § 3.1–3.2, and the raw weighted-sum sweep in Appendix B. The single-classifier background experiment in Appendix A. The full judge prompts and schemas are included in Appendix C.

REFERENCES

Joshua Ainslie, James Lee-Thorp, Michiel de Jong, et al. GQA: Training generalized multi-query transformer models from multi-head checkpoints. In *Proceedings of EMNLP*, 2023. arXiv:2305.13245.

Jacob Austin, Augustus Odena, Maxwell Nye, et al. Program synthesis with large language models. *arXiv preprint arXiv:2108.07732*, 2021.

Iz Beltagy, Matthew E. Peters, and Arman Cohan. Longformer: The long-document transformer. *arXiv preprint arXiv:2004.05150*, 2020.

Yonatan Bisk, Rowan Zellers, Ronan Le Bras, Jianfeng Gao, and Yejin Choi. PIQA: Reasoning about physical commonsense in natural language. In *Proceedings of AAAI*, 2020. arXiv:1911.11641.

Sid Black, Stella Biderman, Eric Hallahan, et al. GPT-NeoX-20B: An open-source autoregressive language model. In *Proceedings of BigScience Workshop on Challenges & Perspectives in Creating Large Language Models*, 2022. arXiv:2204.06745.

Tom B. Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, et al. Language models are few-shot learners. *Advances in Neural Information Processing Systems*, 33, 2020. arXiv:2005.14165.

Thomas F Burns, Letitia Parcalabescu, Stephan Waeldchen, Michael Barlow, Gregor Ziegltrum, Volker Stampa, Bastian Harren, and Björn Deiseroth. Aleph-Alpha-GermanWeb: Improving German-language LLM pre-training with model-based data curation and synthetic data generation. In *Proceedings of the 19th Conference of the European Chapter of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 1267–1283. Association for Computational Linguistics, 2026. doi: 10.18653/v1/2026.eacl-long.58.

Mark Chen, Jerry Tworek, Heewoo Jun, et al. Evaluating large language models trained on code. *arXiv preprint arXiv:2107.03374*, 2021.

Pavel Chizhov, Mattia Nee, Pierre-Carl Langlais, and Ivan P. Yamshchikov. What the HellaSwag? on the validity of common-sense reasoning benchmarks. *arXiv preprint arXiv:2504.07825*, 2025.

Karl Cobbe, Vineet Kosaraju, Mohammad Bavarian, et al. Training verifiers to solve math word problems. *arXiv preprint arXiv:2110.14168*, 2021.

Damai Dai, Chengqi Deng, Chenggang Zhao, et al. DeepSeekMoE: Towards ultimate expert specialization in mixture-of-experts language models. In *Proceedings of ACL*, 2024. arXiv:2401.06066.

DatologyAI Team. BeyondWeb: Lessons from scaling synthetic data for trillion-scale pretraining. *arXiv preprint arXiv:2508.10975*, 2025a.

DatologyAI Team. Luxical: High-speed lexical-dense text embeddings. *arXiv preprint arXiv:2512.09015*, 2025b.

DeepSeek-AI. DeepSeek-V3 technical report. *arXiv preprint arXiv:2412.19437*, 2024.

Mostafa Dehghani, Josip Djolonga, Basil Mustafa, et al. Scaling vision transformers to 22 billion parameters. In *Proceedings of ICML*, 2023. arXiv:2302.05442.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of NAACL*, 2019. arXiv:1810.04805.

William Fedus, Barret Zoph, and Noam Shazeer. Switch transformers: Scaling to trillion parameter models with simple and efficient sparsity. *Journal of Machine Learning Research*, 2022. arXiv:2101.03961.

Leo Gao, Stella Biderman, Sid Black, et al. The Pile: An 800GB dataset of diverse text for language modeling. *arXiv preprint arXiv:2101.00027*, 2020.

Gemma Team. Gemma 2: Improving open language models at a practical size. *arXiv preprint arXiv:2408.00118*, 2024.

Dirk Groeneveld, Iz Beltagy, Pete Walsh, et al. OLMo: Accelerating the science of language models. In *Proceedings of ACL*, 2024. arXiv:2402.00838.

Yuling Gu, Oyvind Tafjord, Bailey Kuehl, et al. OLMES: A standard for language model evaluations. *arXiv preprint arXiv:2406.08446*, 2024.

Suchin Gururangan, Dallas Card, Sarah K. Dreier, et al. Whose language counts as high quality? measuring language ideologies in text data selection. In *Proceedings of EMNLP*, 2022. arXiv:2201.10474.

Dan Hendrycks, Collin Burns, Steven Basart, et al. Measuring massive multitask language understanding. In *Proceedings of ICLR*, 2021. arXiv:2009.03300.

Jordan Hoffmann, Sebastian Borgeaud, Arthur Mensch, et al. Training compute-optimal large language models. *arXiv preprint arXiv:2203.15556*, 2022.

Maximilian Idahl, Benedikt Droste, Björn Plüster, and Jan Philipp Harries. propella-1: Multi-property document annotation for LLM data curation at scale. 2026. arXiv:2602.12414; OpenReview 17tLF0kWFH.

Albert Q. Jiang, Alexandre Sablayrolles, Arthur Mensch, et al. Mistral 7B. *arXiv preprint arXiv:2310.06825*, 2023.

Albert Q. Jiang, Alexandre Sablayrolles, Antoine Roux, et al. Mixtral of experts. *arXiv preprint arXiv:2401.04088*, 2024.

Keller Jordan, Yuchen Jin, Vlado Boza, et al. Muon: An optimizer for hidden layers in neural networks. 2024. `https://kellerjordan.github.io/posts/muon/`.

Armand Joulin, Edouard Grave, Piotr Bojanowski, and Tomas Mikolov. Bag of tricks for efficient text classification. In *Proceedings of EACL*, 2017. arXiv:1607.01759.

John K. Kruschke. Rejecting or accepting parameter values in bayesian estimation. *Advances in Methods and Practices in Psychological Science*, 1(2):270–280, 2018.

Dmitry Lepikhin, HyoukJoong Lee, Yuanzhong Xu, et al. GShard: Scaling giant models with conditional computation and automatic sharding. In *Proceedings of ICLR*, 2021. arXiv:2006.16668.

Jeffrey Li, Alex Fang, Georgios Smyrnis, et al. DataComp-LM: In search of the next generation of training sets for language models. *arXiv preprint arXiv:2406.11794*, 2024.

Shayne Longpre, Gregory Yauney, Emily Reif, et al. A pretrainer's guide to training data: Measuring the effects of data age, domain coverage, quality, and toxicity. In *Proceedings of NAACL*, 2024. arXiv:2305.13169.

Ian Magnusson, Akshita Bhagia, Valentin Hofmann, et al. Paloma: A benchmark for evaluating language model fit. In *Advances in Neural Information Processing Systems, Datasets and Benchmarks Track*, 2024. arXiv:2312.10523.

Ian Magnusson, Nguyen Tai, Ben Bogin, David Heineman, Jena D. Hwang, Luca Soldaini, Akshita Bhagia, Jiacheng Liu, Dirk Groeneveld, Oyvind Tafjord, Noah A. Smith, Pang Wei Koh, and Jesse Dodge. Datadecide: How to predict best pretraining data with small experiments. In *Forty-second International Conference on Machine Learning*, 2025. URL `https://openreview.net/forum?id=p9YlQPF8fE`.

Pratyush Maini, Skyler Seto, He Bai, et al. Rephrasing the web: A recipe for compute and data-efficient language modeling. In *Proceedings of ACL*, 2024. arXiv:2401.16380.

Prasanna Mayilvahanan, Thaddäus Wiedemer, Sayak Mallick, Matthias Bethge, and Wieland Brendel. LLMs on the line: Data determines loss-to-loss scaling laws. In *Proceedings of the 42nd International Conference on Machine Learning*, 2025. arXiv:2502.12120.

OLMo Team. OLMo 3. *arXiv preprint*, 2025. arXiv:2512.13961.

Denis Paperno, Germán Kruszewski, Angeliki Lazaridou, et al. The LAMBADA dataset: Word prediction requiring a broad discourse context. In *Proceedings of ACL*, 2016. arXiv:1606.06031.

Guilherme Penedo, Quentin Malartic, Daniel Hesslow, et al. The RefinedWeb dataset for Falcon LLM: Outperforming curated corpora with web data, and web data only. *arXiv preprint arXiv:2306.01116*, 2023.

Guilherme Penedo, Hynek Kydlíček, Anton Lozhkov, et al. The FineWeb datasets: Decanting the web for the finest text data at scale. In *Advances in Neural Information Processing Systems, Datasets and Benchmarks Track*, 2024. arXiv:2406.17557.

Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. Language models are unsupervised multitask learners. *OpenAI technical report*, 2019.

Colin Raffel, Noam Shazeer, Adam Roberts, et al. Exploring the limits of transfer learning with a unified text-to-text transformer. *Journal of Machine Learning Research*, 21(140):1–67, 2020. arXiv:1910.10683.

Noveen Sachdeva, Benjamin Coleman, Wang-Cheng Kang, et al. How to train data-efficient LLMs. *arXiv preprint arXiv:2402.09668*, 2024.

Keisuke Sakaguchi, Ronan Le Bras, Chandra Bhagavatula, and Yejin Choi. WinoGrande: An adversarial winograd schema challenge at scale. 2020. arXiv:1907.10641.

Noam Shazeer. GLU variants improve transformer. *arXiv preprint arXiv:2002.05202*, 2020.

Shivalika Singh, Angelika Romanou, Clémentine Fourrier, et al. Global MMLU: Understanding and addressing cultural and linguistic biases in multilingual evaluation. *arXiv preprint arXiv:2412.03304*, 2024.

Luca Soldaini, Rodney Kinney, Akshita Bhagia, et al. Dolma: an open corpus of three trillion tokens for language model pretraining research. In *Proceedings of ACL*, 2024. arXiv:2402.00159.

Dan Su, Kezhi Kong, Ying Lin, Joseph Jennings, Brandon Norick, Markus Kliegl, Mostofa Patwary, Mohammad Shoeybi, and Bryan Catanzaro. Nemotron-CC: Transforming Common Crawl into a refined long-horizon pretraining dataset. In *Proceedings of ACL*, pp. 2459–2475, 2025. arXiv:2412.02595.

Jianlin Su, Yu Lu, Shengfeng Pan, et al. RoFormer: Enhanced transformer with rotary position embedding. *Neurocomputing*, 2024. arXiv:2104.09864.

Hugo Touvron, Thibaut Lavril, Gautier Izacard, et al. LLaMA: Open and efficient foundation language models. *arXiv preprint arXiv:2302.13971*, 2023.

Peiyi Wang, Lei Li, Liang Chen, et al. Large language models are not fair evaluators. In *Proceedings of ACL*, 2024. arXiv:2305.17926.

Guillaume Wenzek, Marie-Anne Lachaux, Alexis Conneau, et al. CCNet: Extracting high quality monolingual datasets from web crawl data. In *Proceedings of LREC*, 2020. arXiv:1911.00359.

Alexander Wettig, Aatmik Gupta, Saumya Malik, and Danqi Chen. QuRating: Selecting high-quality data for training language models. In *Proceedings of ICML*, 2024. arXiv:2402.09739.

Sang Michael Xie, Hieu Pham, Xuanyi Dong, et al. DoReMi: Optimizing data mixtures speeds up language model pretraining. In *Advances in Neural Information Processing Systems*, 2023a. arXiv:2305.10429.

Sang Michael Xie, Shibani Santurkar, Tengyu Ma, and Percy Liang. Data selection for language models via importance resampling. In *Advances in Neural Information Processing Systems*, 2023b. arXiv:2302.03169.

Rowan Zellers, Ari Holtzman, Yonatan Bisk, Ali Farhadi, and Yejin Choi. HellaSwag: Can a machine really finish your sentence? In *Proceedings of ACL*, 2019. arXiv:1905.07830.

Biao Zhang and Rico Sennrich. Root mean square layer normalization. In *Advances in Neural Information Processing Systems*, 2019. arXiv:1910.07467.

Lianmin Zheng, Wei-Lin Chiang, Ying Sheng, et al. Judging LLM-as-a-Judge with MT-Bench and Chatbot Arena. In *Advances in Neural Information Processing Systems, Datasets and Benchmarks Track*, 2023. arXiv:2306.05685.

A Background: single classifiers do not displace Dolma 3

This appendix is a precondition of the paper, not one of its results. Before the judge, the distillation, or the mixtures, we asked whether any existing one-dimensional quality classifier already suffices, and in particular whether any of them displaces Dolma 3. The answer motivates the two design choices in the main text. Individual dimensions need higher-fidelity signals than a positive-versus-random split, which is why Section 2 uses an LLM judge and distils its axes. And those dimensions have to be combined, which is why Section 3 mixes annotations instead of committing to one score. The reporting below is held to the same standard as the main text: protocol, exclusions, and the per-construct numbers.

Protocol. For each of 16 English quality classifiers we score a 500M-document Common Crawl sample (fuzzy-deduplicated) and split it into four score quartiles. We then train one 7.7B-A0.65B MoE model on the bottom quartile ('low') and one on the top quartile ('high'), holding the backbone, the matched bilingual 96k tokeniser, and the token budget fixed, and evaluate with the same OLMES suite and BPB metric as the rest of the paper (Appendix I). The two arms of a construct are therefore an extreme top-versus-bottom contrast on that construct's own score, with the low arm serving as the within-construct control. Because both arms are equal-size quartiles, they carry comparable token budgets and epoch counts, and both use the matched bilingual 96k tokeniser.

Rigor caveats. These are exploratory runs and are read differently from the blending fleets. Each arm has *one seed*, so we cannot form the across-seed Student-t posteriors of Appendix I.5; we therefore report point differences and directional counts only, and draw no significance claims. The contrast is top-versus-bottom quartile, not filter-versus-random, so it measures the *shape* of a construct's selection, not the value of filtering per se. Six further arms are excluded: two ran but were never plotted, and four crashed during training and produced no evaluation. Accuracy at this scale is near chance (Appendix I.1); every claim rests on BPB.

Per-construct aggregates. The high bucket does not systematically win. It improves the general (knowledge/QA/LM) accuracy aggregate for only $5/16$ constructs, and the effect is small either way ($|\Delta| \leq 0.06$). The signal is in code BPB, where the sign is set by which construct defines quality: content and academic constructs move code BPB down, and so does the distilled chat-like anchor (`chatlike-anchor` -0.481, `dolma-3` -0.338, `openweb-math` -0.288, `mmlu-pro` -0.266), whereas constructs learnt directly from conversational corpora move it sharply up (`tagengo-gpt4` $+0.750$, `openhermes-2.5` $+0.741$, `oasst2` $+0.599$, `eli5-st` $+0.525$). That swing is an order of magnitude larger than the general-aggregate one (Table 3). The split is the content-versus-conversational axis later isolated in the judge (Section 2.2, Table 8), here already visible as downstream capability: selecting on raw conversational corpora strips the code-adjacent web text that a standard benchmark suite rewards, while selecting for informational content preserves it.

Table 3 lists the low- and high-arm values. Among the top quartiles, Dolma 3 is tied for the best general aggregate (0.346, with `mmlu-pro`) and has the lowest code BPB (0.917). No other single classifier's top quartile improves on Dolma 3 on both aggregates. Beating Dolma 3 is therefore not available by choosing a different one-dimensional classifier. That is the bar Sections 3 and 4 take up, and it is why the paper both enriches the individual signals and mixes them.

Table 3: Single-construct 7.7B-A0.65B MoE ablations (one seed per arm): low (bottom-quartile) vs high (top-quartile) arm on the general accuracy aggregate (↑ better) and the code BPB aggregate (↓ better). Constructs use the English aggregates. The best value in a column is in **bold** and the second-best distinct value is underlined. A tie for best is shared and leaves no second place.

	General (acc, ↑)		Code BPB (↓)	
Construct	low	high	low	high
dolma-3	0.318	**0.346**	1.255	**0.917**
educational-anchor	0.328	0.326	1.107	1.138
chatlike-anchor	0.336	0.324	1.438	0.957
mmlu-usefulness	0.340	0.327	1.027	1.504
content-score	0.327	0.335	1.126	1.166
openweb-math	0.337	0.314	1.294	1.006
openhermes-2.5	0.353	0.306	0.858	1.599
webfaq-v2	0.328	0.324	1.020	1.230
mmlu-pro	0.313	**0.346**	1.334	1.068
eli5-st	**0.356**	0.299	0.992	1.517
philpapers-pile	0.335	0.309	0.970	1.410
pes2o	0.312	0.338	1.136	1.170
philpapers-malteos	0.332	0.335	1.024	1.020
tagengo-gpt4	0.341	0.312	**0.817**	1.566
oasst2	0.350	0.309	0.886	1.486
wikipedia-en	0.339	0.321	1.120	1.334

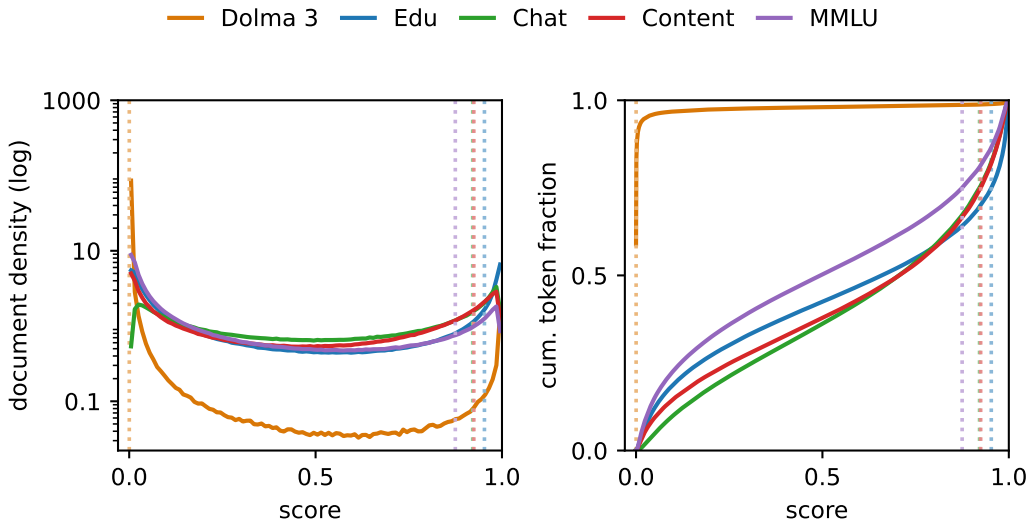

Figure 6: **Left:** overlaid document-level score densities (log scale) for the five classifiers over a 4.1M-row sample of the 500M-document annotated English corpus (each document weighted equally); vertical dotted lines mark each classifier's token-weighted top-25% cutoff (lighter shade). Dolma 3 is far more skewed (mean ~ 0.015, median ~ 0) than the Luxical heads (means 0.36–0.53). **Right:** the same five classifiers as token-weighted CDFs with the same cutoffs; the uneven gradient fields across simplex axes, and that a token-25% cut keeps only 14–17% of rows.

B Raw weighted-sum optimisation and why the simplex is not flat

Section 3.1 mixes the five scores before any rank normalisation. For a weight vector W on the five-simplex and raw scores $r_b(d)$, the mixture is the null-aware weighted mean

$$S(d) \;=\; \frac{\sum_b W_b\, r_b(d)\, \mathbf{1}\{r_b(d) \text{ present}\}}{\sum_b W_b\, \mathbf{1}\{r_b(d) \text{ present}\}}, \tag{2}$$

where b runs over Dolma 3 and the four Luxical heads. A missing score drops out of the numerator and the denominator, so the weights that remain are still a convex combination. This is the same combiner as Eq. equation 1, applied to the raw scores, with no rank normalisation and no Dolma 3 gate.

The failure of the raw Luxical blend is a property of the score distributions, not of the constructs. Dolma 3 $P(\text{high_quality})$ is nearly a point mass: mean ~ 0.117, median ~ 0.019, with about 86% of documents tied near zero, whereas the Luxical heads have means 0.36–0.53 (Fig. 6). Under a raw weighted mean followed by a top-k cut, equal weight in the simplex does not mean equal influence on the selected set: a severely skewed axis contributes almost no ordering information near its mass point, so the surrogate's gradient field is dominated by the axis asymmetry rather than by construct content. The realised effect is visible in the 25 arms, whose score cutoffs span 0.390–0.954 and whose retained-row counts span 70.1M–85.8M for a nominal top-25%-by-token target (Table 4). Any optimiser over these raw axes is partly optimising over distribution shape, which rank-normalisation removes. Fig. 7 plots this cutoff and retained-row field over the simplex.

Table 4 lists the 25 weight vectors of the weighted-sum sweep (Section 3.1): 21 interior Dirichlet(α=1) draws over the five-simplex plus the four Luxical one-hot vertices. For each arm we report the five weights (Dolma 3, educational anchor, chat-like anchor, content-focussed score, MMLU-usefulness), the score cutoff, retained-row count for the nominal top-25%-by-token target, and the mean English BPB of the trained 400M dense model. The spread in cutoffs (0.390–0.954) and retained rows (70.1M–85.8M) for a fixed token target is the concrete signature of the score-geometry problem above.

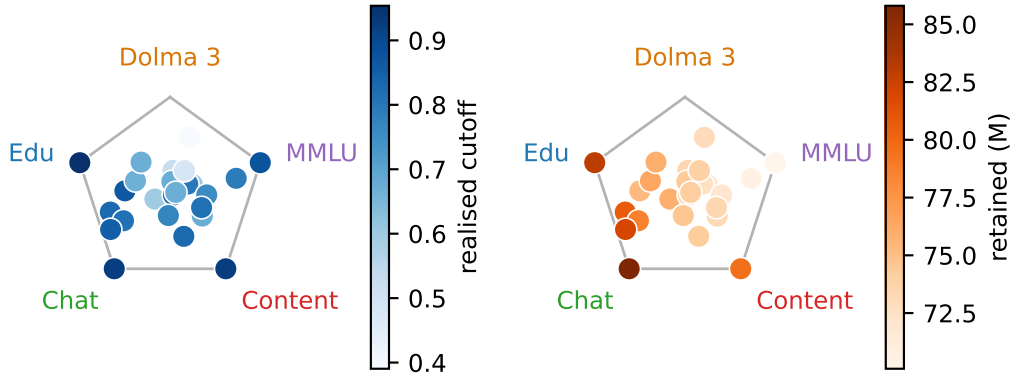

Figure 7: **The simplex is not flat.** The 25 arms projected onto a pentagon whose corners are the five one-hot sources, coloured by the realised score cutoff (left) and retained-row count (right) for a *fixed* top-25%-by-token budget. The cutoff ranges 0.39 (near the Dolma 3 corner) to 0.95 (near the Luxical corners) and retained rows range 70.1M–85.8M: identical token targets, very different operating points.

Table 4: The 25 weighted-sum weight vectors, realised cutoffs, retained rows, and mean BPB (400M dense, one seed each). Rows are the 21 Dirichlet draws followed by the four one-hot vertices. Weights are over Dolma 3, the educational anchor (Edu), the chat-like anchor (Chat), the content-focussed score (Content), and MMLU-usefulness (MMLU).

Arm	Dolma 3	Edu	Chat	Content	MMLU	cutoff	retained rows	mean BPB
Draw 1	0.288	0.111	0.078	0.306	0.218	0.6099	72,184,766	5.303
Draw 2	0.072	0.418	0.452	0.058	0.000	0.8304	80,714,143	4.962
Draw 3	0.080	0.304	0.015	0.236	0.364	0.7970	72,562,799	4.528
Draw 4	0.413	0.197	0.029	0.299	0.062	0.5183	73,398,296	5.433
Draw 5	0.120	0.143	0.035	0.343	0.359	0.7545	72,033,460	5.102
Draw 6	0.301	0.050	0.457	0.072	0.120	0.5983	75,899,145	5.639
Draw 7	0.059	0.380	0.011	0.331	0.220	0.8214	73,298,462	4.873
Draw 8	0.008	0.114	0.278	0.421	0.178	0.8292	74,240,881	5.369
Draw 9	0.237	0.049	0.049	0.601	0.065	0.6761	73,224,746	4.537
Draw 10	0.077	0.288	0.514	0.081	0.040	0.8120	78,584,099	5.124
Draw 11	0.074	0.006	0.087	0.106	0.727	0.7905	70,684,814	4.282
Draw 12	0.032	0.621	0.094	0.191	0.061	0.8648	75,254,169	5.691
Draw 13	0.545	0.049	0.084	0.014	0.308	0.3904	72,957,265	4.766
Draw 14	0.076	0.130	0.383	0.162	0.248	0.7710	74,545,202	4.953
Draw 15	0.203	0.178	0.247	0.056	0.316	0.6740	74,367,304	5.488
Draw 16	0.246	0.408	0.159	0.164	0.022	0.6667	76,398,254	5.574
Draw 17	0.246	0.437	0.118	0.029	0.171	0.6707	75,887,957	5.250
Draw 18	0.420	0.030	0.187	0.147	0.215	0.4839	73,846,769	5.380
Draw 19	0.052	0.283	0.638	0.024	0.004	0.8515	82,021,479	5.544
Draw 20	0.197	0.113	0.291	0.111	0.288	0.6717	74,109,129	5.190
Draw 21	0.016	0.064	0.276	0.157	0.487	0.8127	73,364,935	4.677
Vertex: Edu	0.0	1.0	0.0	0.0	0.0	0.9537	82,959,614	4.866
Vertex: Chat	0.0	0.0	1.0	0.0	0.0	0.9217	85,803,678	4.941
Vertex: Content	0.0	0.0	0.0	1.0	0.0	0.9248	79,492,041	4.356
Vertex: MMLU	0.0	0.0	0.0	0.0	1.0	0.8757	70,088,993	4.638

Each of those arms trains one 400M dense model. Fig. 9 places them, the Dolma 3 vertex, and the surrogate optimum, in the layout of Fig. 3. The Dolma 3 vertex is the four-seed Dolma 3-high replication. The lowest point estimate in the twenty-five is an interior draw. The regression surrogate fit to these arms placed its weight on the two low-noise Luxical raters, $0.45 \cdot$ content-focussed $+$ $0.55 \cdot$ MMLU-usefulness. That materialised optimum does not beat Dolma 3-high (Fig. 10).

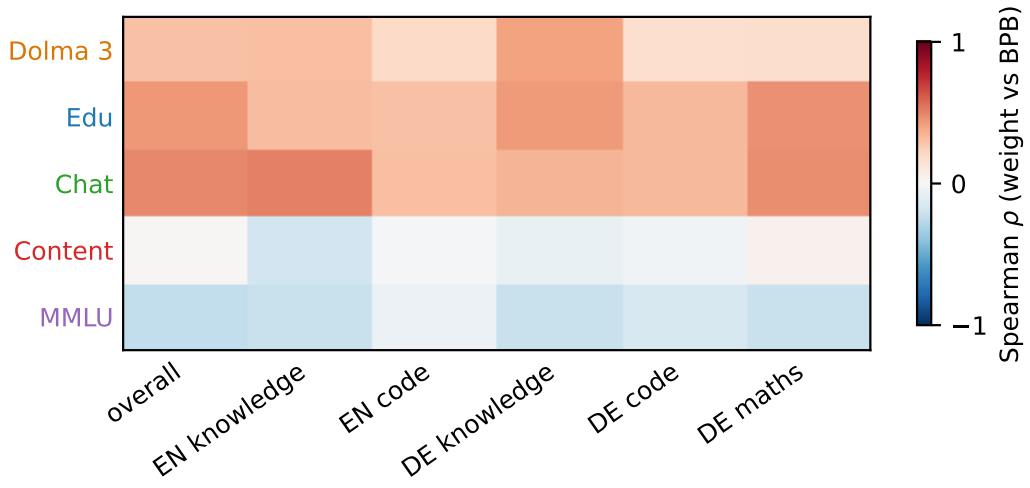

Figure 8: Spearman between each classifier's mixture weight and downstream BPB across eval aggregates. MMLU-usefulness weight helps (-0.06 to -0.25), the content-focussed score is near-neutral, and the Dolma 3/educational/chat-like weights correlate with worse BPB, driving the surrogate optimum onto the two low-noise Luxical raters.

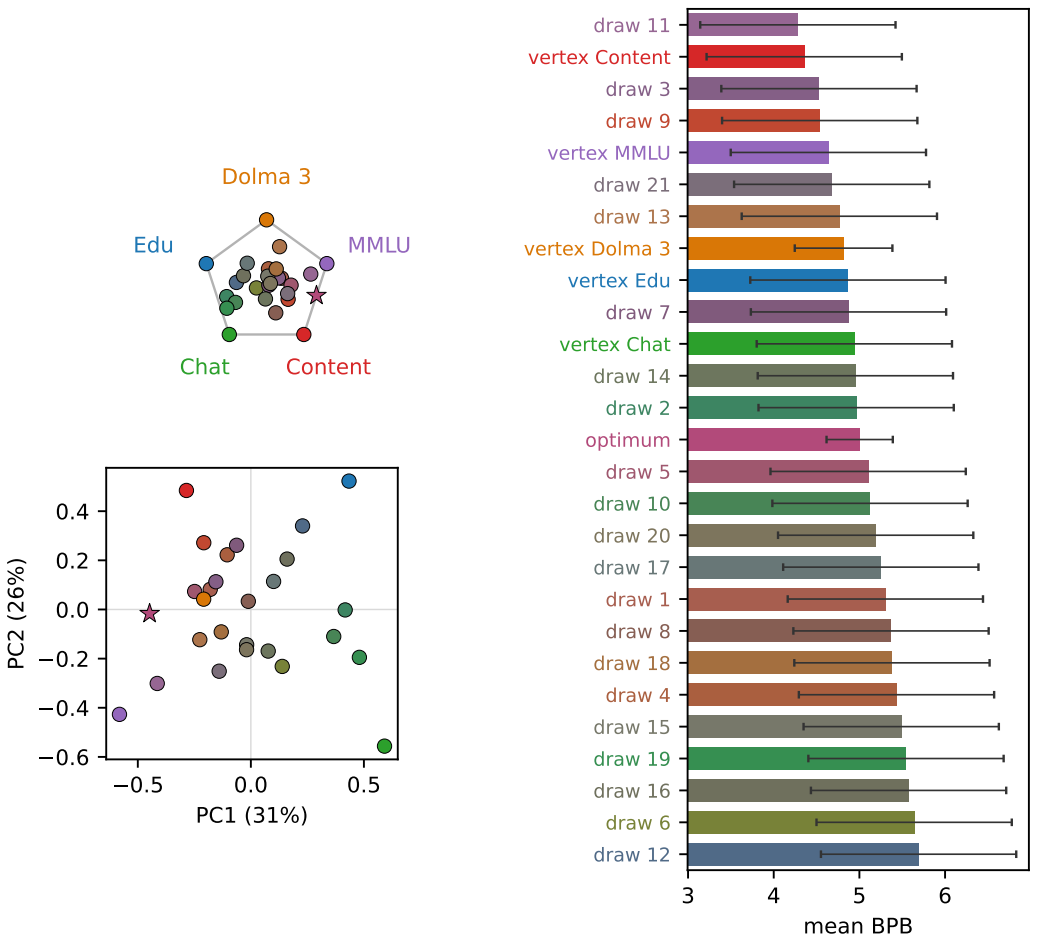

Figure 9: **The raw weighted-sum sweep, without rank normalisation.** The 25 arms of Table 4, plus the Dolma 3 vertex and the surrogate optimum. *Top left:* each arm on the same pentagon as Fig. 3, Dolma 3 at the top. An arm sits at the convex combination of the corners given by its weights in Eq. equation 2. The optimum is the star. *Bottom left:* the same arms in the leading two principal components of the sweep weights; the optimum is projected into that space and drawn as the same star. *Right:* mean bits-per-byte, ordered by increasing value from top to bottom, with a 95% Student-t interval. The Dolma 3 bar is the mean of four Dolma 3-high seeds. The optimum bar is the mean of its five confirmation seeds. Each other arm is one seed, so its interval uses that replication's standard deviation, on three degrees of freedom, as the uncertainty of a single run. A vertex takes its corner's colour; an interior arm, including the optimum, is the blend of those colours. Interior arms from the sweep are the Dirichlet draws, labelled as in Table 4.

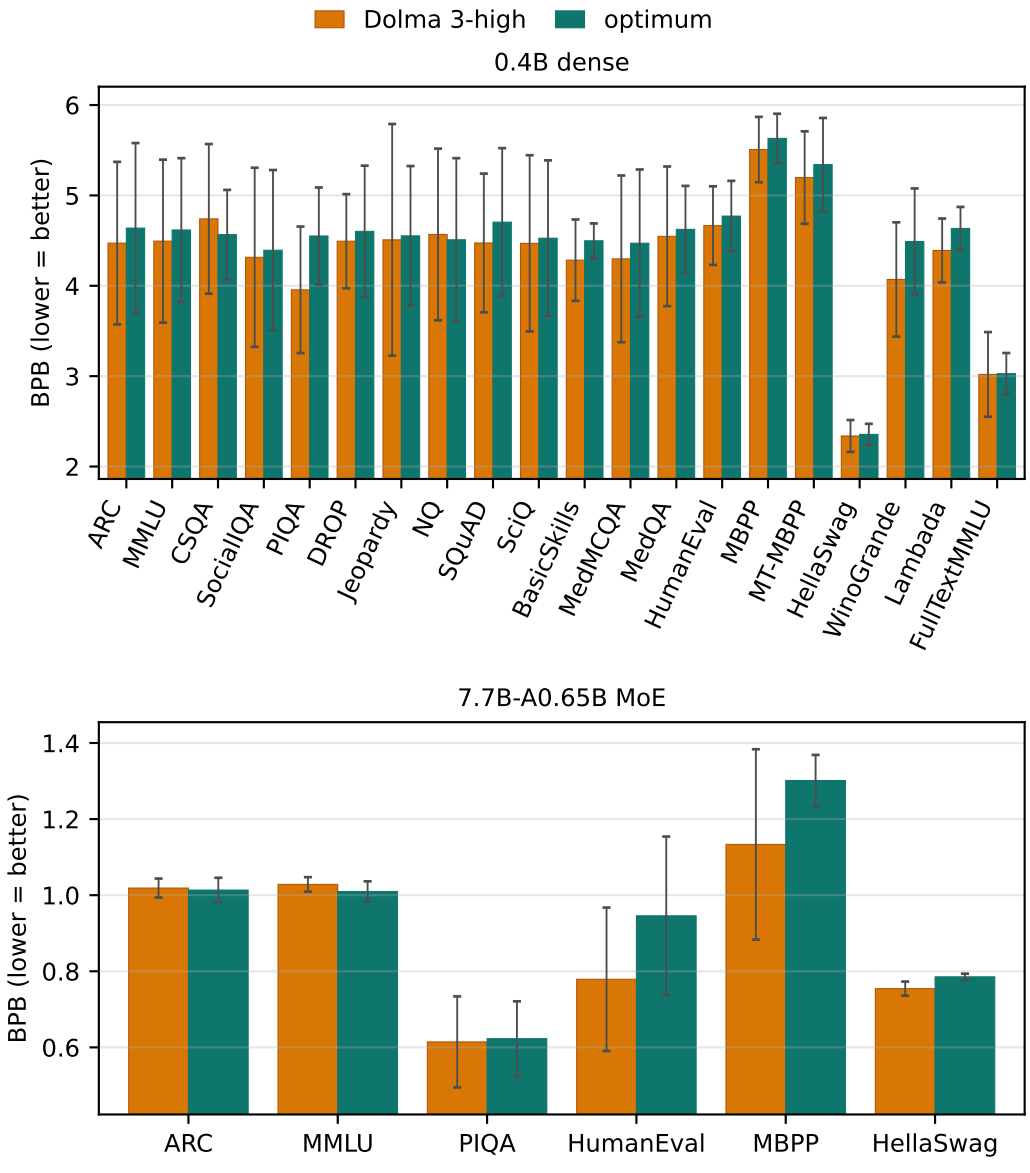

Figure 10: Per-benchmark seed-mean BPB ($\pm 95\%$ credible interval; lower is better) for the optimum $0.45 \cdot$ content-focussed $+ 0.55 \cdot$ MMLU-usefulness versus Dolma 3-high, over every English benchmark each fleet evaluates: 400M dense ($n = 5$, top) and 7.7B-A0.65B MoE (bottom, the six benchmarks its in-training evaluation emits). This is the star in Fig. 9. The optimum is marginally worse or indistinguishable at 400M and clearly worse on the code benchmarks (HumanEval, MBPP) at 7.7B-A0.65B MoE.

C PROMPT TEXTS AND SCHEMAS

Every judge prompt used in the study is reproduced below, together with the Pydantic output schema that constrained decoding enforced. The judge returns structured JSON for every prompt. Families that share a body are given once: the three classifier anchors differ only in the positive-corpus sentence, and the eleven benchmark-usefulness prompts differ only in the task description and the three worked examples spliced into one template. `{document_text}` (for the pairwise prompt, `{document_a_text}` and `{document_b_text}`) is where the document is inserted. A brace that is not a placeholder is doubled, as in a format string. JSON field names below are the schema keys; the descriptive names used in the body of the paper (content-focussed score, format-focussed score, and so on) refer to these fields.

General rubric

```
You are an expert annotator evaluating a web document for a research
project on data curation. You will score the document on twelve
independent dimensions. Each dimension is scored 1 to 5 using the
anchors given. Score each dimension independently -- a document can
be high on one dimension and low on another.

DOCUMENT:
{document_text}

Score each dimension. Anchors:

1. informational_content
   1 = essentially no information (boilerplate, navigation, repeated
phrases, content-free)
   3 = some substantive claims, but mostly general or commonplace
   5 = dense with specific factual claims, named entities, dates,
numbers, citations

2. educational_value
   1 = nothing a learner could acquire from this document
   3 = explains something at a shallow level, or restates known
material
   5 = clearly and accurately teaches a non-trivial concept,
procedure, or fact

3. coherence
   1 = disjointed, non-sequiturs, garbled
   3 = readable but with gaps, repetition, or weak structure
   5 = tightly structured, each sentence advances the argument or
narrative

4. linguistic_quality
   1 = many grammatical errors, ungrammatical fragments dominate
   3 = mostly grammatical with some errors or awkwardness
   5 = fluent, idiomatic, no errors

5. orthography
   1 = pervasive spelling, punctuation, or capitalisation errors
   3 = occasional errors
   5 = clean throughout

6. structural_format
   1 = plain prose with no structural markers
   3 = some structure (paragraph breaks, occasional list)
   5 = heavy structure: headers, bullets, numbered lists, tables, or
Q&A blocks

7. formality_register
```

```
   1 = casual conversational, slang, informal address
   3 = neutral journalistic or general-audience prose
   5 = highly formal academic, legal, or technical register

8. specificity
   1 = vague generalities, no concrete examples
   3 = mixed: some specifics, some hand-waving
   5 = concrete examples, named cases, specific numbers throughout

9. epistemic_hedging
   1 = strong, confident assertions with no qualifiers
   3 = mixed confidence, occasional hedging
   5 = heavily hedged, `may`, `could`, `some researchers suggest`,
etc., throughout

10. instruction_likeness
    1 = no imperative or instructional content
    3 = some how-to or imperative elements
    5 = primarily instructions, how-to, or step-by-step procedure

11. dialogue_likeness
    1 = single-voice monologue or exposition
    3 = some quoted speech or implied addressee
    5 = explicit multi-turn dialogue, Q&A pairs, or chat-like
structure

12. factual_density
    1 = no checkable factual claims
    3 = some checkable factual claims among opinion or exposition
    5 = primarily checkable factual claims (entities, dates,
statistics, citations)

Return JSON only, no other text.
```

Output schema (JSON): twelve integers, each in $\{1, \ldots, 5\}$:

```
{`informational_content`: <int>, `educational_value`: <int>, `coherence`: <int>,
 `linguistic_quality`: <int>, `orthography`: <int>, `structural_format`: <int>,
 `formality_register`: <int>, `specificity`: <int>, `epistemic_hedging`: <int>,
 `instruction_likeness`: <int>, `dialogue_likeness`: <int>, `factual_density`: <int>}
```

Estimated provenance

```
You will see a web document. Assume a quality classifier exists that
was trained on a specific positive corpus and rates this document
highly. Identify the positive corpus most consistent with that
hypothesis.

DOCUMENT:
{document_text}

Candidate positive corpora:
A. OpenHermes-style instruction/chat data (synthetic Q&A, multi-turn
assistant transcripts)
B. r/ExplainLikeImFive -- Reddit explainer-style answers, casual but
informative
C. Wikipedia -- encyclopedic, citation-bearing prose
D. arXiv / peS2o -- academic papers, formal register, equations and
citations
E. Project Gutenberg / books -- long-form narrative or expository
prose
```

```
F. Stack Exchange -- technical Q&A with code snippets and structured
answers
G. News articles -- journalistic register, current events
H. None of the above would rate this document highly

Choose the single most likely positive corpus. Optionally choose a
secondary one if the document is consistent with two.

Return JSON only.
```

Output schema (JSON):

```
{`most_likely`: <str>, `confidence`: <int 1-5>, `secondary`: <str or null>,
 `primary_signal`: <str>}
```

The schema types `most_likely` and `secondary` as free strings rather than as an enumeration of A–H, which is the source of the schema drift reported in Appendix F.

This is the prompt behind the non-native-English block in Fig. 2. It is a descriptive annotation, not a quality score.

Sociolinguistic features

```
You will see a web document. Identify cultural and sociolinguistic
features. Do NOT judge whether the document is good or bad. This is a
descriptive annotation.

DOCUMENT:
{document_text}

Annotate the following:

1. english_variety: one of `US`, `UK`, `AU`, `CA`, `IN`, `
other_world_english`, `mixed`, `unclear`, `non_native_english`
2. cultural_geographic_markers (1-5)
3. dominant_cultural_context: if cultural_geographic_markers ≥ 3,
name the dominant context as a short phrase, else null
4. prestige_register_features (1-5)
5. register_categories: list any that apply, from {{academic,
journalistic, business_corporate, technical_documentation, legal,
religious, casual_web, social_media, marketing, fiction_narrative,
personal_blog, forum_discussion, customer_review,
instructional_how_to, encyclopedic}}
6. non_native_english_features (1-5)
7. western_cultural_reference_density (1-5)

Return JSON only.
```

Output schema (JSON):

```
{`english_variety`: <str>, `cultural_geographic_markers`: <int 1-5>,
 `dominant_cultural_context`: <str or null>, `prestige_register_features`: <int 1-5>,
 `register_categories`: [<str>, ...], `non_native_english_features`: <int 1-5>,
 `western_cultural_reference_density`: <int 1-5>}
```

As with provenance, `english_variety` and `register_categories` are free strings in the schema, not closed enumerations, which is why variety labels sometimes come back as register labels (Appendix F).

One template, three positive corpora. Negatives are random Common Crawl in every case.

Classifier anchors

```
A quality classifier has been trained with the following positive and
negative corpora:

POSITIVES: {pos_description}
NEGATIVES: random documents from a web crawl (Common Crawl)

It outputs a score from 1 (very unlikely to be from the positive set)
to 5 (very likely to be from the positive set). Predict the score
this classifier would assign to the following document.

DOCUMENT:
{document_text}

Return JSON only.
```

The three substitutions for `{pos_description}` are:

- **Chat-like.** OpenHermes 2.5 (synthetic instruction-following chat) and high-scoring r/ExplainLikeImFive answers.
- **Educational.** Documents annotated as having high educational value (clear explanations of academic topics, would help a student learn).
- **Wikipedia.** Wikipedia and academic encyclopedia articles.

Output schema (JSON), shared:

```
{`predicted_score`: <int 1-5>, `primary_reason`: <str>}
```

Content-focussed score

```
You will see a web document. Score its content quality on a 1-5
scale, IGNORING the following features completely:
- Formatting (headers, bullets, numbered lists, Q&A structure,
dialogue)
- Register (formal vs casual, hedged vs confident, academic vs
colloquial)
- Surface markers of `educational tone` (didactic openings, `let`s
explore`, etc.)

Imagine the document rendered as a single plain paragraph in a
neutral voice. Score the informational substance only.

DOCUMENT:
{document_text}

Anchors:
1 = no informational substance; the underlying content is trivial,
boilerplate, or empty
3 = moderate informational substance; the underlying content is real
but unremarkable
5 = high informational substance; specific facts, real expertise,
dense and useful content

Return JSON only.
```

Output schema (JSON):

```
{`content_blind_score`: <int 1-5>, `format_features_observed`: [<str>, ...],
 `rationale`: <str>}
```

Format-focussed score

```
You will see a web document. Score its structural format quality on a
1-5 scale, IGNORING whether the actual content is factually accurate,
interesting, or substantive.

Score only the structural and presentational features:
- Clear use of headers, paragraphs, or sectioning
- Appropriate use of lists, tables, or Q&A blocks where they aid
readability
- Consistent formatting

DOCUMENT:
{document_text}

Anchors:
1 = no structural features, or random/chaotic formatting
3 = some structural features, used appropriately for the content type
5 = strong, consistent structural features that aid navigation and
comprehension

Return JSON only.
```

Output schema (JSON):

```
{`format_blind_score`: <int 1-5>, `rationale`: <str>}
```

The one pairwise prompt in the study. The two documents are sequential draws from the crawl, not a classifier-ordered pair.

Pairwise contrastive prompt

```
You will see two web documents drawn at random from a large web crawl
sample. They may or may not differ in overall quality. Compare them
and explain the most salient differences along dimensions relevant to
text-quality classification. Do not assume either document is better
overall.

DOCUMENT A:
{document_a_text}

DOCUMENT B:
{document_b_text}

For each category below, apply only if document A differs from B
along that dimension. Tag all that apply; rank them by importance.

Categories:
- `format`: A has clearer structure (headers, lists, Q&A) than B
- `register`: A is more formal/educational in tone than B
- `factual_density`: A makes more specific factual claims than B
- `coherence`: A is better organised than B
- `language_quality`: A is more grammatical or fluent than B
- `specificity`: A has more concrete examples than B
- `instruction_format`: A is more like instruction-following or Q&A
than B
- `dialogue_format`: A is more like dialogue than B
- `topic_match`: A is on a topic more typical of high-quality corpora
than B
- `length`: A is substantially longer or shorter than B in a way
relevant to quality
```

```
- `no_clear_difference`: A and B look similar along quality-relevant
dimensions
- `other`: something else (describe in rationale)

Return JSON only.
```

Output schema (JSON):

```
{`ranked_categories`: [<str>, ...], `primary_category`: <str>, `rationale`: <str>}
```

Ask-LLM informativeness screen (Sachdeva et al., 2024)

```
DOCUMENT:
{document_text}

Does the previous document contain informative signal for
pre-training a large-language model? An informative datapoint should
be well-formatted, contain some usable knowledge of the world, and
strictly NOT have any harmful, racist, sexist, etc. content.
OPTIONS:
- yes
- no

Return JSON only.
```

Output schema (JSON):

```
{`informative_signal`: <`yes` or `no`>}
```

Conversational screen (DatologyAI Team, 2025a)

```
Your task is to classify text as either conversational (1) or
non-conversational (0).
Conversational text (1) shows:
Back-and-forth exchanges between participants
Questions followed by relevant answers
Non-conversational text (0) typically:
Presents information in a one-way format
Lacks interaction between participants
Contains formal or encyclopedic writing
Focuses on describing or explaining without dialogue
Rate each text as exactly 0 or 1.

Examples:
Text: `The mitochondria is the powerhouse of the cell. It produces
energy through cellular respiration. This process involves multiple
steps including glycolysis and the Krebs cycle.`
  Output: 0
Text: `A: Hey, I`m having trouble with my code. The function keeps
returning null. B: Can you share the error message you`re getting? A:
Here`s the stack trace: [error details] B: Ah, I see the issue. You
need to initialize the variable first. A: That fixed it, thanks!`
  Output: 1
Text: `Q: What`s the capital of France? A: The capital of France is
Paris. Q: What`s its population? A: Paris has a population of about
2.2 million people.`
  Output: 1
Text: `To install Python, first download the installer from
python.org. Run the executable and follow the installation wizard.
```

```
Make sure to check the `Add Python to PATH` option during
installation.`
  Output: 0
Text: `JavaScript was created by Brendan Eich in 1995 while he was
working at Netscape Communications Corporation. The language was
originally designed for client-side web development.`
  Output: 0
Text: `User1: Did anyone solve the memory leak issue? User2: Yes, it
was related to the event listeners User1: How did you fix it? User2:
We added a cleanup function in the useEffect hook User1: Got it, I`ll
try that`
  Output: 1
Text: `The Renaissance was a period in European history marking the
transition from the Middle Ages to modernity and covering the 15th
and 16th centuries.`
  Output: 0
Text: `- Can we push back the deadline? - I need to check with the
team first. When would you need it by? - Would next Friday work? -
Yes, that should be fine. I`ll update the project timeline.`
  Output: 1

DOCUMENT:
{document_text}

Return JSON only.
```

Output schema (JSON):

```
{`conversational`: <0 or 1>}
```

One template, eleven tasks. Each call shows the task, three formatted examples from that benchmark with ground truth, and the candidate document, then asks for four 1–5 ratings. The examples are fixed, pre-trimmed items (at most 300 words); they are the only part of the prompt that changes besides the task description, so we give the shared body and the eleven descriptions, and reproduce the MMLU example block in full as the pattern the other ten follow.

Benchmark usefulness

```
You are evaluating how useful a web document would be as pretraining
data for improving a language model on a specific downstream
evaluation task. You will be shown the task, three formatted examples
from that benchmark (with their ground-truth answers), and then a
candidate web document.

EVALUATION TASK:
{task_description}

EXAMPLES FROM THE BENCHMARK (with ground truth):
{benchmark_examples}

CANDIDATE DOCUMENT:
{document_text}

Rate the candidate document on the following 1-5 scales, judging only
how useful it would be for improving a model on the evaluation task
above:

1. usefulness
   1 = no value for this task; training on it would not help a model
with these examples
   3 = some indirect value; touches related skills or knowledge
```

```
   5 = highly valuable; directly builds the knowledge, reasoning, or
skills the task requires

2. content_relevance
   1 = subject matter is unrelated to the task`s domain
   3 = partially overlapping subject matter
   5 = closely matches the task`s subject matter and required
knowledge

3. style_match
   1 = tone/register is unlike anything useful for this task
   3 = somewhat compatible style
   5 = style and register closely match what helps with this task

4. format_match
   1 = structure/format bears no resemblance to the task`s format
   3 = some structural overlap
   5 = format closely mirrors the task`s examples (e.g.
multiple-choice, code, step-by-step solutions)

Also give a single short rationale.

Return JSON only, no other text.
```

Output schema (JSON), shared:

```
{`usefulness`: <int 1-5>, `content_relevance`: <int 1-5>, `style_match`: <int 1-5>,
 `format_match`: <int 1-5>, `rationale`: <str>}
```

The eleven `{task_description}` strings are:

- **MMLU.** MMLU (Massive Multitask Language Understanding): four-option multiple-choice questions spanning 57 subjects across STEM, the humanities, the social sciences, and professional domains. The model must select the single correct option (A–D). Scored by generation accuracy on the answer letter.
- **MMLU-Pro.** MMLU-Pro: a harder, reasoning-focused variant of MMLU with up to ten answer options (A–J) per question and more challenging, often quantitative items. The model selects the single correct option. Scored by generation accuracy.
- **PIQA.** PIQA (Physical Interaction QA): physical-commonsense reasoning. Given a goal, the model chooses which of two candidate solutions is more sensible. Ground truth is the index of the correct solution. Scored by likelihood/accuracy over the two solutions.
- **HellaSwag.** HellaSwag: commonsense sentence completion. Given a short context (often from activity or how-to text), the model chooses the most plausible of four endings. Ground truth is the correct ending index. Scored by likelihood/accuracy over the endings.
- **HumanEval (generation).** HumanEval (Generation): Python function synthesis. Given a function signature and docstring, the model generates a body that must pass hidden unit tests. Ground truth is a canonical solution. Scored by executing the generated completion against tests (pass@k).
- **HumanEval (bits-per-byte).** HumanEval (BPB / bits-per-byte): the same Python function-synthesis problems as HumanEval, but scored by the model's bits-per-byte likelihood of the canonical solution given the prompt, rather than by executing generated code. Lower bits-per-byte on the gold solution indicates better performance.
- **MBPP (generation).** MBPP (Mostly Basic Python Problems, Generation): given a short natural-language task description and a few assert-based test cases, the model generates a Python function that must pass the tests. Scored by executing the generated code.
- **MBPP (bits-per-byte).** MBPP (BPB / bits-per-byte): the same basic-Python problems as MBPP, but scored by the model's bits-per-byte likelihood of the reference solution given the task and tests, rather than by executing generated code.

- **MATH.** MATH (Minerva-style, Generation): competition mathematics problems. The model produces a step-by-step solution ending in a final answer, conventionally wrapped in `\boxed{...}` (Minerva formatting). Scored by checking the final boxed answer for mathematical equivalence.
- **GSM8K.** GSM8K (Generation): grade-school math word problems requiring multi-step arithmetic reasoning. The model produces a chain-of-thought solution ending in a final numeric answer (after the '`####`' delimiter in the reference). Scored by exact match on the final number.
- **TriviaQA.** TriviaQA (Generation): open-domain factual question answering. Given a trivia question, the model produces a short free-form answer. Ground truth is the accepted answer (and its aliases). Scored by normalised exact match.

HumanEval generation and HumanEval bits-per-byte share one example block, as do the two MBPP instantiations. The MMLU block, representative of the rest, is:

MMLU examples

```
Example 1
Question: Which of the following is the body cavity that contains the
pituitary gland?
A. Abdominal
B. Cranial
C. Pleural
D. Spinal
Answer: B

Example 2
Question: In 2018, about how many people in the United States were
homeless?
A. 55,000
B. 550,000
C. 5,500,000
D. 55,000,000
Answer: B

Example 3
Question: The concept of `double effect` in ethics is most closely
associated with which of the following?
A. Utilitarianism
B. Kantian ethics
C. Natural law theory
D. Virtue ethics
Answer: C
```

D COMMON CRAWL PREPROCESSING

The one-million-document sample judged in Section 2 is drawn from an English Common Crawl corpus preprocessed with the OLMo 3 / Dolma 3 recipe (OLMo Team, 2025; Soldaini et al., 2024), itself in the lineage of RefinedWeb, DCLM, and FineWeb (Penedo et al., 2023; Li et al., 2024; Penedo et al., 2024). We give only a high-level overview here; the sample this paper studies is taken after deduplication, and the quality partitioning that would normally follow is exactly the step this paper's judge-and-blend method replaces.

Text extraction (linearization). Raw Common Crawl WARC records are linearized into plain text, stripping HTML markup and page metadata that are not useful for training.

Heuristic filtering. A cascade of cheap, coarse filters removes obviously low-quality or non-English text, ordered cheapest-first so the expensive steps see less data: URL blocklists (spam/adult domains); basic document heuristics (too short/long, insufficient alphanumeric content, excessive stopwords); Gopher-style repetition filters; an English language-identification filter (fastText (Joulin et al., 2017), keeping documents with English score ≥ 0.65); and a MadLad-style filter that drops documents with too many 'questionable' sentences (anomalous casing, cursed regexes). A final pass applies non-filtering text modifiers that strip repeated newlines and common boilerplate phrases without removing documents.

Deduplication (three rounds, increasing coarseness). First, exact deduplication over byte-level content hashes removes identical documents. Second, MinHash-based fuzzy deduplication (5-grams, targeting Jaccard similarity ≈ 0.8) removes near-duplicates that differ only in, e.g., headers or footers, keeping the most recent document per cluster. Third, suffix-array substring deduplication removes repeated substrings of at least 500 bytes, the boilerplate headers, footers, and navigation text that recur across many pages. The corpus this paper samples is the output of this stage, hence 'substring-deduplicated Common Crawl' in Section 2.1.

Downstream partitioning (studied here, not inherited). OLMo 3 then partitions the deduplicated corpus into `(topic, quality)` buckets, with a topic classifier and a single Dolma 3 quality classifier, and mixes and upsamples across those buckets. That final quality-scoring-and-mixing step is precisely the subject of this paper: in place of one quality classifier we judge, distil, and blend a plurality of 'quality' axes (Sections 2–4).

Table 5: Distillation training configuration for the shared educational-anchor construct (binary top-25% vs. bottom-25% of the Qwen3-32B educational score), one column per backend. Numeric values are read verbatim from the training run configs; – marks a setting that does not apply to that backend.

Setting	fastText	Luxical + MLP	BERT (fine-tuned)
Input representation	char n-grams 3–6, word n-grams ≤ 3	frozen Luxical-one (192-d) embedding	`bert-base-uncased`
Trainable part	full linear model	MLP head only	all weights
Objective	softmax (2-class)	BCE-with-logits	cross-entropy (2-class)
Learning rate	0.05	3×10^{-4}	2×10^{-5}
Schedule	1 epoch	500 steps	1 epoch, warmup 0.1
Batch size	–	2048 (embed)	32 train / 64 eval
Embedding / hidden dim	100	256	–
Dropout	–	0.1	–
Max sequence length	–	–	256
Weight decay	–	–	0.01
Max grad norm	–	–	1
Min token count	5	–	–
Decision threshold	argmax	0.5	0.5

Table 6: Final held-out validation metrics for the educational-anchor construct ($n_{\text{train}} = 450,012$, $n_{\text{val}} = 49,984$, $1/9$ split, one fixed seed). Positive-class F1 and precision/recall at threshold 0.5. Best per column in bold, second best underlined. fastText's pipeline logs neither AUC nor a validation loss (–).

Backend	Precision	Recall	F1	AUC	Val. loss
fastText	0.865	0.865	0.865	–	–
Luxical + MLP	<u>0.920</u>	<u>0.907</u>	<u>0.913</u>	<u>0.971</u>	<u>0.219</u>
BERT (fine-tuned)	**0.946**	**0.943**	**0.945**	**0.986**	**0.148**

E DISTILLATION TRAINING DETAILS

This appendix records how the three distillation backends are trained, on a single shared construct so the set-ups are directly comparable: the educational anchor (top-25% vs. bottom-25% of the Qwen3-32B educational score), the same 450,012-row training / 49,984-row validation split (one fixed seed) used everywhere in Section 2. The three backends are a fastText linear model over character and word n-grams, a frozen Luxical-one embedding with a trained MLP head, and a fully fine-tuned `bert-base-uncased`.

The training runs logged only endpoint metrics, not per-step learning curves, so we report the exact configuration and the final held-out validation metrics rather than loss curves. Table 5 gives the full per-backend configuration (numeric values read verbatim from the run configs), and Table 6 the final validation metrics. The ordering is the expected one, BERT $>$ Luxical $>$ fastText on every metric, but even the cheapest backend recovers the educational anchor with 0.865 F1, which is why the corpus-scale scoring in Section 3 can use the light backends where the fine-tuned model is too expensive.

F MARGINALS, DIMENSIONALITY, CLASSIFIER METRICS, AND BACKEND AGREEMENT

This appendix is the self-contained account of the judged marginals, the dimensionality of the score space, construct validity, how the distilled backends agree, and the register and surface-form audits. Sections 2.2 and 2.3 state the headline: the axes correlate to differing degrees, both as judgements and as distilled classifiers.

What raw crawl looks like under each construct. Among the twelve general-rubric axes, surface-form axes score high (`orthography` mean 4.35, `linguistic_quality` 3.92) while substance and interactivity axes score low (`educational_value` 2.12, `instruction_likeness` 1.50, `dialogue_likeness` 1.48): the language is generally clean but of low educational or dialogue-like content (Penedo et al., 2024) (Fig. 11). Estimated provenance is dominated by news (378k) and none-of-the-above (369k), with traditionally high-quality classes rare (Wikipedia 39k, arXiv 20k, chat-like data $\sim$1.2k). The clearest marginal finding is that for the benchmark-usefulness prompts, `format_match` is an effective constant 1.0 ($\geq$998k/1M): almost no raw web document resembles a multiple-choice item, a unit-tested function, or a boxed proof, and code and maths tasks sit at the floor of `usefulness` (Fig. 12, Table 7). This is a strong argument for synthetic data (Maini et al., 2024; DatologyAI Team, 2025a) and is why these axes carry no learnable signal (Section 2.3). Both the provenance and sociolinguistic prompts exhibited schema drift, which we report in the score histograms (Fig. 11).

Table 7: Per-benchmark `usefulness` and `format_match` of the benchmark-usefulness prompts, as mean $\pm$ sample standard deviation over the 1M sample ($n = 989,992 - -999,992$). `format_match` is a near-constant floor at 1.0 (tiny dispersion); `usefulness` is a low-valued spike whose modest spread confirms almost no web document resembles a benchmark item (Section 2.2).

Benchmark	`usefulness`	`format_match`
HellaSwag	1.727 $\pm$0.471	1.018 $\pm$0.135
TriviaQA	1.271 $\pm$0.446	1.010 $\pm$0.102
MMLU	1.269 $\pm$0.446	1.002 $\pm$0.053
PIQA	1.122 $\pm$0.329	1.002 $\pm$0.040
MMLU-Pro	1.057 $\pm$0.234	1.001 $\pm$0.043
GSM8K	1.042 $\pm$0.201	1.000 $\pm$0.023
MBPP (BPB)	1.024 $\pm$0.152	1.002 $\pm$0.043
MBPP (gen)	1.019 $\pm$0.139	1.002 $\pm$0.039
HumanEval (gen)	1.019 $\pm$0.135	1.001 $\pm$0.038
HumanEval (BPB)	1.015 $\pm$0.123	1.001 $\pm$0.033
MATH (Minerva)	1.010 $\pm$0.105	1.001 $\pm$0.026

Table 8: Mean $\pm$ sample standard deviation of the three anchor scores conditioned on predicted provenance, with the per-class document count n. Academic and encyclopedic documents score high on the educational and Wikipedia anchors but low on the chat-like anchor; the ordering inverts for chat-like provenance (Section 2.2).

Predicted provenance	Educational	Wikipedia	Chat-like	n
arXiv/peS2o	3.688 $\pm$0.870	3.807 $\pm$0.742	1.455 $\pm$0.550	20,177
Wikipedia	2.863 $\pm$1.060	3.080 $\pm$0.858	1.938 $\pm$0.606	38,880
chat-like data	2.536 $\pm$0.880	2.069 $\pm$0.538	2.500 $\pm$0.693	23,534
none of the others	1.515 $\pm$0.651	1.573 $\pm$0.552	1.559 $\pm$0.592	369,222

Anchors encode independent targets. Conditioning the anchor scores on predicted provenance (Table 8) makes the plurality concrete. arXiv/peS2o documents score 3.67/3.78 on the educational and Wikipedia anchors but only 1.45 on the chat-like anchor; the ordering inverts for chat-like text. 'Educational', 'Wikipedia', and 'chat-like' anchors define different objectives.

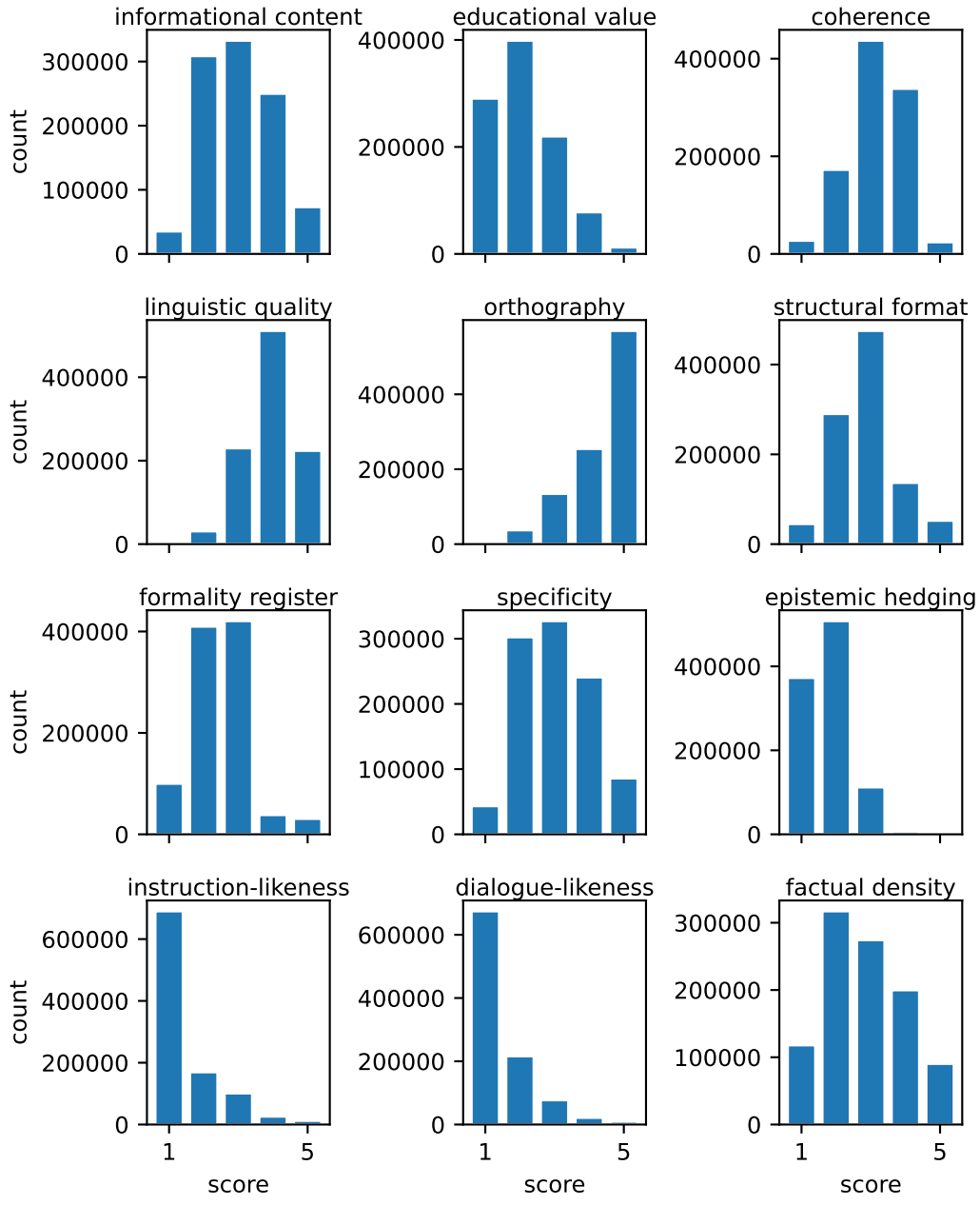

Figure 11: Marginal 1–5 histograms for the twelve general-rubric axes over the 1M sample. Surface-form axes score high (orthography 4.35, linguistic quality 3.92) while substance and interactivity axes score low (educational value 2.12, instruction-likeness 1.50, dialogue-likeness 1.48). Provenance and sociolinguistic prompts also show schema drift here, reported as emitted.

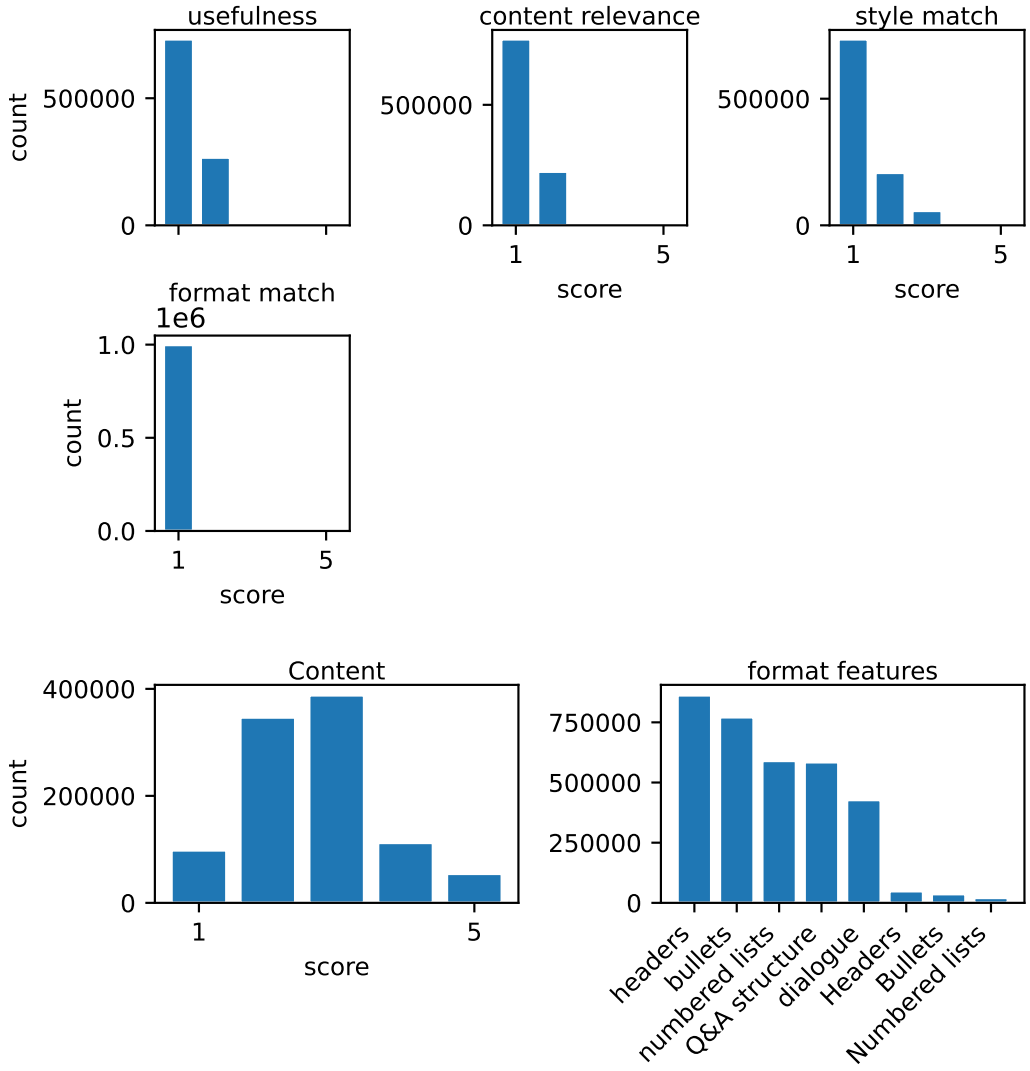

Figure 12: **Top:** MMLU-usefulness per-field histograms; format match is effectively constant at 1 for ∼all 1M documents. **Bottom:** Content distribution, one of the most-centred single-document distributions by mean, alongside the top observed format features.

The latent structure is low-dimensional but plural. A PCA over seventeen content/anchor/probe axes (Fig. 13) finds a large first component. PC1 explains 48.1% of the variance and loads almost uniformly on the content axes (`informational_content` +0.91, the content-focussed score +0.88, `specificity` +0.88, `coherence` +0.87, `educational_value` +0.84); PC2 (10.6%) is an anchor axis; PC3 (8.0%) loads on interactivity/structure. Six components are needed for $> 80\%$ of variance and nine for $> 90\%$. The latent structure is a dominant content factor, a register/structure block (`structural_format`, the format-focussed score), an interactivity block in which `conversational` is essentially uncorrelated with every other axis, an anti-correlated non-native-English block, and an evaluation-relevance block that further splits into maths/code and knowledge. This corroborates the QuRating observation (Wettig et al., 2024) that nominally distinct criteria covary, and it isolates axes that do not. Fig. 14 shows that split document by document: a Content score fans out across the other framings.

Construct validity is limited even within one judge. Constructs probed under different framings rank documents quite differently (Table 9). The two informational/content prompts agree strongly

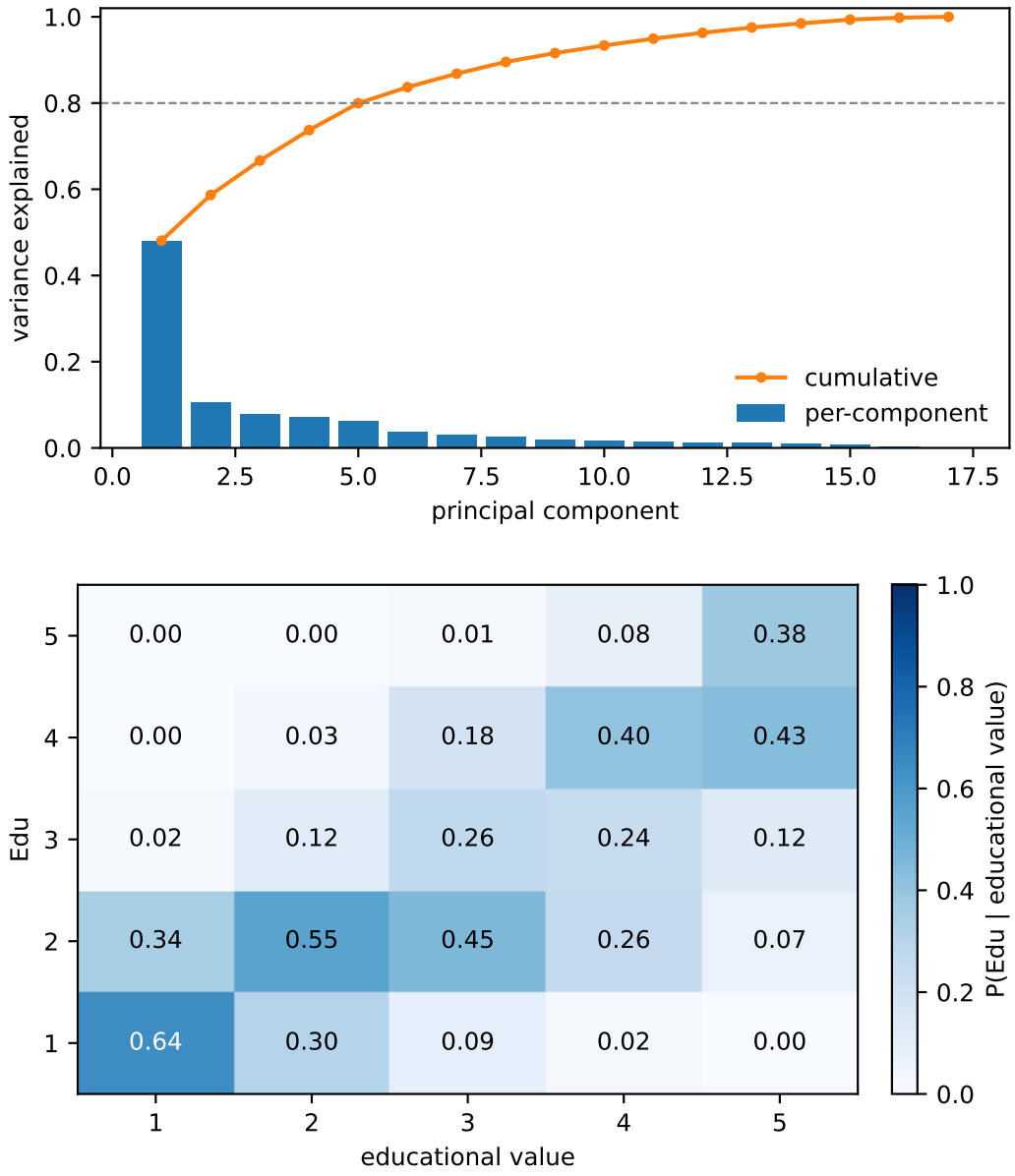

Figure 13: **Top:** PCA scree of the 17-axis judge quality space (eigendecomposition of the Spearman correlation matrix): PC1 $= 48.1\%$, six components reach $> 80\%$, nine reach $> 90\%$. **Bottom:** column-normalised joint distribution of educational value and Edu; agreement is positive but diffuse ($\rho = 0.59$).

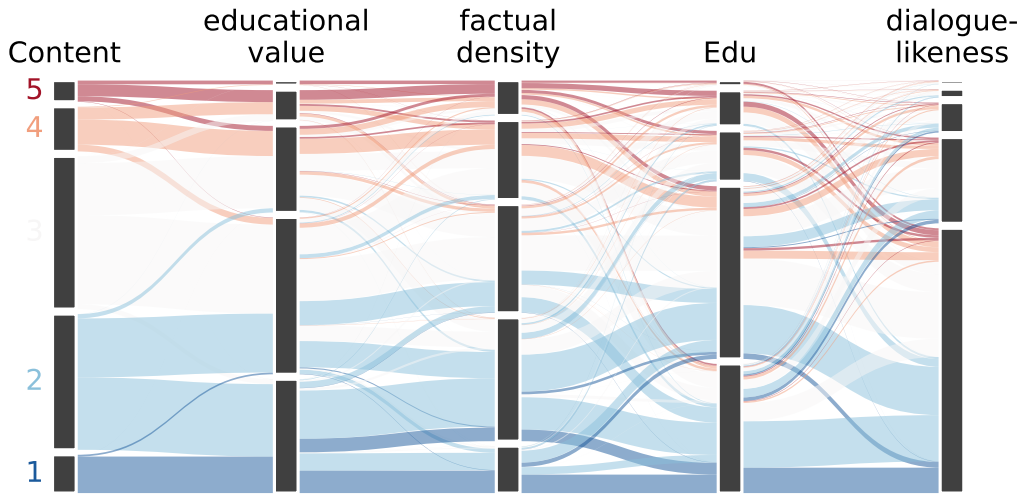

Figure 14: **Score migration.** Each document's score under five framings in turn, with ribbons coloured by its Content score and blocks sized by population. A document's Content score fans out across every score of the other framings: of documents with a Content score of 5, 24% land at an educational-value score of 3, 57% at a score of 4, and 18% at a score of 5. Flat ribbons would mean the framings agree; they do not.

($\rho = 0.821$), but one model re-rating the *same* construct under two framings agrees only moderately: educational value $\rho = 0.593$ (Fig. 13, bottom), structural quality $\rho = 0.589$. The content-focussed and format-focussed scores correlate at $\rho = 0.623$ despite explicit instructions to each prompt to ignore the other's features (Fig. 15), so a 'content' score and a 'format' score are not independent selection signals.

Table 9: Construct (in)validity: Spearman ρ between prompt pairs intended to measure the same or closely related constructs. Even one judge re-rating the same construct under two framings agrees only moderately (educational value 0.593; structural quality 0.589).

Construct	Prompt pair		Spearman ρ
Informational substance	`informational_content`	vs content-focussed score	0.821
Factual substance	`factual_density`	vs content-focussed score	0.743
Educational value	`educational_value`	vs educational anchor	0.593
Structural quality	`structural_format`	vs format-focussed score	0.589
Dialogue/conversation	`dialogue_likeness`	vs conversational screen	0.431
Informativeness	`informational_content`	vs Ask-LLM	0.323

Why the bottom decile looks uncontested. The decile curve in Fig. 2 (right) is asymmetric: disagreement is lowest for the bottom decile (0.39) and highest for the top decile (0.77), with the middle intermediate ($\sim$0.49 at deciles 5–6) and a marked jump between deciles 6 and 7. Documents which fail a diverse set of positive constructs do so jointly, whereas documents that satisfy some positive constructs tend to satisfy *different* ones. Those constructs are properties of one kind of success, as sweetness and redness are properties of a tasty apple: documents that are neither are rejected together. A green, savoury document can still be good under a criterion these prompts never pose, and it would fall in the same bottom bin, with the same apparent consensus. Whether low quality is intrinsically less contested or under-constructed in our design (we included few explicit negative-quality prompts) is something the current setup cannot cleanly distinguish; we flag it as a limitation (Section 6).

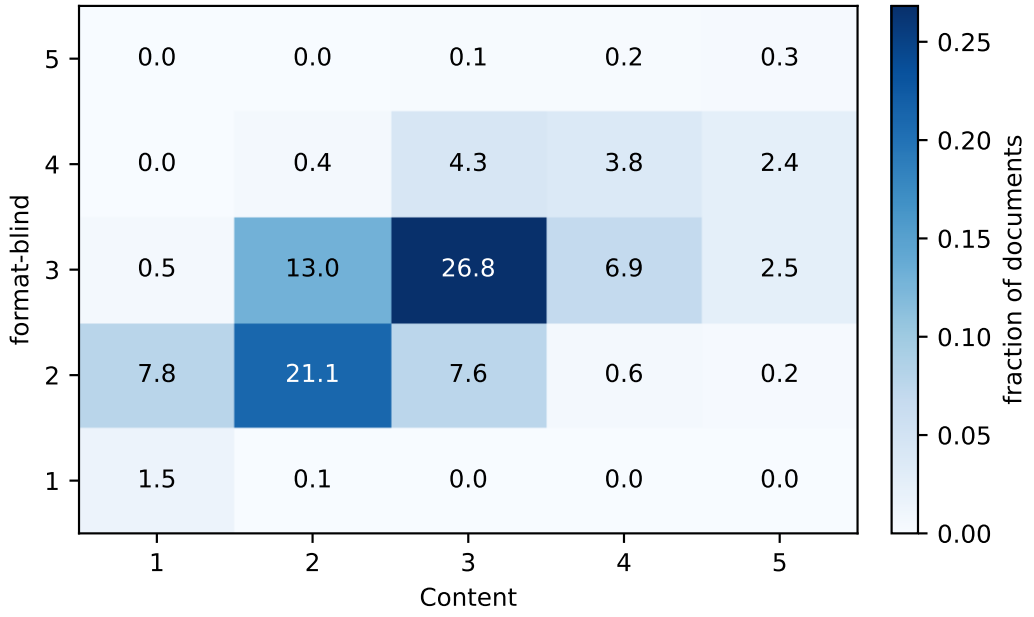

Figure 15: Joint document density of Content vs format-blind; mass concentrates near the diagonal (21.1% at $(2, 2)$, 26.8% at $(3, 3)$), consistent with $\rho = 0.623$ despite the explicit blinding instructions in each prompt.

Metric agreement is not selection agreement. Backends trained on the same axis agree strongly in aggregate rank on the learnable tasks, yet retain materially different top-25% sets: Jaccard 0.47–0.61 (Fig. 16). At the extreme, the top-10% sets of the educational and chat-like anchor classifiers overlap at Jaccard 0.09–0.15, so they retain almost-disjoint samples. Even a rank correlation of $\rho = 0.75$ is compatible with disjoint top-25% sets. Swapping fastText for Luxical or BERT on the same construct, at a fixed top-k threshold, still changes the corpus that reaches the model when their reported metrics match. The choice of backend is a data-selection decision.

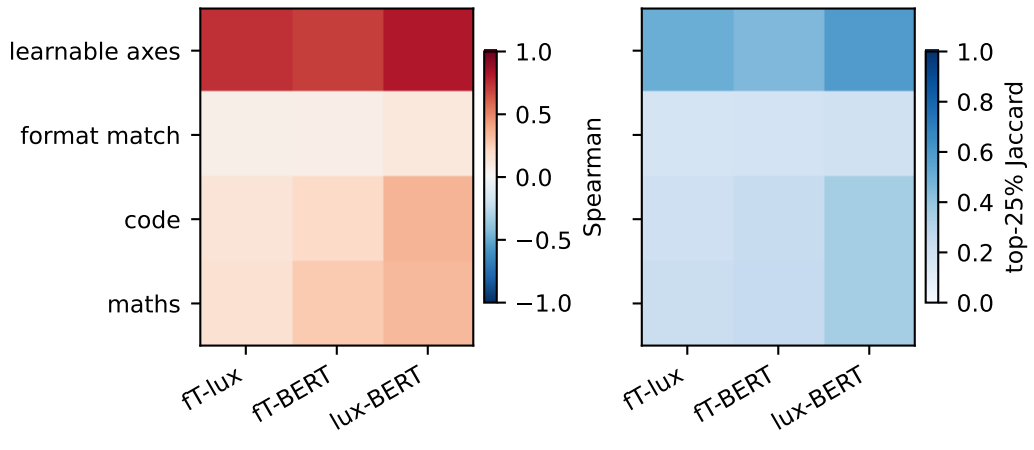

Figure 16: Cross-backend agreement on a new random sample of Common Crawl over all 59 binary distillation tasks, averaged within task families: mean Spearman rank agreement (left, red = agree) and mean top-25% retained-set Jaccard (right, darker = more overlap) for the three backend pairs. The learnable axes agree most; the format match, code, and maths families sit near zero. Even where rank agreement is high, top-25% Jaccard is only 0.47–0.61.

Two disagreement structures. Where judge disagreement across constructs rises towards the top decile (Section 2.2), classifier disagreement across backends on a single axis is U-shaped: lowest at both extremes and highest at the decision boundary (Fig. 17). Across constructs the contested region is which documents count as high quality; across backends distilling one construct the contested region is the boundary on that construct. Held-out BERT scores reproduce the judge's block structure (Fig. 18).

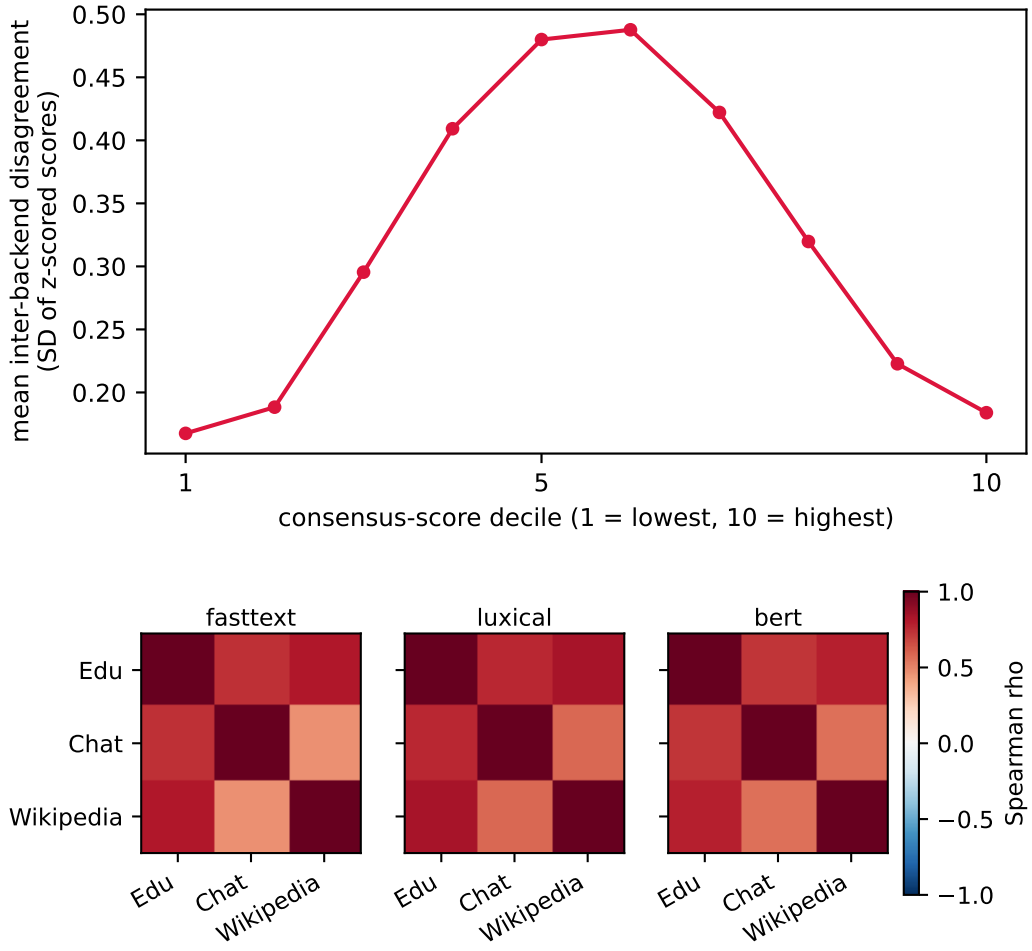

Figure 17: **Top:** mean inter-backend disagreement by consensus-score decile, averaged over the nine learnable axes: roughly normal (lowest at both extremes, peaking mid-range), the complement of the judge-side asymmetry in Fig. 2. **Bottom:** rank agreement among the three anchor classifiers (Edu, Chat, Wikipedia) per backend; Edu↔Wikipedia highest, Chat↔Wikipedia lowest, reproducing the judge-level anchor divergence.

Categorical tasks are fragile across backends. The categorical judgements are learnable in aggregate (provenance macro-F1 up to 0.701, English variety up to 0.749 under Luxical/BERT; both collapse under fastText, macro-F1 0.235 and 0.155). Agreement between backends is weak: the fastText and BERT models trained on estimated provenance have Cohen's $\kappa = 0.12$, and on English variety $\kappa = 0.013$. On English variety, fastText collapses to a single dominant class (87% `UK`) while BERT spreads across `US` (46%) and `non_native` (23%), closer to the judge's own distribution. Multi-class distillation of these judgements is fragile at this model scale. The same ranking, BERT and Luxical above fastText, is the one Table 2 shows for the binary tasks.

Register, not English variety. Revisiting Gururangan et al. (2022) at the document level, anchored quality barely differs by judged English variety (US vs non-native Cohen's $d = -0.031$ for the

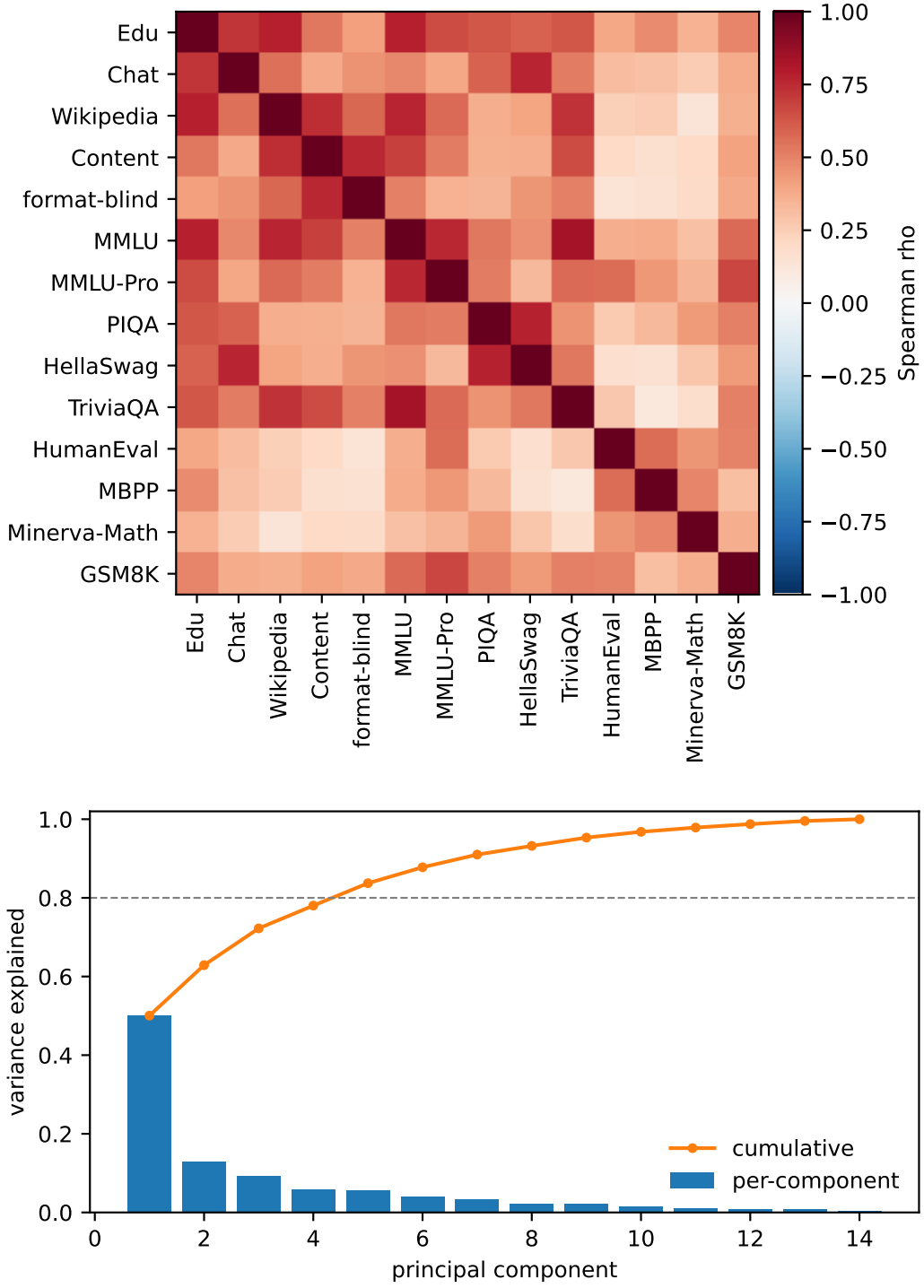

Figure 18: Held-out BERT classifier scores reproduce the judge's low-but-plural structure: a content/anchor block plus a weakly-coupled code/maths block (top), and a PCA scree with PC1 $= 50.1\%$ and five components for $> 80\%$ (bottom), closely matching the judge's 48.1%/six.

Wikipedia anchor, $+0.007$ for the educational anchor, $+0.111$ for educational value); sociolinguistic register does the work, with prestige/formality registers correlating 0.30–0.57 with most quality axes and non-native-English features uniformly slightly negative (Fig. 19). The filter's bias is a register preference, which suggests register-aware auditing before large training runs.

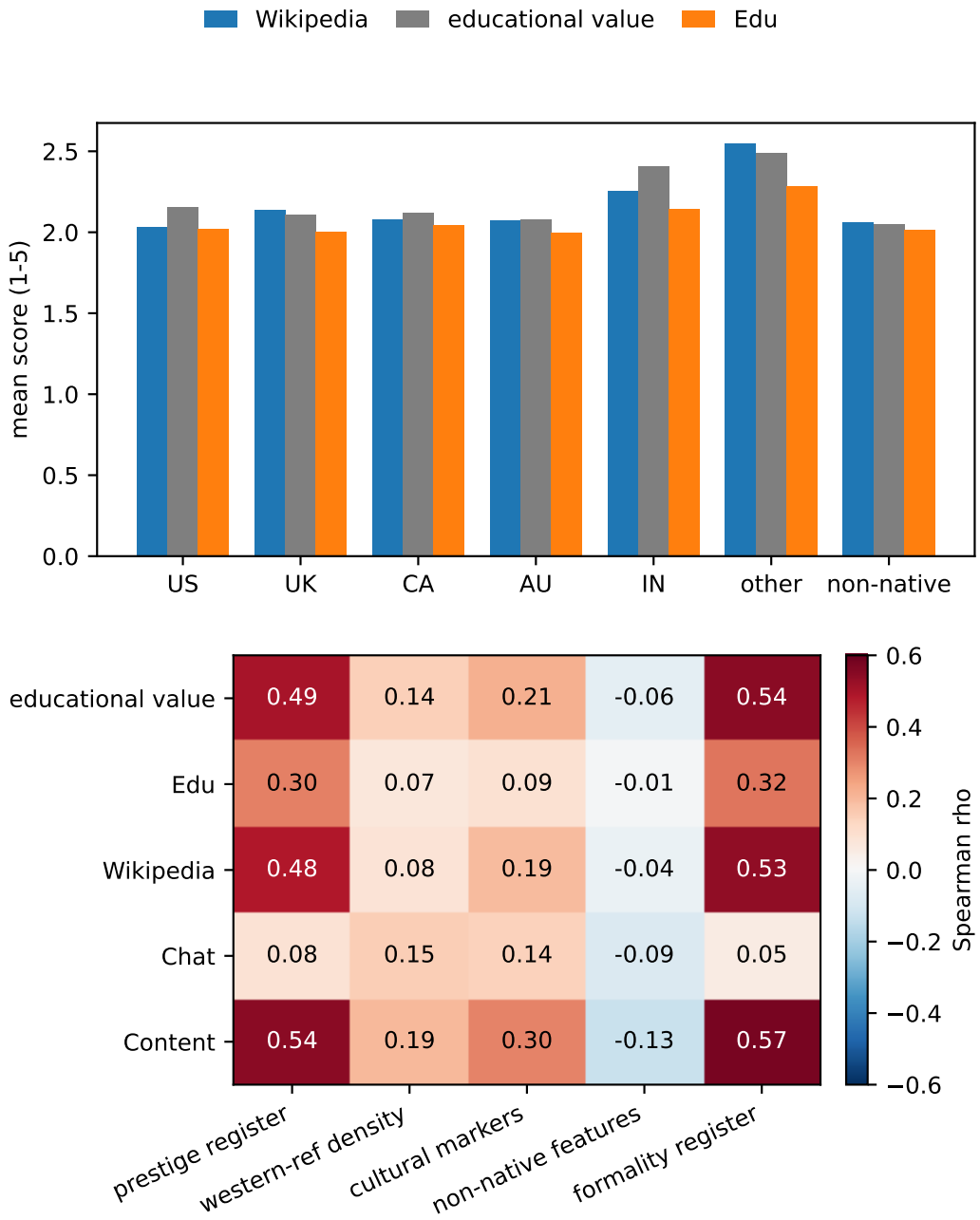

Figure 19: **Top:** mean anchored quality by judged English variety; differences are small (all means $\approx$ 2.0–2.5), with US vs non-native Cohen's $d = -0.031$ (Wikipedia), $+0.007$ (Edu), $+0.111$ (educational value). **Bottom:** Spearman of five quality measures against five sociolinguistic features; prestige/formality correlate 0.30–0.57, non-native features are uniformly slightly negative.

A third of the signal is surface form. Ordinary least squares from eleven heuristic text features recovers a substantial minority of judge-score variance ($R^2 = 0.42$ for `structural_format`,

0.34 for `educational_value`, 0.31 for `informational_content`, 0.19 for the educational anchor; Fig. 20). Roughly a third of judge-score variance is predictable from trivial features. This sets a floor the distilled classifiers clear (BERT reaches AUC 0.987 on the educational anchor against that OLS R^2 of 0.19; Table 2), and part of the signal may be an artefact of the upstream heuristic filtering pipeline that produced the corpus.

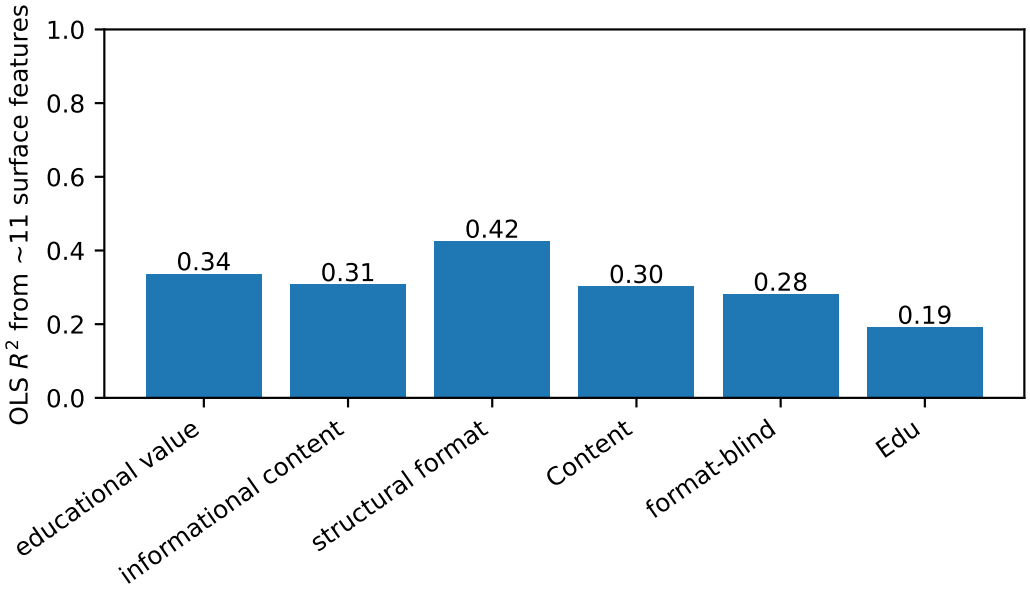

Figure 20: Variance in judge scores recovered by OLS on eleven heuristic surface features: $R^2 = 0.42$ (structural format), 0.34 (educational value), 0.31 (informational content), 0.30 (Content), 0.28 (format-blind), 0.19 (Edu); the strongest single predictors are document length and mean word length.

G MECHANISTIC INTERPRETABILITY OF THE DISTILLED CLASSIFIERS

This appendix stands on its own. The main text only records the conclusion (Section 2.3): the split between the educational and chat-like anchors is visible inside the trained models, and it is causal. We apply input-dimension ablations, activation patching, and sparse autoencoders (SAEs). fastText is fully linear, so per-token attributions are exact; the Luxical backbone is frozen, so we reconstruct the 192-d embeddings and causally ablate dimensions; BERT is fine-tuned, so we use patching, probing, and a logit lens.

fastText per-word contributions. On documents where the anchors most disagree, the exact per-word logit contributions separate cleanly by anchor: education-vocabulary tokens drive the educational anchor positive while remaining near-neutral for the Wikipedia and chat-like anchors (a store-listing app page, educational $+1.09$, Wikipedia -1.50, chat-like -1.50); Wikipedia-chrome tokens drive the Wikipedia anchor (a near-empty stub, Wikipedia $+1.24$); everyday conversational tokens drive the chat-like anchor (a casual blog thread, chat-like $+1.02$) (Fig. 21). The same split appears token by token on held-out documents: many tokens move the three anchor heads in different directions (Fig. 22).

Luxical geometry and SAE. Decomposing the 59×192 matrix of head Jacobians recovers the same low-dimensional structure (Fig. 23). An overcomplete SAE (512 features, mean $L_0 \sim 61$) trained on the frozen 192-d space yields readable, causally-verified features: an econometrics feature (`econometric, repec, nber`) loads $+8.8$ on the educational anchor and $+7.5$ on the Wikipedia anchor but -3.0 on the chat-like anchor; a casual-blog feature (`grandma, knitting, blogging`) loads $+6.9$ on the chat-like anchor but -1.1 on the Wikipedia anchor (Fig. 24). The chat-like classifier's preference for conversational registers and aversion to academic vocabulary is therefore interpretable and causal.

BERT: where the signal forms and what it encodes. Linear probes on the pooled `[CLS]` representation recover both the target and a battery of register/format features at high R^2 (the educational-anchor model recovers its own target at 0.70, `educational_value` 0.62, `formality_register` 0.61, stopword ratio 0.58; Fig. 25). Activation patching shows the quality-relevant `[CLS]` information forms late: negligible recovery through layer 8, rising to 0.34–0.39 at layer 11 (Fig. 26). An SAE over the final `[CLS]` (2,048 features) finds the score spread across features: causal ablation of each top feature moves the score in the sign of its correlation but only slightly (Fig. 27).

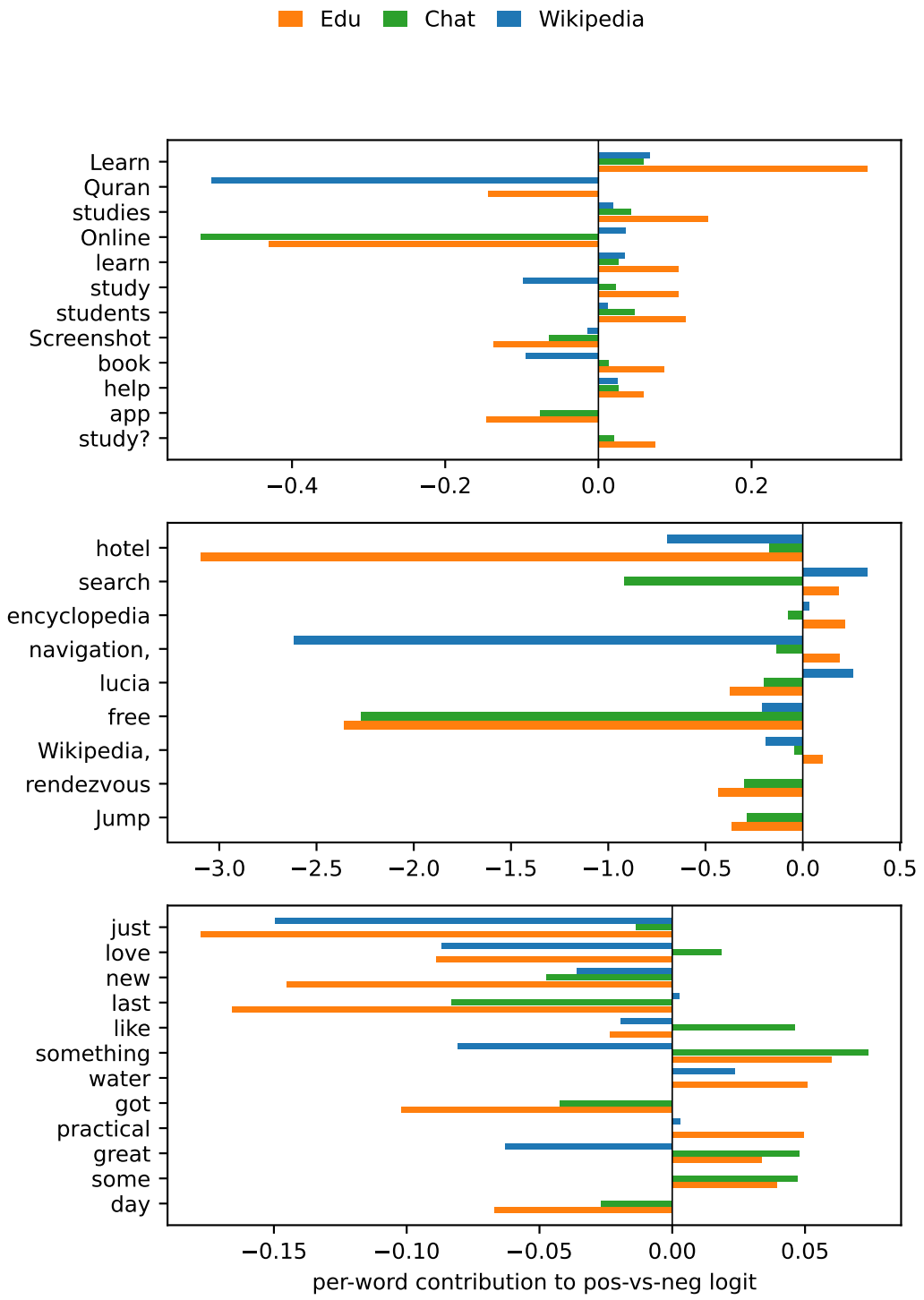

Figure 21: Exact per-word logit contributions under the three fastText anchor classifiers (Edu orange, Chat green, Wikipedia blue), on the content words where the anchors most disagree. **Top:** an Edu-high store listing. **Middle:** a Wikipedia-high stub. **Bottom:** a Chat-high casual blog thread.

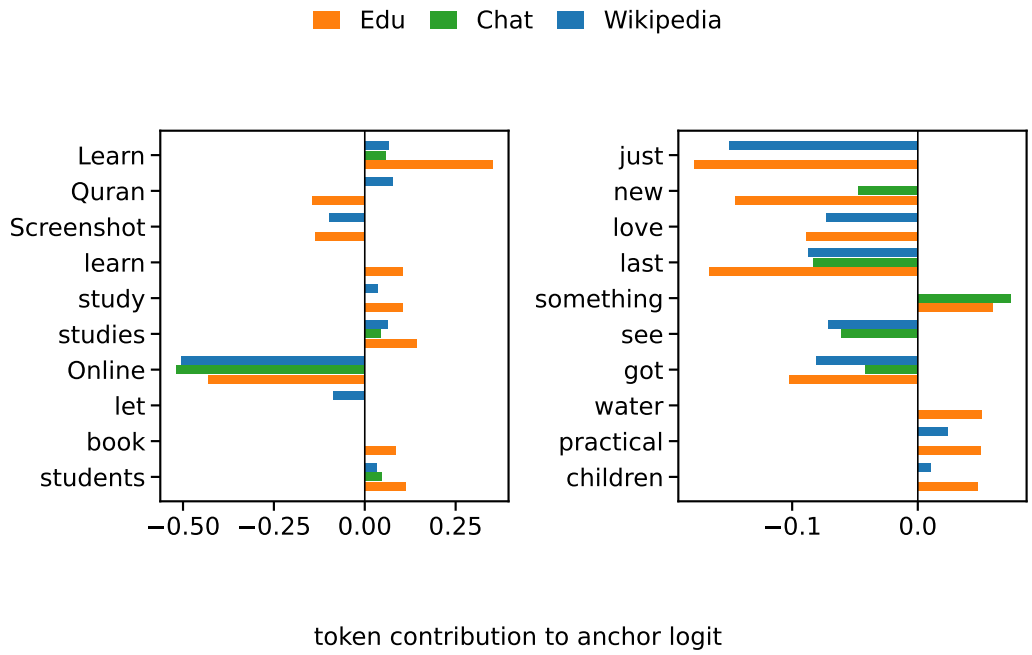

Figure 22: **The same words, different quality.** fastText per-token contributions to the logit of each of the three anchor heads (educational, chat-like, Wikipedia) for two flagship held-out documents, ordered by cross-head spread. Some tokens ("Online", "app") depress all three heads, but many ("Learn", "practical", "great") move the heads in different directions: the anchors read the same tokens differently.

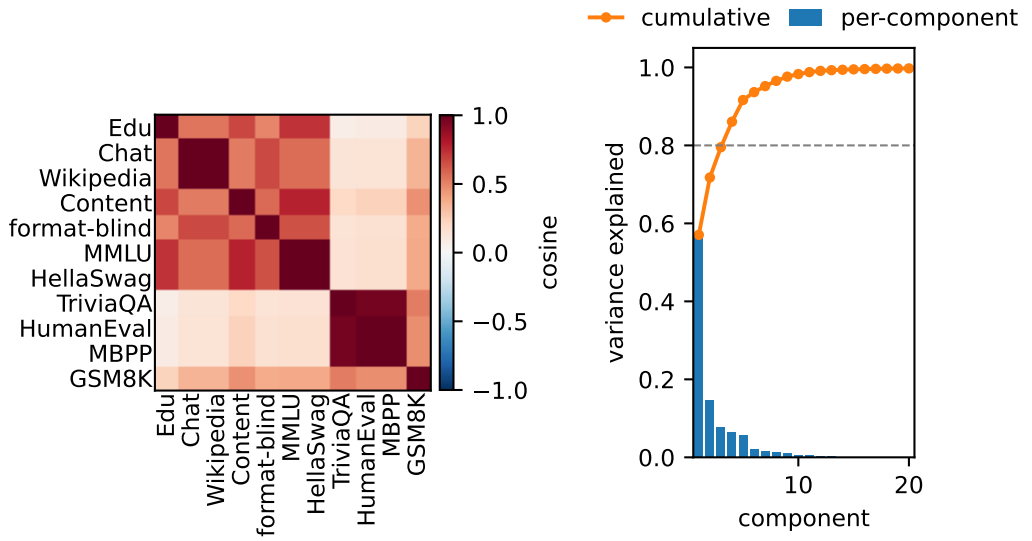

Figure 23: Cross-task cosine of the Luxical heads' effective input directions (Jacobian at the origin) in the shared frozen 192-d space (left): a knowledge/anchor block separate from a tight code block (HumanEval↔MBPP 0.96). PCA of all 59 head directions (right): PC1 = 57%, four components for >80%. Anchor cosines: educational vs Wikipedia 0.74, educational vs chat-like 0.67, chat-like vs Wikipedia 0.58 (most distinct).

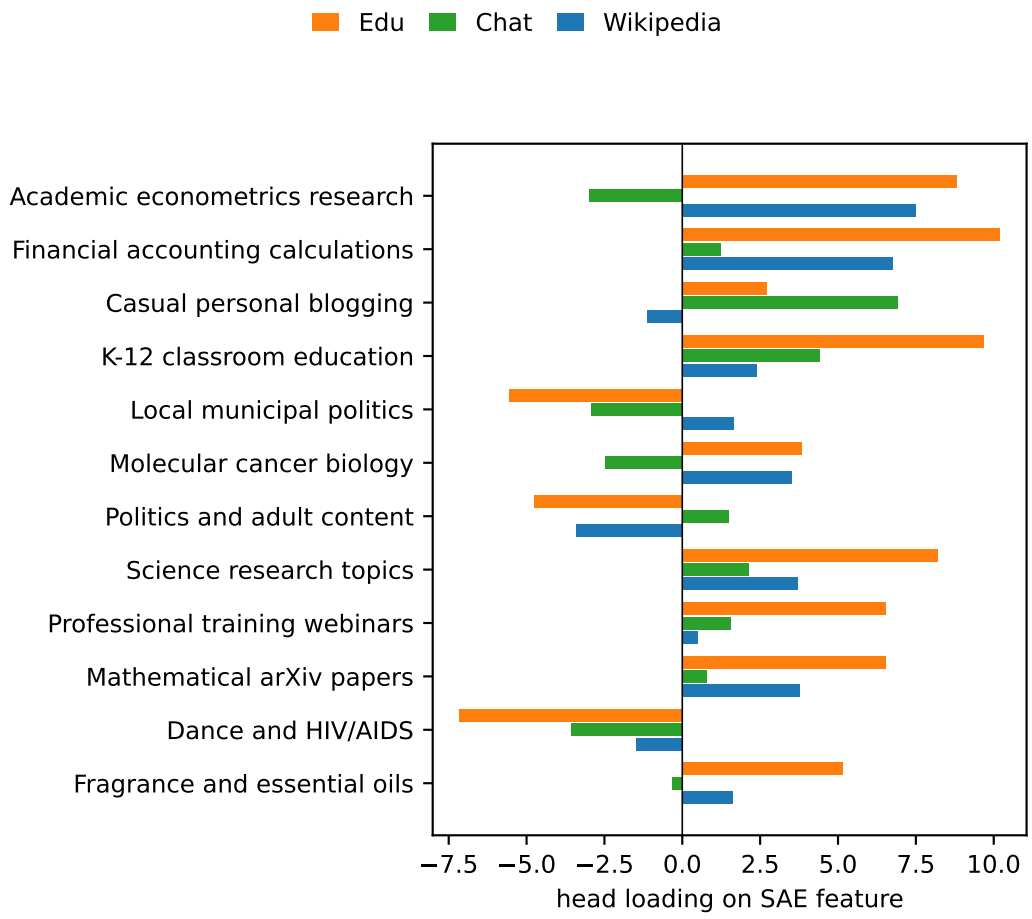

Figure 24: SAE features whose loadings onto the three Luxical anchor heads differ most, labelled by an LLM from each feature's top characteristic words. Academic/econometrics/arXiv features load strongly positive for Edu and Wikipedia but negative for Chat; casual personal blogging is the reverse.

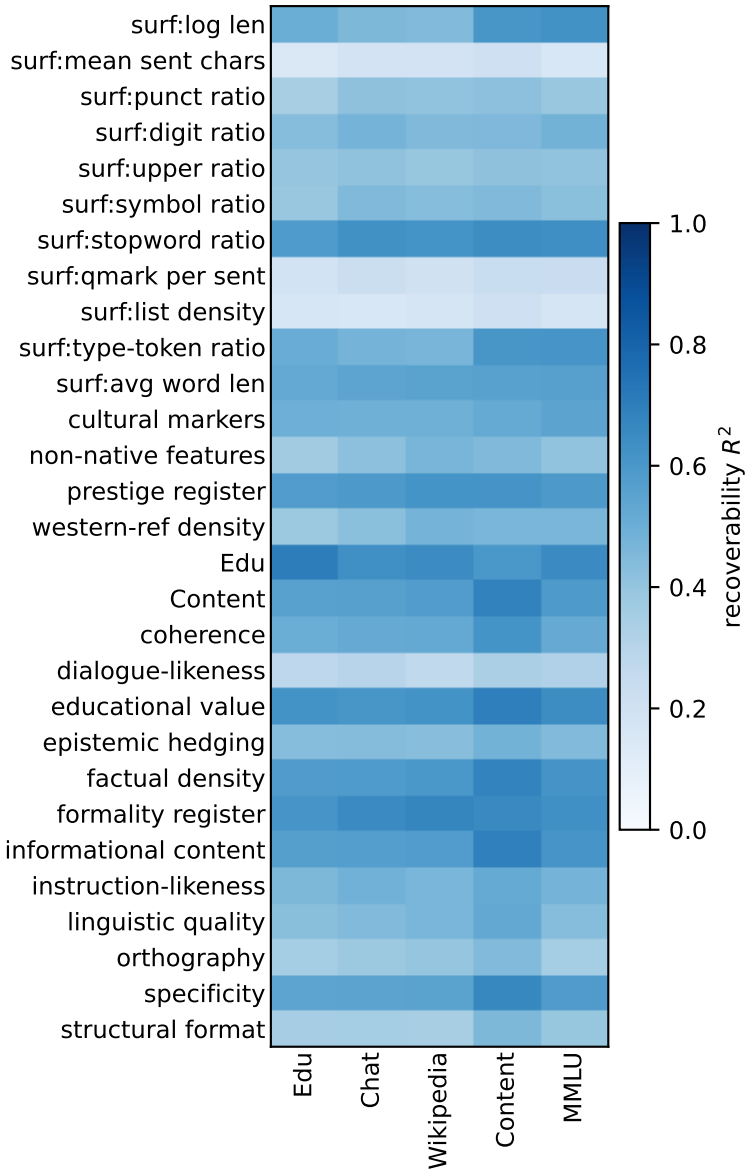

Figure 25: Linear-probe recoverability (R^2) of surface and judge features from the pooled `[CLS]` representation.

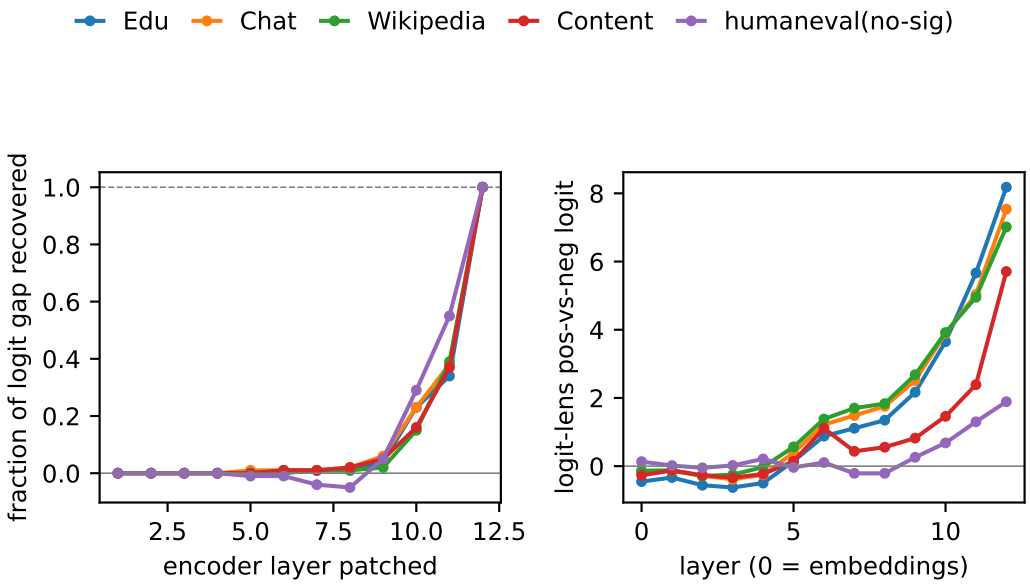

Figure 26: Left: fraction of the high-vs-low logit gap recovered when the clean `[CLS]` state is patched into the corrupt run at each encoder layer. Right: logit-lens decode of each layer's `[CLS]`. The quality signal localises to the final layers.

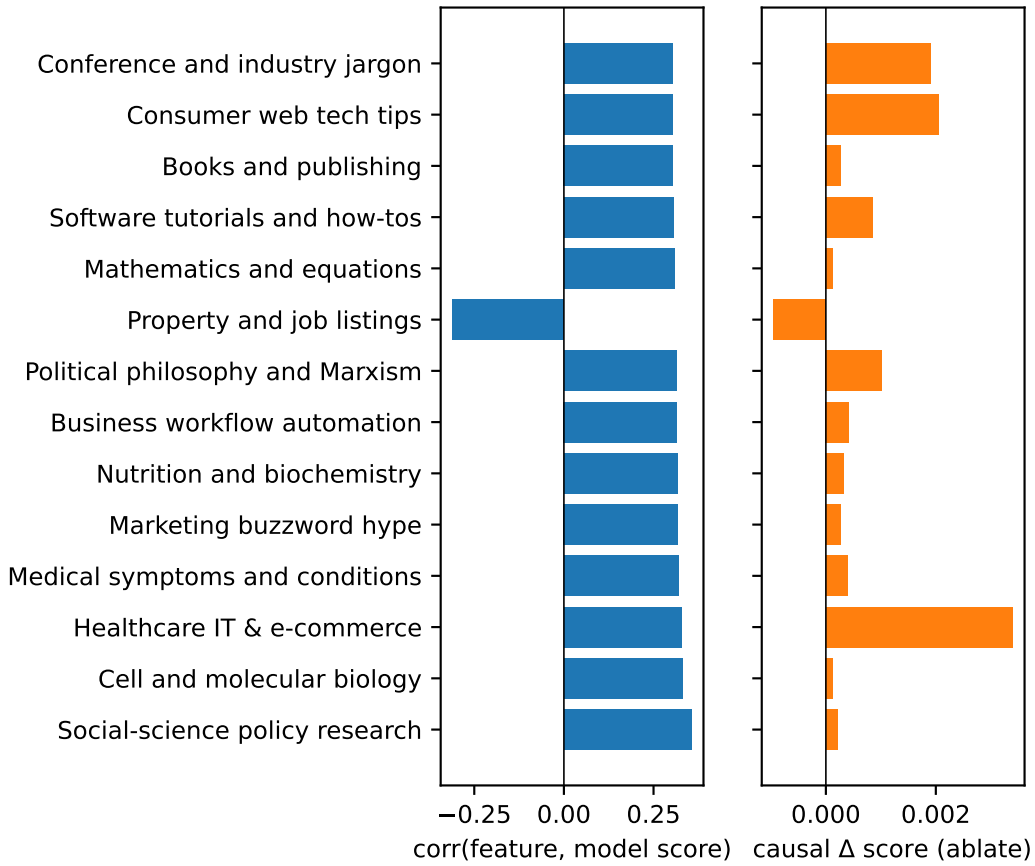

Figure 27: For the top SAE features of the educational-anchor BERT model, the correlation with the model score alongside the causal Δ score when the feature is ablated ($\sim 100\times$ smaller scale). Signs agree; magnitudes are uniformly tiny.

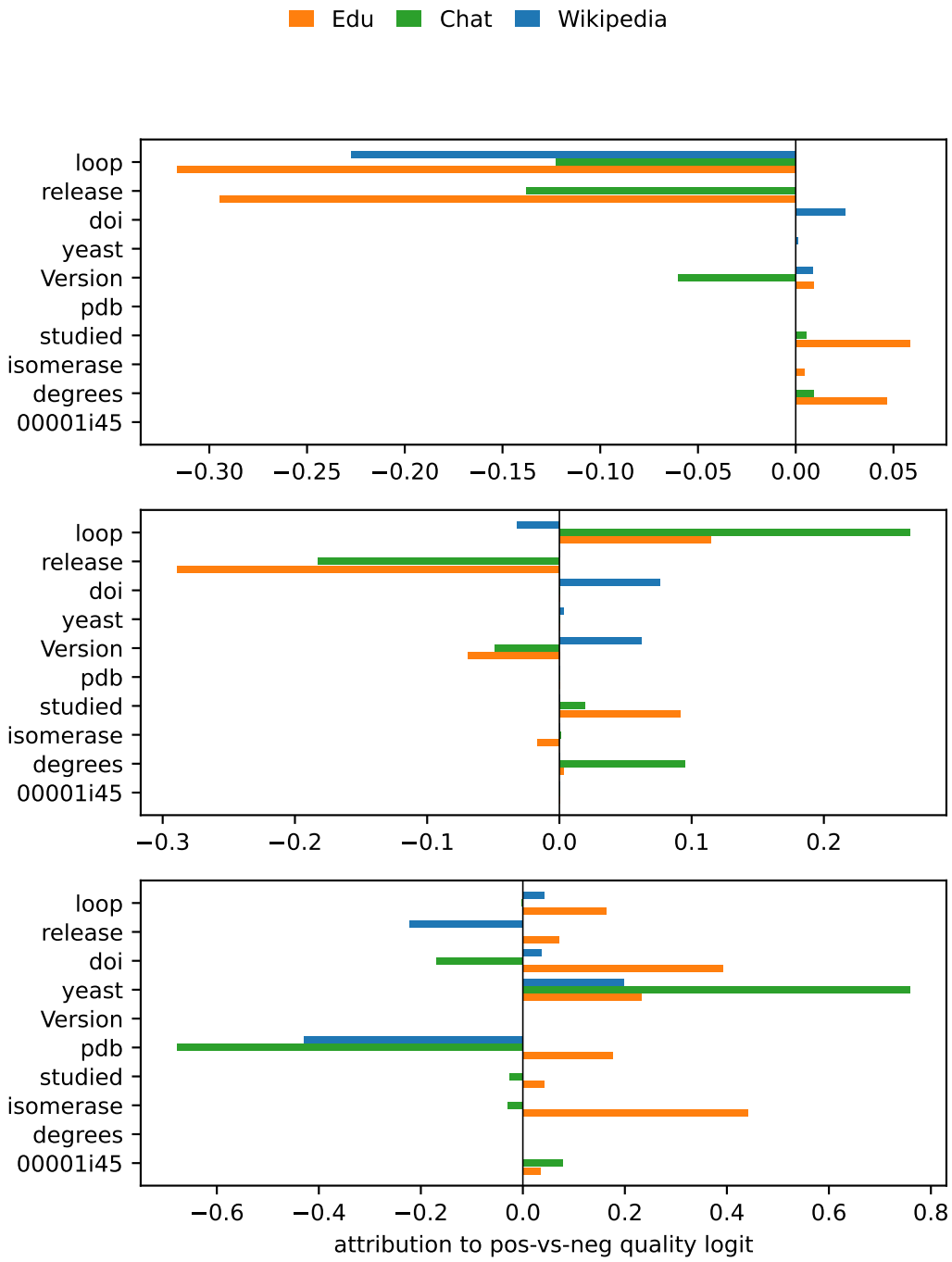

Figure 28: Word/token attributions for a held-out structural-biology database entry under all three backends (**top** fastText exact contribution, **middle** Luxical occlusion Δlogit, **bottom** BERT integrated gradients), on a shared set of tokens (each panel keeps its own magnitude scale); Edu orange, Chat green, Wikipedia blue. All three attribute the Edu and Wikipedia scores to scientific and citation vocabulary; the Chat series finds no overall positive signal.

H ABLATION MODEL ARCHITECTURES

The blending results of Sections 3.1–3.2 are read off two pretraining backbones: a 412M-parameter dense Transformer used for cheap screening, and a sparse mixture-of-experts (MoE) model with 7.7B total and ∼650M active parameters, which we denote 7.7B-A0.65B in the conventional total-active MoE notation, used for confirmation. We train both on the identical data arm and tokenisation for a given comparison, so the only difference between the two settings is the architecture. This appendix specifies each model completely and states why the design choices are the standard ones, so that a reader can judge whether a data conclusion is likely to be architecture specific.

Why two backbones. A data intervention that helps only one architecture is not a property of the data. The MoE we confirm on has its own degrees of freedom (routing, expert count, load balancing, shared experts), and a filter tuned against it risks over-fitting those choices rather than the corpus. We therefore pair the MoE with an independent, publicly documented dense recipe whose data rankings can be checked against the open literature. Holding the data pipeline (tokeniser included) fixed and swapping only the architecture is the cleanest available test that a ranking reflects the data: pretraining data, not model size or even the Transformer-versus- state-space distinction, is the dominant determinant of loss-to-loss transfer, with the tokeniser the next largest effect (Mayilvahanan et al., 2025). The dense backbone also removes the MoE-specific axes from the comparison.

Dense 400M (DCLM/OpenLM 1×). The dense model is a faithful reproduction of the DCLM-400M-1× configuration (Li et al., 2024), trained in the open `open_lm` architecture family that DCLM uses. It is a pre-normalisation, decoder-only Transformer in the GPT-2/LLaMA lineage (Radford et al., 2019; Touvron et al., 2023):

- 412M parameters; 24 layers; model width $d = 1024$; 8 attention heads; head dimension 128; full multi-head attention (no grouped-query sharing at this scale); sequence length 2048; untied input and output embeddings.
- Gain-only (bias-free) LayerNorm on the residual stream, and full-projection QK-LayerNorm on the queries and keys for training stability (Dehghani et al., 2023; Gemma Team, 2024).
- SwiGLU feed-forward blocks (Shazeer, 2020) with hidden width 2816 (the $\frac{2}{3} \cdot 4d$ rule rounded to a multiple of 256).
- Rotary position embeddings (Su et al., 2024) with $\theta = 10^4$ and no length scaling. A maximally-faithful variant additionally matches OpenLM's NeoX-style rotate-half RoPE, low-precision gain-only LayerNorm, and $1/\sqrt{\text{fan-in}}$ truncated-normal (depth-scaled) initialisation; the ablations here use the default variant, and we verified the two agree in ranking.

The tokeniser is the only piece that changes across our uses of this model. The weighted-sum simplex sweep and the Luxical-blend confirmation (Section 3.1, Appendix B) use the GPT-NeoX 50k vocabulary (Black et al., 2022) that DCLM ships with; the gated rank-blend swap (Section 3.2) uses the 96k BPE vocabulary, so that the dense confirmation matches the tokenisation the MoE mixture was optimised around. Only the (untied) embedding and output projection resize; the transformer stack and the token budget are unchanged.

Training follows the DCLM recipe at the Chinchilla-optimal token horizon (Hoffmann et al., 2022): 20 tokens per parameter, i.e., 8.2B tokens over 7851 steps at a global batch of 512 sequences (local batch 32, 16 data-parallel shards). The optimiser is AdamW ($\beta = (0.9, 0.95)$, $\varepsilon = 10^{-8}$) with peak learning rate 3×10^{-3} decayed by a cosine schedule to 3×10^{-5}, weight decay 0.033, an output z-loss of 10^{-4}, and 2000 warmup steps. One model is trained per data arm per seed.

The dense model is deliberately small. It is adequate for the low-noise likelihood signals we use to rank data (Appendix I), but downstream accuracy at 400M is weak and only becomes dependable at $\geq$ 1B; we therefore treat the dense runs as a screen and confirm at the MoE scale rather than reading accuracy off them.

The 7.7B-A0.65B MoE. The MoE is the ablation model, 7.7B parameters in total of which roughly 650M are *active* per token (only a subset of experts fire). It is a hybrid sliding-window/full-attention sparse Transformer rather than a few-large-expert Mixtral-style model (Jiang et al., 2024); the fine-grained-plus-shared-expert design follows the DeepSeek line (Dai et al., 2024; DeepSeek-AI, 2024),

on the sparse-FFN premise established by GShard and Switch (Lepikhin et al., 2021; Fedus et al., 2022). Table 10 lists the configuration.

Table 10: 7.7B-A0.65B MoE configuration for the ablation runs. Values are the run-time settings used for those runs, not later defaults.

Component	Setting
Active parameters	$\sim$650M (sparse; total larger)
Width / heads / KV heads	1024 / 16 / 2 (grouped-query attention)
Head dimension	128
Depth	50 layers: 2 dense feed-forward, then 48 MoE
Attention pattern	sliding window of 513 tokens on four of every five layers; full attention on every fifth layer and the last
Normalisation	RMSNorm; per-head QK-RMSNorm ($\varepsilon = 10^{-6}$)
Expert / dense MLP	SwiGLU
Routed experts	256 experts, top-6, expert hidden width 192
Shared expert	1 (always on), hidden width 192
Router	token-choice top-k, sigmoid-logit scoring, $1/\text{fan-in}$ init
Load balancing	global quantile balancing with a residual score-bias term
Position embedding	rotary (CosSin), $\theta = 2 \times 10^3$
Tokeniser / vocabulary	96k BPE
Sequence length	16384
Token horizon	150 tokens/active-param ($\approx$97.6B tokens)
Steps	10342 (distribution-aware/mixture fleets); 7947 (weighted-sum Luxical confirm)
Parallelism	64-way data-parallel shard, local batch 9

Reasoning for MoE choices. Grouped-query attention (Ainslie et al., 2023) is the now-standard key/value-cache reduction from the LLaMA-2/3 and Mistral families. The hybrid of a short sliding window with periodic full-attention layers is the long-context-efficient pattern popularised by Longformer and adopted by Mistral and Gemma-2 (Beltagy et al., 2020; Jiang et al., 2023; Gemma Team, 2024); the two leading dense feed-forward layers before routing are the DeepSeek-style stability convention. Fine-grained experts with a shared always-on expert and sigmoid (rather than softmax) routing scores follow DeepSeekMoE/DeepSeek-V3 (Dai et al., 2024; DeepSeek-AI, 2024), which report better compute-quality trade-offs than the coarse expert granularity of Mixtral (Jiang et al., 2024). RMSNorm and QK-normalisation are the standard stabilisers (Zhang & Sennrich, 2019; Dehghani et al., 2023), and rotary embeddings are as above (Su et al., 2024). Peak learning rates are set per parameter group by a fitted scaling law: for a budget of T billion tokens the two-dimensional backbone uses Muon (Jordan et al., 2024) at $\eta = 0.555\,T^{-0.4073}$, embeddings and one-dimensional parameters use 2η with AdamW, and the attention/router heads use $(\eta/32)/(d/2048)$; at this scale ($T \approx 97.6$) this puts the backbone learning rate near 0.086. The optimiser, schedule, and load-balancing rule are training-dynamics choices held fixed across every data arm, so they cannot confound a mixture comparison; they are documented here only for completeness.

Dense versus MoE. Table 11 summarises the two backbones side by side. The two settings differ in three respects that matter for reading a ranking: the MoE is trained on many more tokens with repetition, on a larger vocabulary, and with a sparse routed feed-forward. A ranking is therefore read within a backbone.

Table 11: The two ablation backbones. The dense model screens; the MoE confirms.

	Dense 400M	7.7B-A0.65B MoE
Active parameters	412M	$\sim$650M
Feed-forward	dense SwiGLU	256-expert top-6 MoE (+1 shared)
Attention	full multi-head	GQA, hybrid sliding/full window
Sequence length	2048	16384
Tokeniser	NeoX 50k (weighted-sum) / 96k (gated blend)	96k
Token horizon	8.2B (1$\times$ Chinchilla)	$\approx$97.6B (150 tok/param)
Optimiser	AdamW	Muon (backbone) + AdamW
Role	screening	confirmation

I EVALUATION PROTOCOL

Every mixture comparison in the paper is decided on held-out *bits-per-byte* (BPB), aggregated into a small number of capability areas and read through Bayesian pairwise posteriors. This appendix gives the metric, the task suites run at each scale, how the capability aggregates are formed, and why these choices suit ranking pretraining data at small scale.

Why bits-per-byte, not accuracy. For a scored continuation x the metric is

$$\mathrm{BPB}(x) \;=\; \frac{-\log_2 p_\theta(x)}{|x|_{\text{UTF-8}}}, \tag{3}$$

the model's negative log-likelihood of the target divided by its length in UTF-8 bytes (Magnusson et al., 2024). Lower is better. Three properties make it the decision metric.

Tokeniser invariance. The denominator is bytes, not tokens, so a 50k and a 96k tokeniser sit on the same scale and a data ranking is not confounded by differences in token fertility. Empirically the full-text MMLU BPB ranking is the most stable across the two tokenisers we use (Spearman $\rho \approx 0.89$).

Signal at small scale. A likelihood on the full answer text is a dense, low-variance signal, whereas multiple-choice accuracy at 400M is close to chance and only becomes reliable at ≥ 1B (Magnusson et al., 2025; Li et al., 2024). We also avoid letter-scored BPB (scoring only the option letter, e.g., the two bytes of " B"), which is near-random and high-variance at these scales; where we score MMLU by likelihood we use the full-text variant, which scores the *text* of the correct option rather than its label.

Comparability. BPB as defined in Eq. equation 3 is a standard cross-model reporting convention (Magnusson et al., 2024), so our numbers are commensurable with open baselines rather than harness-specific.

Accuracy is used only as a secondary diagnostic. Where we report it, in the single-construct ablations of Appendix A, it is a DataDecide-style centred-accuracy aggregate (Li et al., 2024; Magnusson et al., 2025); the main-text blending decisions are made on BPB, not on accuracy.

Which benchmarks, and why. Task implementations follow the OLMES standard (Gu et al., 2024), which fixes the few-shot count, the prompt format, and the length-normalised log-likelihood scoring rule per task. Using OLMES rather than ad-hoc harness variants makes our per-benchmark numbers comparable to OLMo/DataDecide (Groeneveld et al., 2024; Magnusson et al., 2025) and keeps the scoring identical across the dense and MoE fleets.

The suite is chosen to cover the capabilities a quality filter is supposed to move, in both languages of the corpus:

- *Knowledge / QA:* MMLU and full-text MMLU (Hendrycks et al., 2021), ARC, PIQA (Bisk et al., 2020), and a block of OLMo-Base-Easy QA (CommonsenseQA, SocialIQA, SciQ, DROP, Jeopardy, NaturalQuestions, SQuAD, MedMCQA/MedQA).
- *Language modelling:* HellaSwag (Zellers et al., 2019), WinoGrande (Sakaguchi et al., 2020), and LAMBADA (Paperno et al., 2016). On the MoE fleets we prefer GoldenSwag (Chizhov et al., 2025), the decontaminated HellaSwag subset, and drop plain HellaSwag from the aggregate where the cleaner signal is available.
- *Code:* HumanEval (Chen et al., 2021) and MBPP (Austin et al., 2021), scored by BPB.
- *Maths:* GSM8K (Cobbe et al., 2021) and MATH-Minerva, scored by BPB where the metric is healthy (see below).
- *German:* Global-MMLU-DE (Singh et al., 2024) and German ARC/PIQA/HellaSwag ports, plus German code and maths BPB, so an English-only view would miss cross-lingual regressions.

Two exclusions are deliberate. MATH-Minerva BPB is dropped from an aggregate whenever the metric returns a degenerate 0.0 (as it does for the 400M English suite), so a broken key cannot enter a mean; maths is claimed only on fleets where the key is healthy. And completion/execution accuracy, F1, and abstention ("IDK") metrics, though logged on some MoE suites, are omitted from the decision rule because they are high-variance at these scales.

Suites run at each scale. The dense 400M suite is BPB-only. In English it runs the OLMES MMLU/ARC/PIQA/HellaSwag set, full- text MMLU, HumanEval/MBPP BPB, and the OLMo-Base-Easy extras (WinoGrande, LAMBADA, CommonsenseQA, SocialIQA, DROP, Jeopardy, NaturalQuestions, SQuAD, SciQ, BasicSkills, MedMCQA/MedQA, MT-MBPP). In German it runs Global-MMLU, the ellamind ARC/PIQA/HellaSwag ports, MATH-Minerva-DE BPB, and HumanEval/MBPP-DE BPB.

The 7.7B-A0.65B MoE suite overlaps this English core (MMLU, ARC, PIQA, HellaSwag, HumanEval/MBPP BPB) and adds GSM8K and MATH-Minerva and GoldenSwag, over the same German block. Where a fleet's in-training evaluation does not emit the cleaner language-modelling keys, they are supplied by a separate backfill (GoldenSwag, WinoGrande, LAMBADA) and folded into the aggregate.

Capability aggregates. Each capability area is the unweighted mean of its member BPB values, and the English-overall (and German-overall) figures are the mean over the concatenated member keys of that language, computed directly from the per-benchmark BPB rather than a precomputed rollup so that a missing member simply drops out. The exact membership tracks what each fleet's evaluation emits:

- *Weighted-sum Luxical confirmation, dense.* The DCLM English taxonomy: QA (the OLMES knowledge set), code (HumanEval, MBPP, MT-MBPP), language modelling (HellaSwag, WinoGrande, LAMBADA), and full-text MMLU. Fig. 10 (top) plots each benchmark individually; the English-overall mean concatenates their keys.
- *Weighted-sum Luxical confirmation, MoE.* The same taxonomy, except the Language-Modelling area is plain HellaSwag BPB (the only LM key that fleet's evaluation emits). Fig. 10 (bottom) plots each benchmark the fleet reports (ARC, MMLU, PIQA, HumanEval, MBPP, HellaSwag).
- *Figure 1 panel 3.* At 400M, QA is the twelve English knowledge tasks of the sixteen-benchmark fleet suite, code is HumanEval, MBPP, and MT-MBPP, and language modelling is WinoGrande. At the MoE the mean is those same benchmarks with WinoGrande left out, so it keeps the twelve QA tasks and the three code tasks and has no language-modelling member. EN is the unweighted mean of that suite. Fig. 4 plots every benchmark in that mean.
- *German benchmarks of those arms (Fig. 5).* Global-MMLU, PIQA, HumanEval, and MBPP, at both scales. These four are an out-of-distribution check: the filters are fit on English data only, and a German gain is positive transfer. The aggregate in Section 4 is the unweighted mean of the four, over seeds that scored all four. One 400M optimum seed's final eval is missing Global-MMLU, so that seed is left out of the aggregate and that bar uses the other three seeds. PIQA, HumanEval, and MBPP use all four.

Because area membership differs slightly between fleets, we compare arms *within* a fleet and never mix an aggregate across suites.

From aggregates to a verdict. A point difference alone does not settle a comparison. For each aggregate and each pair of arms we place a non-informative (Jeffreys) prior on the per-arm mean, giving a Student-t posterior over the mean from the across-seed spread, draw 20,000 samples per arm, and report the trichotomy $P(\text{better})$, $P(\text{ROPE})$, $P(\text{worse})$ under a region of practical equivalence of $\pm 1\%$ of the arms' magnitude (Kruschke, 2018). A one-seed arm has no within-arm spread; for the verdict its posterior is then a normal centred on that seed, with scale equal to the pooled within-arm standard deviation of the other arms in the comparison. Reporting the ROPE mass, rather than a p-value, lets a small seed count ($n = 3$–5, and the one-seed MoE random arm) express 'indistinguishable' honestly instead of forcing a direction.

J DISTRIBUTION-AWARE COMBINED SCORE VS RAW SOURCE SCORES

The distribution-aware recipe of Section 3.2 maps each of the ten gate×rater basis columns to its mid-rank empirical CDF and combines them by a null-aware weighted mean (Eq. equation 1). Figure 29 characterises the resulting per-document combined score $S(d)$ against each of the five *raw* source scores it is built from, over the $1{,}052{,}333$-row output sample whose combined score is recomputed directly from the raw columns with the Dolma 3 gate ($\tau = 0.0216$).

Three things are visible. First, no single raw source explains the blend: the panel-wise Pearson correlations span 0.49 (chat-like anchor) to 0.69 (content-focussed score), with Dolma 3 at 0.68. A rank-normalised gated blend of ostensibly-correlated quality scores is therefore not a stand-in for any one of them, the same non-collapse that Section 2.2 establishes for the raw judgements. Second, the Dolma 3 panel shows the gate as geometry: an empty upper-left region (low raw Dolma 3 cannot reach a high combined score, since the identity×Dolma 3 term carries the largest weight, 0.3351) and a sharp lower envelope where the $\sim 86\%$ of documents below the gate median collapse onto a narrow low-S band through the shared mid-rank of the gated-out zero. Third, the four Luxical panels are monotone-increasing but noisy, split by a horizontal discontinuity near $S \approx 0.5$: the identity-gated and Dolma 3-gated halves of the basis place documents on either side of the Dolma 3 gate into separate combined-score bands, so the same raw Luxical score maps to two different blended scores depending on whether the document clears the gate.

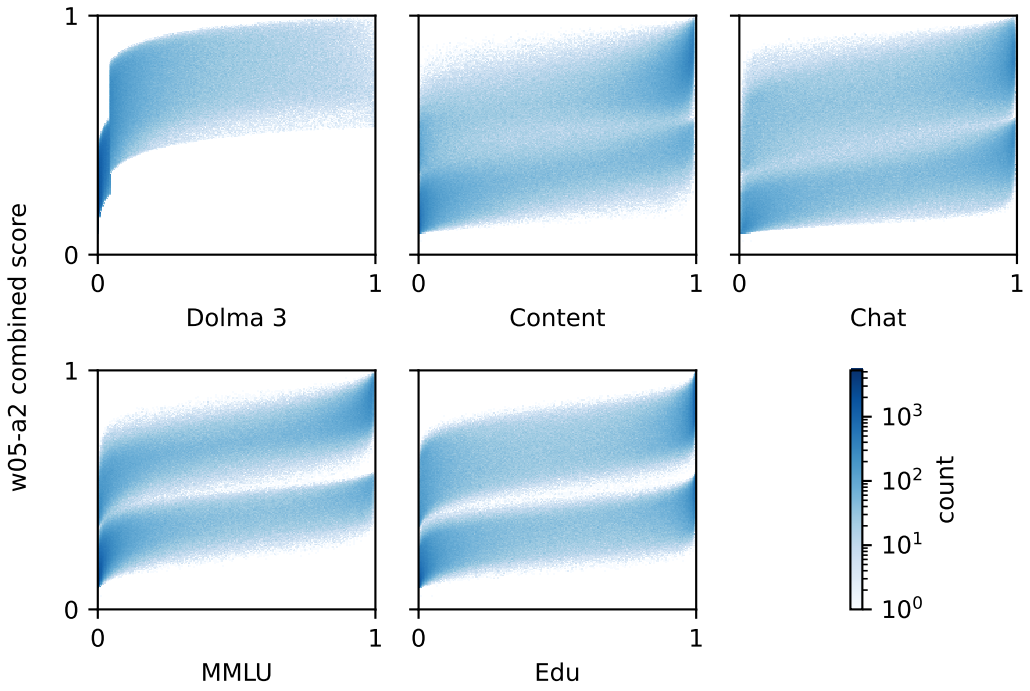

Figure 29: The gated rank-blend combined score $S(d)$ (Eq. equation 1) against each of the five raw source scores it is built from, as log-count 2D histograms over the $1{,}052{,}333$-row sample (combined score recomputed from the raw columns with the Dolma 3 gate). Per-panel Pearson r spans 0.49–0.69: the blend is multi-source, not a proxy for any single head. The Dolma 3 panel exposes the gate (empty upper-left, sharp lower envelope); the Luxical panels split into two horizontal bands at the Dolma 3 gate.

K ANNOTATED DOCUMENT EXEMPLARS

To avoid redistributing copyrighted web text, we do not reproduce the document bodies. Each exemplar is instead identified by its Common Crawl `warc_id`, and Table 12 lists all of them with their annotations and a short editorial description. Together they illustrate, one dissociation at a time, why a single quality scalar loses information: the extreme tail where diverse constructs converge, the mid-range where they cleanly split, and the low tail rejected by every metric.

Reading the source documents on Common Crawl. The `warc_id` is the record's `WARC-Record-ID` (the `urn:uuid` of the WARC response record) in the publicly available Common Crawl archives; our sample is the English, substring-deduplicated Common Crawl sample used for judging (Section 2.1). Common Crawl's public index is keyed by URL and capture time rather than by record id, so retrieval proceeds by locating the capture and then confirming the record id:

1. Look up the capture by its origin URL in the Common Crawl URL index, either through the CDX API at `https://index.commoncrawl.org/` or the columnar index under `s3://commoncrawl/cc-index/table/cc-main/warc/`. The matching index row gives the WARC file path together with the record's byte offset and length.
2. Fetch that byte range from `https://data.commoncrawl.org/` (an HTTP `Range` request on the WARC file path, or an equivalent read from `s3://commoncrawl/`) and parse the returned record with any WARC reader (e.g., `warcio`).
3. Confirm you have the intended capture by matching the record's `WARC-Record-ID` header against the `warc_id` listed here.

Because step 1 is URL-keyed, the origin URL is required to locate a record; the `warc_id` then uniquely and permanently pins the exact capture within the Common Crawl archives. Potentially sensitive content should be treated with care, and we redact it from any description below.

Table 12: All annotated exemplars, identified by Common Crawl `warc_id` (WARC-Record-ID). Document bodies are not reproduced; see the retrieval instructions above. Scores are on the judge's native scales (1–5 for rubric and anchor axes; screen labels as noted).

Dissociation	warc_id	Key annotations	Description
Convergent high quality	25b243a3-6aa9-4d92-8730-950f4608681d	rubric informational, educational, coherence, factual all $= 5$; educational anchor $= 5$	Clinical reference on viral meningitis: dense, specific, structured; jointly top-ranked under two differently-framed prompts.
Content vs. educational	5c91944c-1ef3-4faf-b210-918289ad4a4a	rubric informational $= 5$, educational value $= 1$, factual density $= 5$	Store contact-information page: maximally specific and factual, yet judged of no educational value.
Benchmark-useful (rare)	319477ca-f38b-4d76-9d65-c784dd8c060f	MMLU-usefulness usefulness, content, style, format all $= 5$	NASA technical proceedings: one of the very few documents any benchmark-usefulness prompt rates maximally useful.
Content-focussed high	f67076d0-64c6-497f-bbe6-a10cc61a6bdb	content-focussed score $= 5$	Australian legislative-drafting explainer: substantive professional content, scored highly even with formatting ignored.
Format-focussed high	fe0d5e02-4b1b-472f-b3cb-01fc1ddcc1c7	format-focussed score $= 5$	Pharmaceutical summary of product characteristics: rigid, consistently sectioned structure regardless of content.

continued on next page

Table 12 continued from previous page.

Dissociation	warc_id	Key annotations	Description
Anchor low	d51679ad-7790-4703-a257-c3a1d986eca0	educational anchor = 1	News brief on a court filing: fluent and coherent but rated minimally educational.
Anchor low	4d338690-5c50-4197-8ee7-f23f204015db	educational anchor = 1	Travel-nurse job listing: boilerplate recruitment down-weighted by the educational anchor.
Educational exemplar	7ee7aa49-642b-4783-9da4-fec4721ead29	educational anchor = 5	Bar-exam flashcards: explicitly didactic Q&A, the prototypical 'educational' document.
Ask-LLM negative	f8af60a7-2dda-40b4-b70d-5dd8e8638095	Ask-LLM informativeness: rejected	Incoherent essay-mill page with high non-native-English features.
Ask-LLM negative	43e00802-15c1-4419-a574-dce78f243d41	Ask-LLM informativeness: rejected	Templated adult-services spam; agreed low quality by other metrics.
Provenance: Wikipedia	dc1a0dee-65b5-4644-b3a2-14b9c53c0270	Wikipedia anchor $\sim$ 3.1	Encyclopedic prose on homonymous hemianopia.
Provenance: arXiv/peS2o	944dc094-6033-4355-b8b3-246e222b38fb	educational/Wikipedia anchor $\sim$ 3.7; chat-like anchor $\sim$1.45	Scholarly abstract on CRISPR-Cas9.
Provenance: chat-like data	46dc3c77-8961-4950-814c-e77791711570	rarest provenance class ($\sim$1.2k/1M)	Chat-like conversation log.
Conversational positive	de562360-714b-4e9f-bc07-b90b12ee30e8	conversational screen = 1	Multi-participant forum thread, one of the $\sim 6\%$ the conversational screen labels positive.