# TOKEN MASS, NOT GEOMETRY: A CONFOUND IN EMBEDDING-BASED DATA SELECTION FOR LANGUAGE MODEL PRETRAINING

**Anonymous authors**
Paper under double-blind review

## ABSTRACT

Cluster/embedding-based data-selection methods choose pretraining subsets by geometry: representativeness (proximity to a cluster centroid), atypicality, or redundancy. We stress-test this family under matched training compute, three seeds, a common held-out evaluation, and an explicit random baseline. Under the usual *fixed-document-budget* protocol, near-centroid ("representative") selection beats random on out-of-distribution (OOD) loss by a small, seed-stable margin ($\sim$0.01 nats on C4). We show this gain is a **retained-token-mass confound**, not a property of embedding geometry. Centroid distance is strongly anti-correlated with document length ($\rho = -0.413$ in an LSA space, $\rho = -0.686$ in the OPT-125M space), so at a fixed document budget, representative selection keeps longer documents and hence *more tokens*, training with less repetition. When we instead match the *token* budget across arms, the benefit vanishes: near-centroid selection ties or slightly underperforms random. This holds across model scales (124M and 350M, three seeds each) and two OOD sets, and a progression of controls shows the near/far-centroid gap shrinking from $0.044$ nats (document budget) to $0.016$ (length-stratified) to $0.002$ (token-matched). Native SemDeDup, evaluated in its own OPT-125M embedding space, also does not beat random on this corpus. We recommend that evaluations of embedding-based selection control for retained token mass and document length, and report against a random baseline at matched compute with multiple seeds: practices that, in our experiments, overturn an apparently positive result.

## 1 INTRODUCTION

Data selection is a central lever for efficient language model (LM) pretraining. A common recipe embeds a document pool, clusters it, and retains a subset by a geometric criterion: keep the most *representative* examples near cluster centroids, keep *atypical* examples far from them, or remove near-duplicate *redundant* examples (Sorscher et al., 2022; Abbas et al., 2023; Tirumala et al., 2023). Such methods are commonly reported to match or beat full-data training on a fraction of the tokens.

Their evaluations, however, share recurring weaknesses: single model scale, single training run, single evaluation distribution, and, critically, no control for confounds correlated with the selection score. We focus on one such confound and show it accounts for the apparent benefit of geometric selection at the scales we study. When selection retains a fixed number of *documents* but arms differ in the *length* of the documents they keep, they end up training on different numbers of *tokens*, and therefore with different amounts of data repetition at a fixed compute budget. If the selection score is correlated with length, an evaluation attributes to "embedding geometry" what is really a token-budget/repetition effect.

Why single out this confound? Selection criteria are motivated as capturing *semantic* properties, how representative, diverse, or redundant a document is and their gains are attributed to that content. Document length, by contrast, is a trivial surface statistic that no one intends to select on, yet it can ride along with an embedding-based score and move held-out loss on its own. If a large share of a method's reported benefit is attributable to length (through the tokens it drags in), three things follow. The claimed *mechanism* is largely illusory: the model is not benefiting from "representativeness"



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.

but from seeing more, longer text with less repetition. *Comparisons* between methods that induce different length distributions are not apples-to-apples, because they implicitly compare different token budgets. And *transfer* is unreliable: a practitioner applying the method to a corpus with a different length profile, or under a token rather than document budget, may see no benefit at all. Establishing whether this occurs requires only a few inexpensive controls, which, as we show, are routinely omitted.

We make this concrete. Our contributions are:

1. **A retained-token-mass confound.** At a fixed document budget on C4, near-centroid selection beats random on OOD loss, and OOD loss across selection arms is almost perfectly rank-ordered by the number of retained tokens. Centroid distance is strongly anti-correlated with document length ($\rho = -0.413$ in LSA space), so representative selection systematically keeps longer documents and more tokens (Section 4.2).
2. **Token-matched controls remove the effect.** When arms are matched on *token* budget rather than document count, near-centroid selection no longer beats random—it ties or slightly underperforms it. A progression of controls (document budget $\rightarrow$ length-stratified $\rightarrow$ token-matched) shrinks the near/far-centroid gap from $0.044$ to $0.016$ to $0.002$ nats (Section 4.2).
3. **The effect replicates across scale, embedding space, and OOD set.** The document-budget gain and its removal under token matching hold at both 124M and 350M parameters (three seeds each); the length–centroid coupling is present and *stronger* in a neural embedding space ($\rho = -0.686$ for OPT-125M); and the arm ordering reproduces on a second OOD distribution (Wikipedia) (Sections 4.2–4.3, Appendix B).
4. **A fair SemDeDup baseline and a prediction result.** Native SemDeDup in OPT-125M space does not beat random on this corpus (Section 4.4); and a linear lexical (LSA) projection predicts per-document difficulty better than semantic embeddings but yields no selection benefit (Section 4.1).

We do not propose a new selection method. Our contribution is a controlled demonstration that a specific, cheap-to-check confound accounts for apparent gains from embedding-based selection at the scale we study, with recommendations for evaluation. Prior work has noted that off-the-shelf embeddings are imperfect proxies for pretraining-loss similarity (Sam et al., 2025) and that model-based filters can exhibit document-length biases; we sharpen this into a controlled, quantified confound analysis for cluster-geometry selection.

## 2 Related Work

**Data selection and pruning.** Many methods score or prune examples by per-example signals: error/gradient scores such as EL2N and GraNd (Paul et al., 2021), forgetting statistics, and reference-model perplexity, where Marion et al. (2023) find medium-perplexity documents can match full-data training. A geometry-based family scores examples by position relative to embedding clusters. Sorscher et al. (2022) showed data pruning can beat power-law scaling and that the optimal direction depends on data abundance: keep hard/atypical examples when data are plentiful, prototypical ones when scarce. Our *typical*/*atypical* arms instantiate these two directions.

**Deduplication.** SemDeDup (Abbas et al., 2023) clusters embeddings and removes near-duplicates within clusters; D4 (Tirumala et al., 2023) adds prototypicality pruning. Their strongest gains are reported on large, comparatively uncurated corpora with abundant redundancy. We evaluate SemDeDup natively in OPT-125M space and find it does not beat random on our (curated) corpus at the scale studied; we treat this as evidence about this corpus and scale, not a verdict on the method.

**Data mixing.** A separate line reweights source *proportions* rather than selecting individual documents: DoReMi (Xie et al., 2023a) optimizes domain weights via group distributionally-robust optimization on a small proxy model; DSIR (Xie et al., 2023b) resamples by $n$-gram importance weights toward a target distribution; and CLIMB (Diao et al., 2025) searches cluster-mixture weights with proxy models. These operate on an axis orthogonal to ours: we hold the cluster mixture fixed (per-cluster budgets proportional to cluster size) and vary only *within-cluster* selection. We therefore do

not compare against mixing methods experimentally, but note that a length/token-mass confound is equally worth checking there, since reweighting sources also changes the length distribution of the retained data.

**Embeddings for selection, and confounds.** Sam et al. (2025) analyze how well embedding-space distances predict pretraining-loss generalization and how diversity-based curation with different embeddings affects pretraining, finding off-the-shelf embeddings underperform embeddings trained on the target data, directly relevant to our Section 4.1 and clustering setup, and consistent with our finding that a simple lexical projection can out-predict semantic encoders. Document-length biases in model-based data filtering have also been observed, where quality filters systematically favor shorter (or longer) documents. Our contribution relative to this prior awareness is specific: we run *token-matched* and *length-stratified* controls against cluster-geometry selection under a matched-compute, multi-seed protocol, and show the confound is load-bearing, it accounts for essentially all of the apparent benefit at the scales we study, rather than being a secondary caveat.

**Geometric instance reduction.** Our atypicality arm and its linear-projection variant are inspired by classical instance-reduction methods in low-dimensional PCA spaces (Joshi & Haspel, 2020); our result highlights that in learned high-dimensional embeddings this geometry couples strongly to document length.

## 3 Experimental Setup

**Corpora.** Main experiments use C4 (Raffel et al., 2020) (`en`), a pool of $6\times10^6$ documents ($\sim$3B tokens). We evaluate OOD on held-out FineWeb-Edu (Penedo et al., 2024) ($\sim$53M tokens) and, as a second OOD set, English Wikipedia ($\sim$62M tokens; Appendix B). A difficulty-prediction experiment (Section 4.1) uses a FineWeb-Edu sample.

**Embeddings.** (i) A *linear lexical* (LSA) embedding: TF–IDF over 30k terms, truncated SVD to 300 components. (ii) OPT-125M (Zhang et al., 2022) last-layer, last-token hidden states (native SemDeDup space; also used for the cross-embedding correlation). (iii) Semantic sentence encoders MiniLM (Wang et al., 2020) and GTE-large (Li et al., 2023), used in the prediction experiment.

**Clustering and mixture.** $k$-means ($k = 8000$, cosine-normalized) via FAISS (Johnson et al., 2019). Per-cluster budgets are proportional to cluster size, so all arms preserve the cluster mixture and differ only in within-cluster selection.

**Selection arms and budgets.** Criteria (all in LSA space unless noted): *random*; *typical/atypical* (retain nearest/farthest documents to the centroid $\delta_i = \|\mathbf{e}_i - \boldsymbol{\mu}_{c(i)}\|_2$, the SSL-prototype score of Sorscher et al. 2022); *semdedup* (retain least-redundant by max intra-cluster cosine similarity; Abbas et al. 2023), in LSA and OPT spaces; *long/short* (retain longest/shortest documents by *tokenizer* length). We use two budget conventions: **document-budget**, retaining $50\%$ of documents per cluster (the standard protocol), and **token-matched**, retaining documents in the criterion's order until $50\%$ of the cluster's *tokens* are kept—equalizing total tokens (hence repetition) across arms.

**Model, training, evaluation.** We train GPT-2-style models at 124M and 350M parameters (context 1024) from scratch with `nanoGPT` (Radford et al., 2019; Karpathy, 2022). Every arm is trained at *matched compute* (124M: $\approx$0.98B tokens/run; 350M: $\approx$3.1B), sampled with replacement, identical optimizer settings (Appendix A), **three seeds**, reported as mean$\pm$std. All arms are scored on the same held-out set with identical evaluation batches.

**Formal definitions.** Let $\mathbf{e}_i \in \mathbb{R}^d$ be the normalized embedding of document $i$, $c(i)$ its cluster, and $\boldsymbol{\mu}_c = \frac{1}{|c|}\sum_{j\in c}\mathbf{e}_j$ the cluster centroid. Prototypicality scores documents by centroid distance $\delta_i = \|\mathbf{e}_i - \boldsymbol{\mu}_{c(i)}\|_2$ (retain small $\delta_i$ for *typical*, large for *atypical*); redundancy by maximum intra-cluster similarity $\sigma_i = \max_{j\in c(i), j\neq i}\langle \mathbf{e}_i, \mathbf{e}_j\rangle$ (retain small $\sigma_i$ for *semdedup*); and length by tokenizer length $\ell_i$. Writing $r = 0.5$ for the retention fraction, the **document-budget** rule keeps the top $r\,|c|$ documents of each cluster by the criterion, whereas the **token-matched** rule keeps documents in criterion order until their cumulative length first reaches $r\sum_{j\in c}\ell_j$. The document-budget rule fixes

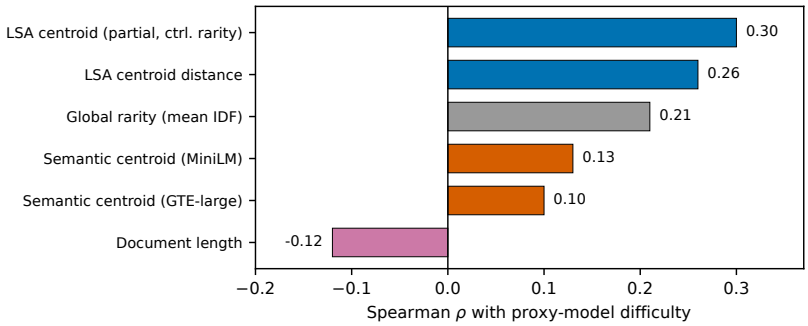


Figure 1: Rank correlation (Spearman $\rho$) between per-document centroid distance and proxy-model difficulty (Pythia-160M NLL) on FineWeb-Edu. The linear LSA projection predicts difficulty best and its power survives controlling for global rarity; semantic embeddings are weaker.

the *count* of retained documents but lets total retained tokens $\sum_{i \in \text{kept}} \ell_i$ vary across arms; the token-matched rule fixes total retained tokens (to within $0.1\%$ in our data) and lets the document count vary. Because all arms train the same number of tokens at matched compute, only the token-matched rule equalizes the amount of data repetition, which is the quantity our analysis isolates.

# 4 Results

## 4.1 A linear lexical projection predicts difficulty better than semantic embeddings

On a 200k-document FineWeb-Edu sample we take per-document NLL under Pythia-160M (Biderman et al., 2023) as a difficulty target and measure the rank correlation between a document's centroid distance and its difficulty in each embedding space (Figure 1). The linear (LSA) projection is the strongest predictor ($\rho = 0.26$), and controlling for global word rarity *increases* its predictive power (partial $\rho = 0.30$), indicating topic-relative lexical atypicality rather than mere rarity; the signal grows with retained variance (100→800 components: $\rho$ 0.19 → 0.26). Semantic encoders are weaker predictors ($\rho \approx 0.10$–$0.13$), consistent with a small model's per-document loss being driven more by surface lexical statistics than by meaning-level features, in line with evidence that networks learn lower-order, statistically simpler structure earlier in training (Belrose et al., 2024). This is a positive finding about *prediction*; as the remaining sections show, it does not yield a *selection* benefit, predictive correlation is necessary but not sufficient for a useful selection signal.

## 4.2 The document-budget gain is a retained-token-mass confound

Under the standard document-budget protocol, near-centroid selection beats random on OOD loss (Table 1, Figure 2): *typical* 3.717 vs. *random* 3.727. But Table 1 has a striking property: **OOD loss is almost perfectly rank-ordered by the number of retained tokens** (Spearman $\rho \approx -1$ between selected tokens and loss), as Figure 2 makes visible. Because every arm trains the same $\sim$0.98B tokens sampled with replacement, arms with smaller pools repeat data more; *typical* keeps nearly twice the tokens of *atypical* despite the same document count.

The reason is a strong coupling between centroid distance and document length: on C4, $\rho(\text{length}, \text{centroid distance}) = -0.413$ in LSA space (Figure 3) and $-0.686$ in the OPT-125M space. Near-centroid documents are systematically long, so representative selection is, in part, long-document selection. This coupling is not an LSA artifact—it is *stronger* in a modern neural embedding space. In a bag-of-words space the direction is expected: a long document has many terms and a denser, more "average" term vector near its cluster mean, whereas a short document is sparse and idiosyncratic.

**Why more retained tokens lowers loss.** We treat the following as a hypothesis rather than a claim. At a fixed compute budget, an arm with a larger token pool repeats data less, and less repe-

Table 1: Document-budget arms on C4 (retain 50% of *documents* per cluster). OOD loss on FineWeb-Edu, mean±std over 3 seeds (124M). "Tokens" is the retained pool size. OOD loss is monotone in retained tokens; *typical*/*atypical* differ ∼2× in tokens at equal document count.

| Arm | Selection criterion | Tokens | OOD loss |
|---|---|---|---|
| *long* | longest documents | 2.27B | $3.716 \pm 0.002$ |
| *typical* | near-centroid (LSA) | 1.98B | $3.717 \pm 0.003$ |
| *random* | uniform | 1.49B | $3.727 \pm 0.001$ |
| *semdedup-OPT* | dedup, OPT-125M space | 1.48B | $3.730 \pm 0.003$ |
| *semdedup-LSA* | dedup, LSA space | 1.06B | $3.759 \pm 0.001$ |
| *atypical* | far-centroid (LSA) | 1.01B | $3.761 \pm 0.002$ |
| *short* | shortest documents | 0.72B | $3.783 \pm 0.001$ |

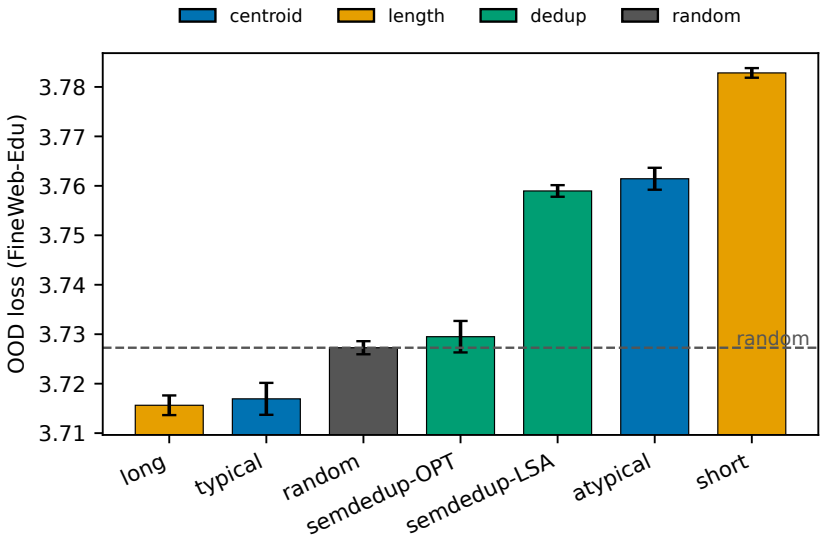


Figure 2: Document-budget OOD loss across arms (mean±std, 3 seeds; 124M), colored by selection type. The arms that retain the most tokens (*long*, *typical*) achieve the lowest loss and the arms that retain the fewest (*short*, *atypical*) the highest, with *random* and native SemDeDup in between; the dashed line marks random. The ordering tracks retained token mass rather than a semantic property of the criterion.

tition generally improves held-out loss. Independently, mean per-token loss is dominated by tokens deep inside a document, predicted with substantial left context; the first tokens of a document, predicted with little context, are comparatively high-loss. A corpus of longer documents therefore contains a smaller fraction of low-context, high-loss "document-onset" positions and offers more long-range context per example. Both effects lower held-out loss, and both are consequences of retaining more/longer text—not of embedding-space representativeness. Disentangling the repetition and context components is left to future work; for our purposes it suffices that the effect is driven by token mass, which the token-matched control equalizes.

**Controlling the confound.** We isolate geometry from token mass with two controls. (i) *Length-stratified*: within each (cluster × length-decile) cell, retain the near- or far-centroid half, so the two arms share nearly the same length distribution. (ii) *Token-matched*: retain documents until $50\%$ of a cluster's *tokens* are kept, equalizing total tokens (and hence repetition) across arms to within $0.1\%$ (all ≈1.496B). Table 2 traces the effect through these controls (and Figure 4 shows the length-stratified step visually): the near/far-centroid gap collapses from $0.044$ nats (document budget) to $0.016$ (length-stratified) to $0.002$ (token-matched), and the benefit of *typical* over *random* goes from

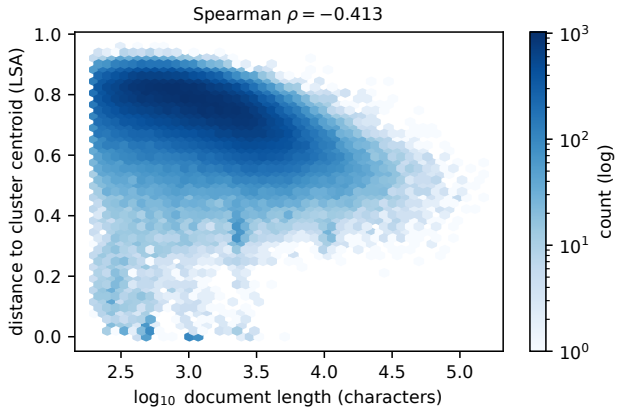


Figure 3: Per-document distance to cluster centroid (LSA) vs. document length on C4 (hexbin; color is log count). Longer documents lie closer to their centroid ($\rho = -0.413$; $-0.686$ in OPT space), so near-centroid selection preferentially retains long documents.

Table 2: Controlling the confound (C4, OOD loss, 3 seeds). "typical − random": negative means typical is better. "near/far gap": *typical* vs. *atypical*. Progressively controlling token mass removes the effect. 350M values from Section 4.3 (*atypical* was not run at 350M, so the near/far gap is reported at 124M only).

| Quantity (nats) | 124M | 350M |
|---|---|---|
| typical − random, document budget | $-0.010$ | $-0.017$ |
| typical − random, token-matched | $+0.003$ | $+0.009$ |
| near/far gap, document budget | 0.044 | – |
| near/far gap, length-stratified | 0.016 | – |
| near/far gap, token-matched | 0.002 | – |

$-0.010$ (typical better) to $+0.003$ (typical slightly *worse*) once tokens are matched. In other words, at matched tokens, near-centroid selection provides no benefit over random.

Read together, the three controls form a monotone sequence: as we progressively remove the token-mass confound, first partially, by matching the length distribution, then fully, by matching the token budget: the near/far-centroid gap shrinks from 0.044 to 0.016 to 0.002 nats, i.e. by $\sim$95% overall. The residual after length stratification (0.016) is itself consistent with residual token-mass differences (the stratified arms still differ 1.64B vs. 1.35B tokens), and it too closes once tokens are equalized. We therefore attribute essentially all of the apparent representativeness benefit to retained token mass, of which document length is the driver.

### 4.3 The effect persists at 350M parameters

To test whether the confound is a small-scale artifact, we repeat the key comparison (*typical*/*random*, document-budget and token-matched) at 350M parameters (353M exactly) ($\sim$3.1B tokens/run, 3 seeds). The pattern replicates and, if anything, strengthens (Figure 5): under a document budget *typical* beats *random* by 0.017 nats at 350M (vs. 0.010 at 124M), and token-matching again removes the benefit (*typical* $+0.009$ nats *worse* than random). Thus the confound is not specific to the smallest scale: at both 124M and 350M, the apparent gain appears under a document budget and disappears under a token budget.

### 4.4 SemDeDup, evaluated natively, does not beat random

A natural objection to Table 1 is that *semdedup-LSA* applies SemDeDup's rule in the wrong embedding space. We therefore evaluate SemDeDup natively, embedding the same pool with OPT-

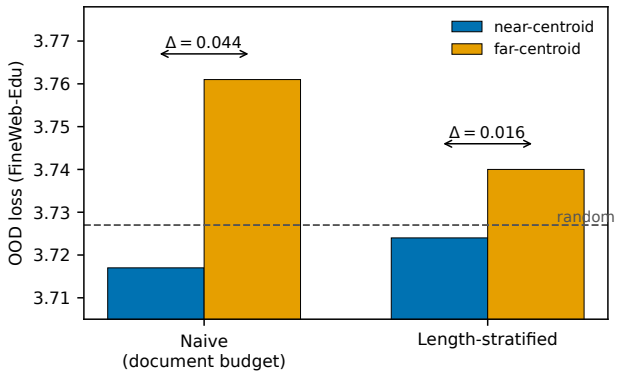


Figure 4: The near/far-centroid gap under length control (124M, OOD loss, 3 seeds). Under a naive document budget the gap is $0.044$ nats; length-stratified selection (which equalizes the length distribution of the two arms) shrinks it to $0.016$; token-matching (not shown here; Table 2) shrinks it further to $0.002$. The dashed line marks random.

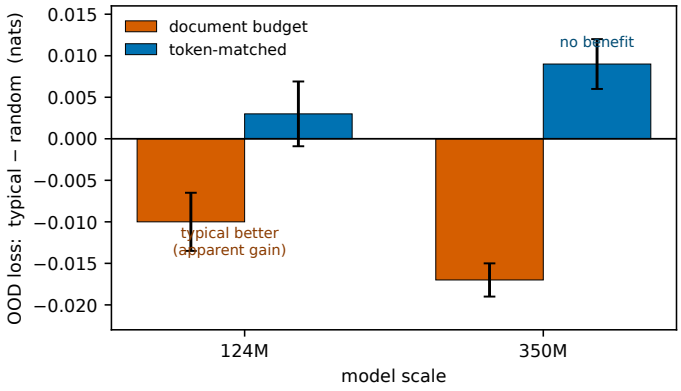


Figure 5: OOD loss difference *typical* − *random* (negative = typical better), under a document budget vs. a matched token budget, at 124M and 350M (mean±std, 3 seeds). The document-budget "gain" appears at both scales and is removed, slightly reversed, once the token budget is matched.

125M (last-layer, last-token) and deduplicating within OPT-space clusters. Native SemDeDup ($3.730 \pm 0.003$) is within noise of random ($3.727 \pm 0.001$; a paired bootstrap over seeds does not separate them), and is markedly better than deduplication in LSA space ($3.759$), so the embedding space matters for the rule, but does not make it beat random here. Its retained subset is nearly length-neutral (1.48B tokens, close to random's 1.49B), so this is *not* the token-mass confound; SemDeDup simply does not help on this curated corpus at this scale, consistent with its reported reliance on redundancy-rich data. We emphasize this is a statement about our corpus and scale, not a verdict on the method: a fair test would use a redundancy-rich corpus, which we leave to future work.

## 5 Discussion

**A cheap, load-bearing confound.** Our central finding is methodological: the apparent benefit of embedding-based selection under the standard document-budget protocol is, at the scales we study, a retained-token-mass effect induced by the correlation between selection score and document length. We recommend four inexpensive controls. First, *report the correlation between the selection score and document length*: a strong correlation (here $-0.413$ to $-0.686$) is an immediate warning that observed gains may be length-mediated. Second, *report retained token counts per arm and prefer a token-matched budget*: a document budget silently varies the token count, and hence data

repetition at fixed compute, across arms, which we find is the actual driver; a token budget removes it. Third, where feasible, *stratify by length* so compared subsets share a length distribution, isolating the selection signal from length composition. Fourth, *compare at matched compute against an explicit random baseline with multiple seeds*, since (below) the effects are small relative to seed variance and single-seed comparisons are unreliable. None of these controls is expensive; their omission changed the sign of our conclusion.

**Prediction does not imply selectability.** A linear lexical projection predicts difficulty better than semantic embeddings (Section 4.1), yet no selection benefit survives the token-matched control. Predictive correlation with a proxy model's loss should not be equated with selection utility.

**Effect sizes.** Differences between selection strategies here are $\sim 10^{-2}$ nats, only a few times seed variance; a single-seed atypicality "win" in our early runs did not survive additional seeds. This both motivates our protocol and bounds the practical stakes of the confound at these scales, though the confound's *mechanism* (length$\rightarrow$tokens$\rightarrow$ repetition) is scale-agnostic, and its magnitude did not shrink from 124M to 350M.

**Limitations.** We study two model scales (124M, 350M) and one main training corpus; larger scales may differ. Our token-matched control equalizes total tokens but arms still differ in length *composition*; the length-stratified control equalizes composition but not exactly token count, together they bracket the confound, but neither is a perfect isolation of geometry. The prediction experiment uses a small proxy model and a different corpus than the selection experiments. We do not evaluate data-mixing methods or a redundancy-rich corpus for SemDeDup.

## 6 Conclusion

Under matched compute, multiple seeds, and a common evaluation, near-centroid data selection appears to beat random—but the gain is a retained-token-mass confound, not embedding geometry. Centroid distance is anti-correlated with document length ($\rho = -0.413$ LSA, $-0.686$ OPT), so representative selection keeps more tokens; matching the token budget removes the benefit, at both 124M and 350M and across two OOD sets, with the near/far-centroid gap collapsing from $0.044$ to $0.002$ nats as the confound is controlled. Native SemDeDup does not beat random on this corpus. We recommend controlling for token mass and length, and reporting against a random baseline at matched compute—simple practices that overturn an apparently positive result.

### AI Use Statement

Generative AI tools were used substantially: for (i) writing and debugging the experiment code (data preparation, embedding, clustering, selection, training configuration, evaluation); and (ii) drafting this manuscript. All AI-assisted artifacts were reviewed and verified by the authors: code was inspected and its behavior validated; all numerical results were produced by the authors' own execution of the code on the hardware described; and all cited references were checked against their originals. The authors take full responsibility for the final content.

### Ethics Statement

This work uses publicly available corpora and models, involves no human subjects or PII, and releases no new models or datasets. Its central finding is cautionary, it identifies a confound that can lead to over-claiming and aims to improve the rigor of selection-method evaluation.

### Reproducibility Statement

All experiments use public corpora (C4, FineWeb-Edu, Wikipedia) and models (OPT-125M, Pythia-160M, GPT-2-scale via nanoGPT). Section 3 and Appendix A specify pool construction, embeddings, clustering, selection rules (document-budget and token-matched), training budgets at both scales, and the evaluation protocol. Code and configurations will be released.

## REFERENCES


Amro Abbas, Kushal Tirumala, Dániel Simig, Surya Ganguli, and Ari S. Morcos. SemDeDup: Data-efficient learning at web-scale through semantic deduplication. *arXiv preprint arXiv:2303.09540*, 2023.

Nora Belrose, Quintin Pope, Lucia Quirke, Alex Mallen, and Xiaoli Fern. Neural networks learn statistics of increasing complexity. In *International Conference on Machine Learning (ICML)*, 2024.

Stella Biderman, Hailey Schoelkopf, Quentin Anthony, Herbie Bradley, Kyle O'Brien, Eric Hallahan, Mohammad Aflah Khan, Shivanshu Purohit, USVSN Sai Prashanth, Edward Raff, Aviya Skowron, Lintang Sutawika, and Oskar van der Wal. Pythia: A suite for analyzing large language models across training and scaling. In *International Conference on Machine Learning (ICML)*, pp. 2397–2430, 2023.

Shizhe Diao, Yu Yang, Yonggan Fu, Xin Dong, Dan Su, Markus Kliegl, Zijia Chen, Peter Belcak, Yoshi Suhara, Hongxu Yin, Mostofa Patwary, Yingyan Celine Lin, Jan Kautz, and Pavlo Molchanov. CLIMB: CLustering-based iterative data mixture bootstrapping for language model pre-training. *arXiv preprint arXiv:2504.13161*, 2025.

Jeff Johnson, Matthijs Douze, and Hervé Jégou. Billion-scale similarity search with GPUs. *IEEE Transactions on Big Data*, 7(3):535–547, 2019.

Arpita Joshi and Nurit Haspel. A novel data instance reduction technique using linear feature reduction. *Journal of Artificial Intelligence and Systems*, 2:191–206, 2020.

Andrej Karpathy. nanoGPT. `https://github.com/karpathy/nanoGPT`, 2022.

Zehan Li, Xin Zhang, Yanzhao Zhang, Dingkun Long, Pengjun Xie, and Meishan Zhang. Towards general text embeddings with multi-stage contrastive learning. *arXiv preprint arXiv:2308.03281*, 2023.

Max Marion, Ahmet Üstün, Luiza Pozzobon, Alex Wang, Marzieh Fadaee, and Sara Hooker. When less is more: Investigating data pruning for pretraining LLMs at scale. *arXiv preprint arXiv:2309.04564*, 2023.

Mansheej Paul, Surya Ganguli, and Gintare Karolina Dziugaite. Deep learning on a data diet: Finding important examples early in training. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2021.

Guilherme Penedo, Hynek Kydlíček, Loubna Ben Allal, Anton Lozhkov, Margaret Mitchell, Colin Raffel, Leandro Von Werra, and Thomas Wolf. The FineWeb datasets: Decanting the web for the finest text data at scale. *Advances in Neural Information Processing Systems (NeurIPS) Datasets and Benchmarks Track*, 2024.

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, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. Exploring the limits of transfer learning with a unified text-to-text transformer. *Journal of Machine Learning Research*, 21(140):1–67, 2020.

Dylan Sam, Ayan Chakrabarti, Afshin Rostamizadeh, Srikumar Ramalingam, Gui Citovsky, and Sanjiv Kumar. Analyzing similarity metrics for data selection for language model pretraining. *arXiv preprint arXiv:2502.02494*, 2025.

Ben Sorscher, Robert Geirhos, Shashank Shekhar, Surya Ganguli, and Ari S. Morcos. Beyond neural scaling laws: beating power law scaling via data pruning. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2022.

Kushal Tirumala, Dániel Simig, Armen Aghajanyan, and Ari S. Morcos. D4: Improving LLM pretraining via document de-duplication and diversification. In *Advances in Neural Information Processing Systems (NeurIPS) Datasets and Benchmarks Track*, 2023.

Wenhui Wang, Furu Wei, Li Dong, Hangbo Bao, Nan Yang, and Ming Zhou. MiniLM: Deep self-attention distillation for task-agnostic compression of pre-trained transformers. *Advances in Neural Information Processing Systems (NeurIPS)*, 33:5776–5788, 2020.

Sang Michael Xie, Hieu Pham, Xuanyi Dong, Nan Du, Hanxiao Liu, Yifeng Lu, Percy Liang, Quoc V. Le, Tengyu Ma, and Adams Wei Yu. DoReMi: Optimizing data mixtures speeds up language model pretraining. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2023a.

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 (NeurIPS)*, 2023b.

Susan Zhang, Stephen Roller, Naman Goyal, Mikel Artetxe, Moya Chen, Shuohui Chen, Christopher Dewan, Mona Diab, Xian Li, Xi Victoria Lin, Todor Mihaylov, Myle Ott, Sam Shleifer, Kurt Shuster, Daniel Simig, Punit Singh Koura, Anjali Sridhar, Tianlu Wang, and Luke Zettlemoyer. OPT: Open pre-trained transformer language models. *arXiv preprint arXiv:2205.01068*, 2022.

## A Training and selection hyperparameters

124M model: 12 layers, 12 heads, width 768, context 1024, no bias/dropout. 350M model: 24 layers, 16 heads, width 1024. Optimization: AdamW ($\beta_1$=0.9, $\beta_2$=0.95), peak LR $6 \times 10^{-4}$ (124M) / $3 \times 10^{-4}$ (350M) with warmup and cosine decay to $0.1\times$, weight decay $0.1$, grad clip $1.0$. Tokens/step: 124M $196{,}608$ ($5000$ steps $\approx 0.98$B tokens); 350M $262{,}144$ ($12000$ steps $\approx 3.1$B tokens). Clustering: FAISS $k$-means, $k$=8000, cosine-normalized. LSA: `TfidfVectorizer` (max_features 30k, sublinear TF, English stopwords, min_df 5) + `TruncatedSVD` (300). Document-budget selection retains $50\%$ of documents per cluster; token-matched selection retains documents in criterion order until $50\%$ of cluster tokens are kept, using GPT-2 tokenizer lengths. Evaluation samples 400 batches of $24 \times 1024$ tokens from the held-out set with a fixed generator, identical across checkpoints.

## B Second out-of-distribution evaluation (Wikipedia)

To check the finding is not specific to FineWeb-Edu, we re-evaluate the same C4-trained (124M) document-budget checkpoints on a held-out English Wikipedia sample ($\sim$62M tokens), a cleaner encyclopedic domain more distant from web-crawled C4. The arm ordering matches Table 1 (Table 3): the long-document arms (*typical*, *long*) are best and tied, the short-document arms (*atypical*, *short*) worst, and *random*/native SemDeDup in between.

Table 3: Second OOD evaluation on English Wikipedia (loss, mean$\pm$std over 3 seeds; same 124M document-budget checkpoints as Table 1).

| Arm | Wikipedia OOD loss |
|---|---|
| *typical* | $4.020 \pm 0.014$ |
| *long* | $4.024 \pm 0.005$ |
| *semdedup-OPT* | $4.040 \pm 0.012$ |
| *random* | $4.040 \pm 0.001$ |
| *atypical* | $4.076 \pm 0.011$ |
| *short* | $4.126 \pm 0.004$ |