

AttentionInfluence: Adopting Attention Head Influence for Weak-to-Strong Pretraining Data Selection

Kai Hua[†], Steven Wu, Ge Zhang, Ke Shen[†]

ByteDance Seed

[†]Corresponding authors

Abstract

Recently, there has been growing interest in collecting reasoning-intensive pretraining data to improve LLMs' complex reasoning ability. Prior approaches typically rely on supervised classifiers to identify such data, which requires labeling by humans or LLMs, often introducing domain-specific biases. Due to the attention heads being crucial to in-context reasoning, we propose **AttentionInfluence**, a simple yet effective, **training-free** method **without supervision signal**. Our approach enables a **small pretrained language model** to act as a strong data selector through a simple attention head masking operation. Specifically, we identify retrieval heads and compute the loss difference when masking these heads. We apply AttentionInfluence to a 1.3B-parameter dense model to conduct data selection on the SmoLM corpus of 241B tokens, and mix the SmoLM corpus with the selected subset comprising 73B tokens to pretrain a 7B-parameter dense model using 1T training tokens and WSD learning rate scheduling. Our experimental results demonstrate substantial improvements, ranging from **1.4pp** to **3.5pp**, across several knowledge-intensive and reasoning-heavy benchmarks (i.e., MMLU, MMLU-Pro, AGIEval-en, GSM8K, and HumanEval). This demonstrates an effective **weak-to-strong** scaling property, with small models improving the final performance of larger models—offering a promising and scalable path for reasoning-centric data selection.

Date: May 13, 2025

Correspondence: Kai Hua at huakai.dev@bytedance.com, Ke Shen at shenke@bytedance.com

1 Introduction

The identification of high-quality pretraining data has been a key factor enabling Large Language Models' (**LLMs**) creation. Commonly recognized high-quality pretraining materials include academic papers (e.g., arXiv), books (e.g., Project Gutenberg), high-quality code (e.g., GitHub), and instruction datasets [25]. Existing approaches often rely on manually curated high-quality seed data to train classifiers for extracting additional high-quality pretraining data from massive web corpora. However, as the size and diversity of LLMs' pretraining data requirements continue to grow, these carefully curated classifiers suffer from the high manual effort requirements and relatively low diversity of identified data. This raises a critical research question: *How can we continue to identify diverse high-quality pretraining data efficiently and scalably?*

Current mainstream methods[40] typically use supervised or weakly supervised data to train classifiers to identify high-quality data. For instance, LLaMA2[43] uses reference information of Wikipedia documents, which can be seen as weakly supervised data to train a fasttext[21] classifier and then recognize Wikipedia-

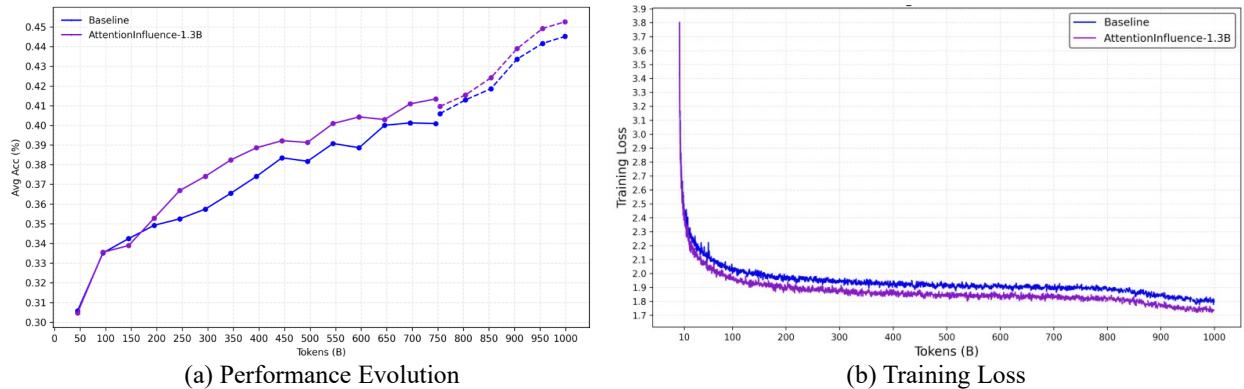


Figure 1 (a) Performance evolution on comprehensive benchmark evaluations during pretraining. The first 750 billion tokens correspond to the pretraining phase, represented by solid lines, while the subsequent 250 billion tokens represent the learning rate annealing phase, represented by dashed lines, using the same dataset. After around 100 billion tokens, AttentionInfluence-1.3B consistently outperforms the baseline across a wide range of tasks on average, including the annealing phase. **(b) Training Loss during pretraining.** AttentionInfluence-1.3B consistently achieves a lower loss than the baseline.

like documents. LLaMA3[13] and FineWeb-Edu[32] use LLM-generated responses to train a classifier for educational value, which can be regarded as a much sparser form of distillation from a larger LLM(up to 70B dense parameters) than knowledge distillation[17]. While other approaches like DCLM aim to fit user preferences through utilizing signals of user behavior, these methods may introduce potential bias and do harm to diversity[25]. There are also efforts to train several domain classifiers and combine them for practical use [46]. However, we assume that these methods fail to capture the essence of what makes data reasoning-intensive, and as a result, they can be labor-intensive and require significant data engineering efforts. Moreover, there exists a risk that the classification results from small models distilled from larger models’ responses may not improve the final performance of larger models.

Therefore, we propose **AttentionInfluence**, which leverages the intrinsic mechanism of existing LLMs’ attention heads for pretraining data selection to achieve weak-to-strong generalization. Existing research suggests that feedforward networks (FFNs) store atomic knowledge[12], while attention mechanisms execute algorithms and store procedural knowledge[31, 47]. These mechanistic interpretability insights inspire us to hypothesize that the data activating more important attention heads are high-quality and about procedural knowledge. To be specific, we select the data with a relatively larger loss difference when small pretrained language models process them with and without masking retrieval heads. Compared with mainstream data selection methods [21, 25], our method is training-free and more generalizable.

To validate AttentionInfluence, we adopt a LLaMA2-alike-1.3B pretrained checkpoint for data selection from SmoLM-Corpus. We then pretrain a 7B dense language model—SmoLM-7B—as our baseline on the SmoLM-Corpus, a 241B-token curated dataset that already applies strong quality filtering with an education-focused classifier (FineWeb-Edu-Dedup) to retain high-quality data. As shown in (a) of Figure 1, despite this strong baseline, AttentionInfluence still yields consistent improvements, demonstrating its ability to further improve the overall data quality through better data selection beyond existing heuristics or classifiers. AttentionInfluence shows consistent improvement against SmoLM-7B across a wide range of tasks, demonstrating the effectiveness of the selected data. We further compare AttentionInfluence with a trained classifier (FineWeb-Edu Classifier) and find that it selects data that is more balanced and broadly distributed across content categories, and preferentially favors longer and more comprehensive samples. Despite being entirely supervision-free and training-free, AttentionInfluence also shows strong agreement with classifier-based patterns, validating its reliability and generalizability.

In summary, our key contributions are as follows:

1. We propose **AttentionInfluence**, a novel framework that leverages intrinsic model behaviors—specifically

attention head mechanisms—for **effective data selection without any supervision signal**.

2. We show that data selected by AttentionInfluence is **high-quality and well-distributed**, yielding consistent improvements in downstream training.
3. We demonstrate that this approach exhibits **“weak-to-strong” scaling property**, where data selected by smaller models significantly improves the training of larger models, resulting in performance gains without relying on human-labeled data, LLM-generated data, or training any classifiers.

2 Related Work

2.1 Data Selection

Many works use heuristic filtering rules[35] or perplexity[2] of existing LLMs to assess the quality of pretraining data, which makes them training-free. Scaling filter[25] uses the perplexity difference between two LLMs trained on the same data to evaluate text quality. However, when two LLMs trained on the same data are unavailable, training LLMs incurs substantial computational cost. In contrast, methods that rely on training a model with high-quality labeled data gain more attention owing to their higher accuracy and superior versatility across different data categories. For instance, LLaMA2 uses reference information from Wikipedia documents, which can be seen as weak supervision data to train a fasttext[21] classifier and then recognize Wikipedia-like documents. LLaMA3[13] and FineWeb-Edu[32] use LLM-generated responses to train a classifier for educational value, which can be regarded as a much sparser form of distillation from a larger LLM (up to 70B dense parameters) than knowledge distillation[17]. While other approaches like DCLM[24] aim to fit user preferences by utilizing user behavior signals. Some recent and more advanced approaches[33, 45, 52] instead train multi-class classifiers to make fine-grained distinctions among various content types and depend on labeled data generated by proprietary commercial large language models such as GPT-3.5-turbo, GPT-4, and GPT-4o. There are also efforts to train several domain classifiers and combine them for practical use [46]. AttentionInfluence can be seen as a **training-free** method and is available at minimal cost without any training cost or human-labeled or LLM-labeled data.

2.2 Data Mixture

There are also efforts to optimize the data mixture by either relying on human selection or using automatic frameworks. Ye et al. [49] propose Data Mixing Laws by introducing a nested prediction framework that combines scaling laws of training steps, model sizes, and data mixtures, enabling efficient optimization of large-scale pretraining data using only small-scale experiments. Xie et al. [48] propose DoReMi (Domain Reweighting with Minimax Optimization), a method that uses a small proxy model and distributionally robust optimization to automatically learn optimal domain mixture weights for language model pretraining. Held et al. [14] conduct data mixing by designing a lightweight method that leverages LLMs to estimate data utility from small samples, enabling compute-efficient optimization with comparable performance to ablation-based approaches. OLMo et al. [30] introduce OLMo and uses the existing data mixture designed for a different model family. REGMIX [27] is a regression-based framework for optimizing data mixtures in language model pretraining by training small proxy models on diverse mixtures and predicting performance using regression.

2.3 Mechanistic Interpretability

Understanding the inner workings of LLMs is crucial for advancing artificial general intelligence safely. Consequently, studies in mechanistic interpretability [5, 11, 28, 31, 47, 53] are increasingly being conducted. Olsson et al. [31] primarily investigates the relationship between certain heads in large language models (LLMs) and their in-context learning capabilities. Bricken et al. [5] extracts a large number of interpretable features from a one-layer transformer with a sparse autoencoder in order to analyze the behavior of the neural network. Lv et al. [28] investigates mechanisms for factual recall in Transformer-based LLMs, focusing on attention head extraction, MLP activation, and the collaborative mechanism between attention heads and MLPs. Wu et al. [47] identifies a class of attention heads, termed retrieval heads, which retrieve relevant information in LLMs. These heads exhibit universal, sparse, and dynamically activated behavior, and they play a crucial role in enabling chain-of-thought reasoning. Fu et al. [11] aims to estimate the importance

of different attention heads for contextual QA tasks that require both retrieval and reasoning capabilities, enabling efficient head-level KV cache compression for language model inference. Inspired by the findings of Qiu et al. [34], Wu et al. [47], AttentionInfluence adopts a similar and simple proxy task to detect specific important heads, namely the retrieval heads in this paper. AttentionInfluence naturally extends the insights from Wu et al. [47], broadening their application beyond model analysis and inference acceleration to include effective and efficient data selection.

2.4 Influence Measure

Ruis et al. [38] uses influence functions to recognize pretraining documents important for learning factual knowledge and mathematical reasoning separately. Mirror Influence[22] realizes an efficient data influence estimation to select high-quality data. MATES[50] continuously adapts a data influence model to the evolving data preferences of the pretraining model and then selects the most effective data for the current pretraining progress. Our work is similar to Mirror Influence in that we use data influence estimation to select high-quality data. However, while Mirror Influence requires a high-quality dataset to train a strong reference model and create a model pair with significant differences in capabilities to compute delta loss, our approach uses the attention mechanism to derive a weaker reference model from the base model. This enables us to obtain two models with a significant capability gap and compute delta loss to evaluate data quality.

3 Preliminary

To estimate the impact of each pretraining data sample on LLMs' intrinsic reasoning and retrieval capabilities, we adapt the retrieval score defined in [47] and model it as a token-level recall rate based on the attention head behavior. We denote the current token being generated as w at the decoding step t while decoding the LLM. We further denote the attention scores of a head as $\mathbf{a} \in \mathcal{R}^{|\mathbf{x}|}$ where x represents the vocabulary. Consequently, $|\mathbf{x}|$ denotes the size of the vocabulary. We assume that an attention head h performs a copy-paste operation for corresponding content \mathbf{k} , if and only if the following two conditions are met:

Condition 1: The generated token w appears in the corresponding content k :

$$w \in \mathbf{k} \quad (1)$$

Condition 2: The token w receives the highest attention score among all positions visible to the current query token in this head:

$$j \in \mathbf{i}_q, \text{ where } \mathbf{i}_q = \{j \mid j < t\} \text{ is the set of positions visible at decoding step } t \quad (2)$$

$$j = \arg \max(\mathbf{a}), \mathbf{x}_j = w \quad (3)$$

Let \mathbf{g}_h denote the set containing all tokens copied and pasted by a given head h , we define:

$$\text{Retrieval score for head } h = \frac{|\mathbf{g}_h \cap \mathbf{k}|}{|\mathbf{k}|} \quad (4)$$

4 Method

Lin et al. [26] demonstrate that a well-trained reference model can serve as a proxy to fit the desired data distribution of the LLM pretraining by comparing the data loss gap between the base model and the reference model. By comparing the token-level data loss gap between the base model and the reference model, they can identify important tokens that align better with the target distribution. Inspired by recent work Ko et al. [22], Lin et al. [26], we propose **AttentionInfluence** to select high-quality pretraining data based on the data loss gap from a *<weak model, strong model>* pair. However, while existing approaches [22, 26] focus on building a stronger reference model as the *strong model*, AttentionInfluence points out that it is cheaper and more controllable to degrade the base model to a weaker version, thus constructing a *<weak model, strong model>* pair.

Existing studies [31, 47] point out that specific attention heads (i.e., **retrieval heads**) plays a critical role in LLMs’ in-context learning, retrieval, and reasoning capabilities. We find that the language model’s retrieval heads emerge early in training, gradually strengthen, and eventually become entrenched in the middle to late stages, playing a crucial role in the model’s performance, as shown in Figure 2. Inspired by this insight, AttentionInfluence identifies the specific attention heads that are important for targeted LLM capabilities and obtains a degraded reference model by disabling them. Then, AttentionInfluence selects high-quality pretraining data based on the sample-level data loss gap from the constructed \langle weak model, strong model \rangle pair.

We detail the AttentionInfluence method in the following section.

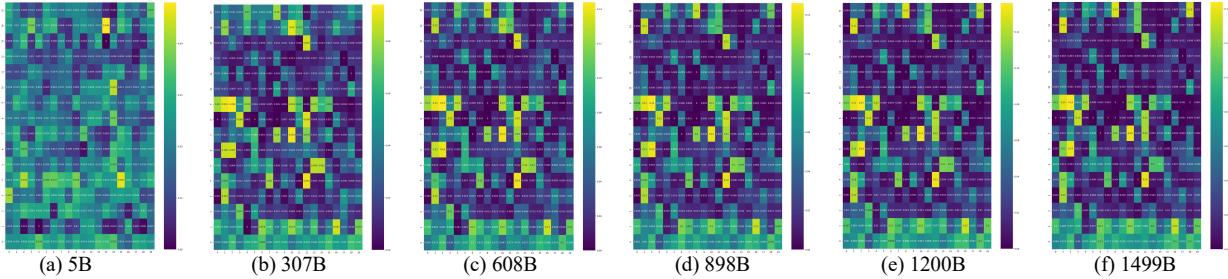


Figure 2 The evolution of retrieval heads in a 1.3B dense model.

4.1 Detecting Specific Important Heads

In this work, we detect the retrieval heads as specifically important heads for reasoning, because Wu et al. [47] reveals that retrieval heads are extremely relevant to LLMs’ retrieval and reasoning ability.

Inspired by the Key-Passage Retrieval evaluation task proposed in CLongEval[34], we adopt a similar and simple proxy task to evaluate the retrieval ability of LLMs in a controlled setting, and identify attention heads that are strongly associated with retrieval and reasoning. To this end, we construct a synthetic test dataset consisting of 800 samples. Each sample is formatted as a 3-shot retrieval task in natural language, consisting of a `context`, three in-context demonstrations, and a query `hash_key`. The sample template is detailed in Appendix A. Each `context` is a JSON object with k key-value pairs, where each key is a randomly generated 32-character alphanumeric string (`hash_key`), and each value (`text_val`)¹ is a natural language sentence sampled from a corpus of web documents. The task requires the model to retrieve the `text_val` from the `context` and output the `text_val` corresponding to the given query `hash_key`. The inclusion of three in-context demonstrations (i.e., 3-shot) is designed to simulate a few-shot learning scenario and help the model understand the task. Considering the context length limitation of existing pretrained models, we constrain the total length of each test sample—including both the input prompt and the answer—to be close to but not exceeding 4,096 tokens.

Next, we compute retrieval scores for each attention head across test samples, as described in Section 3. In this work, we use a 1.3B-parameter model based on the LLaMA2-alike architecture as the small pretrained language model. We use the average score as the head’s final retrieval score and sort them by it. Referring to Wu et al. [47], we select the heads ranked in the top 5% as specifically important heads.

4.2 Calculating AttentionInfluence Score

We obtain a reference model by masking the important heads of the base model detected in the first phase, and compute the AttentionInfluence score based on the base model and reference model. For details on the masking operation, refer to Appendix C. First, we use the base model to compute the mean token-level cross-entropy loss ($\mathcal{L}_{\text{base}}$) of each sample in the corpus. Subsequently, we compute the corresponding loss (\mathcal{L}_{ref}) using the reference model. Finally, we use the relative delta between $\mathcal{L}_{\text{base}}$ and \mathcal{L}_{ref} as an AttentionInfluence

¹Each `text_val` is capped at a maximum of 30 tokens.

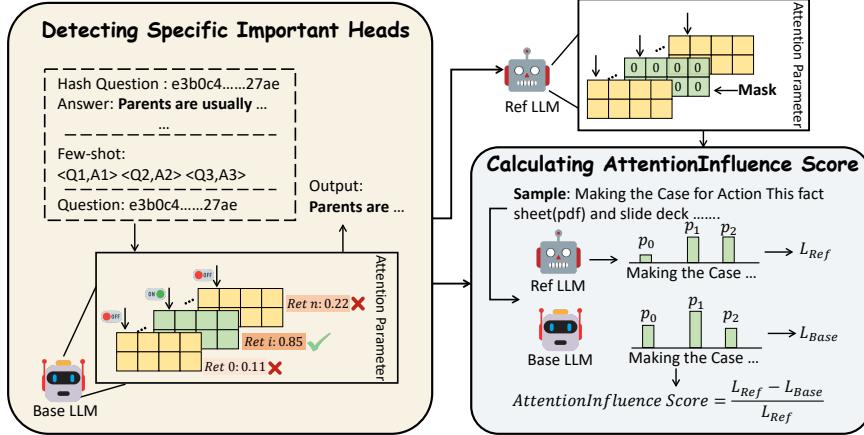


Figure 3 The illustration of AttentionInfluence.

Score to quantify the reasoning intensity of each sample, which can be denoted as:

$$\text{AttentionInfluence Score} = \frac{\mathcal{L}_{\text{ref}} - \mathcal{L}_{\text{base}}}{\mathcal{L}_{\text{base}}} \quad (5)$$

Since the loss of a language model for data from different domains (e.g., general/math/code) cannot be directly compared due to significant distribution differences, we restrict the AttentionInfluence Score to be compared only within the same domain (e.g., general/math/code). We consider that a higher AttentionInfluence Score indicates a higher reasoning intensity of the sample.

5 Experiments and Results

In this section, we present experimental analyses to validate the effectiveness of reasoning-intensive data selected by AttentionInfluence.

5.1 Experimental Details

We apply AttentionInfluence to a **LLaMA2-alike-1.3B** pretrained model to rank the SmolLM-Corpus dataset² [3]. The specifications of the model are described in [Appendix E](#). Specifically, we select the top 20% of samples based on the AttentionInfluence score, yielding approximately **73.1B** reasoning-intensive tokens for pretraining.

To evaluate the effectiveness of AttentionInfluence, we pretrain a **7B** dense model using a combination of the full SmolLM-Corpus and the selected 73.1B tokens. For comparison, we pretrain another model of identical architecture and size using only the SmolLM-Corpus, serving as the baseline. The model architecture follows that of LLaMA2, and detailed hyperparameters are listed in [Table 9](#). Further information on the SmolLM-Corpus dataset and pretraining configurations is provided in [Appendix E](#).

Following Grattafiori et al. [13], we adopt a comprehensive set of benchmark evaluations across **four** major categories under the few-shot setting to holistically compare our model with the baseline: **1) Aggregate Benchmarks**, including AGIEval-en [54], MMLU [15], MMLU-Pro [44], GPQA [37], and C-Eval [19]; **2) Mathematics, Code, and Reasoning**, comprising GSM8K [9], MATH [16], HumanEval [7], ARC Challenge [8], DROP [10], and BBH [41]; **3) Commonsense Reasoning and Understanding**, including HellaSwag [51], ARC-Easy [8], WinoGrande [39], CommonSenseQA [42], PiQA [4], OpenBookQA [29], and TriviaQA [20]; and **4) Reading Comprehension**, represented by RACE [23]. Details of the evaluation setup are provided in [Appendix E](#).

²<https://github.com/huggingface/smollm/tree/main/text/pretraining>

Model	#Tokens	Avg.	Metrics							
			ARC-C	ARC-E	ARC(C+E)	Wino.	Hella.	CSQA	OpenBookQA	
SmolLM-1.7B	1T	-	-	-	59.95	54.70	62.83	38.00	42.60	
			PIQA	TriviaQA	MMLU	MMLU-Pro	AGIEval-en	RACE	DROP	
			75.90	13.14	39.35	10.92	-	-	-	
			BBH	GSM8K	MATH	HumanEval	C-Eval	GPQA		
			-	4.62	-	-	-	-	-	
Baseline w/o LRD	746B	40.09	ARC-C	ARC-E	ARC(C+E)	Wino.	Hella.	CSQA	OpenBookQA	
			55.89	81.69	68.79	66.22	71.79	49.14	45.40	
			PIQA	TriviaQA	MMLU	MMLU-Pro	AGIEval-en	RACE	DROP	
			79.27	45.57	41.76	13.80	21.10	40.67	31.71	
			BBH	GSM8K	MATH	HumanEval	C-Eval	GPQA		
Ours w/o LRD	746B	41.34	ARC-C	ARC-E	ARC(C+E)	Wino.	Hella.	CSQA	OpenBookQA	
			56.66	82.03	69.35	65.43	71.90	53.48	43.60	
			PIQA	TriviaQA	MMLU	MMLU-Pro	AGIEval-en	RACE	DROP	
			77.58	45.68	45.10	17.19	22.50	41.72	32.03	
			BBH	GSM8K	MATH	HumanEval	C-Eval	GPQA		
Baseline w/ LRD	1T	44.51	ARC-C	ARC-E	ARC(C+E)	Wino.	Hella.	CSQA	OpenBookQA	
			58.79	83.92	71.36	70.24	75.63	59.62	48.00	
			PIQA	TriviaQA	MMLU	MMLU-Pro	AGIEval-en	RACE	DROP	
			80.63	51.07	50.05	19.36	24.50	41.15	36.09	
			BBH	GSM8K	MATH	HumanEval	C-Eval	GPQA		
Ours w/ LRD	1T	45.26	ARC-C	ARC-E	ARC(C+E)	Wino.	Hella.	CSQA	OpenBookQA	
			59.98	84.26	72.12	68.03	75.49	61.59	46.60	
			PIQA	TriviaQA	MMLU	MMLU-Pro	AGIEval-en	RACE	DROP	
			79.54	51.20	51.48	22.03	26.30	42.30	36.52	
			BBH	GSM8K	MATH	HumanEval	C-Eval	GPQA		
			36.22	23.73	10.80	26.55	33.06	24.26		

Table 1 The main results on various benchmarks. The LRD denotes learning rate decay.

5.2 Results

1) Aggregate Benchmarks: On challenging aggregate benchmarks such as MMLU, MMLU-Pro, and AGIEval-en, AttentionInfluence consistently outperforms the baseline, indicating stronger general knowledge and reasoning capabilities. These improvements—**+1.4pp** on MMLU, **+2.7pp** on MMLU-Pro, and **+1.8pp** on AGIEval-en—underscore the effectiveness of AttentionInfluence in identifying a diverse distribution of pretraining data that supports both **comprehensive knowledge acquisition** and **reasoning-intensive learning**.

2) Math, Code, and Reasoning: AttentionInfluence yields substantial improvements on complex multi-step reasoning tasks such as GSM8K (**+2.7pp**), HumanEval (**+3.5pp**), and BBH (**+0.9pp**), suggesting that the selected data distribution better facilitates problem-solving and advanced reasoning. Additional gains on ARC-Challenge, DROP, and MATH further demonstrate that **AttentionInfluence enhances reasoning generalization across a wide range of tasks**.

3) Commonsense Reasoning and Understanding: On benchmarks including CSQA, PiQA, and OpenBookQA, AttentionInfluence achieves competitive or superior results. Notably, its performance on ARC-Easy and TriviaQA indicates that AttentionInfluence maintains strong results on factual and commonsense tasks, despite being primarily designed to select reasoning-intensive data.

4) Reading Comprehension: On RACE, AttentionInfluence surpasses the baseline by **+1.2pp**, reflecting enhanced discourse-level understanding and reasoning capabilities.

Performance Evolution During Pretraining: We evaluate the training dynamics of AttentionInfluence-1.3B on the specified tasks. As shown in [Figure 1](#), our method consistently outperforms the baseline throughout the pretraining process. Full results for all evaluation tasks are provided in [Figure 9](#) and [Figure 10](#). The performance gap emerges early—well before reaching 100B tokens—and remains stable over time. After approximately 100 billion tokens, AttentionInfluence-1.3B demonstrates a clear and consistent advantage over the baseline, on average, across multiple tasks. These improvements persist throughout all training phases, including both before and after the learning rate decay (LRD). Although the performance margin slightly narrows

following the LRD, this effect primarily results from saturation as training approaches 1T tokens on the 7B model, which is trained on the SmoLLM-Corpus containing only 241B tokens. Nonetheless, AttentionInfluence maintains a stable advantage without requiring any additional supervision signals. Moreover, on benchmarks that primarily require simple factual knowledge, the performance of LLMs trained with AttentionInfluence is comparable to that of the baseline (see [Figure 9](#)). In contrast, on reasoning-intensive benchmarks, our models achieve significantly better results than the baseline (see [Figure 10](#)). These findings demonstrate that AttentionInfluence is effective at selecting data with higher reasoning intensity.

Mirror Effects in AttentionInfluence: For tasks with performance gains—such as MMLU, MMLU-Pro, AGIEval-en, DROP, BBH and GSM8K—we observe that masking retrieval heads in the pretrained 1.3B model leads to a significant performance drop (see [Appendix D](#) for details). This suggests a mirror effect: when the performance of the 1.3B model significantly degrades on certain tasks due to masking the specific important heads, the data selected by AttentionInfluence-1.3B tends to improve performance on these tasks when used to train a 7B model. This observation supports the insight discussed in [subsection 4.1](#), demonstrating the interpretability of AttentionInfluence and its predictive power in identifying evaluation metrics likely to show improvement before any training.

Increasing Parameter Size of AttentionInfluence: Furthermore, as shown in [Table 2](#), LLMs pretrained on data selected by AttentionInfluence-7B exhibit superior performance on these challenging knowledge-intensive and reasoning-intensive benchmarks. This indicates that increasing the model size in AttentionInfluence enables the selection of samples with higher reasoning intensity. The comparison details between the 1.3B and 7B methods are provided in [Table 10](#) of [Appendix F](#).

In conclusion: These results validate that **AttentionInfluence effectively identifies high-quality pretraining data that enhances the knowledge and reasoning capabilities of LLMs**, yielding particularly notable gains on benchmarks requiring comprehensive knowledge and complex reasoning, as shown in [Table 1](#). Furthermore, AttentionInfluence can be combined with the FineWeb-Edu Classifier to achieve comprehensive improvements in LLM performance on tasks that require either simple factual knowledge, advanced reasoning, or both.

Model	MMLU	GPQA	MATH	C-Eval	AGIEval-en	BBH
1.3B	51.48	24.26	10.80	33.06	26.30	36.22
7B	53.18	24.87	11.75	36.85	26.85	36.80

Table 2 Comparison of results trained with AttentionInfluence-1.3B/7B on relatively difficult benchmarks.

6 Discussion

6.1 Reliability of AttentionInfluence

Domain	FineWeb-Edu Classifier			AttentionInfluence		
	Edu Score	Reasoning Score	Token Len	Edu Score	Reasoning Score	Token Len
FineWeb-Edu-Dedup	0.99	0.52	1610.12	0.99	0.49	1629.73
Cosmopedia-V2	1.0	0.87	825.46	1.0	0.80	893.805
Python-Edu	0.98	0.76	414.15	0.98	0.87	820.71
OpenWebMath	0.99	0.52	1022.855	0.96	0.88	2255.575

Table 3 The quality score of the data selected by AttentionInfluence and FineWeb-Edu Classifier.

To validate the effectiveness of AttentionInfluence, we design two metrics—**Education Score** and **Reasoning Score**—to quantify the quality of the selected data. Specifically, we randomly sample 200 examples from the top 20% ranked by AttentionInfluence and the FineWeb-Edu classifier, respectively, and employ GPT-4o as the evaluator. The detailed scoring criteria and prompt design for both metrics are provided in [Appendix G](#).

As shown in [Table 3](#), both AttentionInfluence and FineWeb-Edu classifier yield comparable scores on education-related content. However, AttentionInfluence achieves substantially higher scores in reasoning, indicating that **the samples selected by AttentionInfluence exhibit greater reasoning intensity**.

Additionally, we analyze the length of the selected samples. In the Python-Edu and OpenWebMath domains, AttentionInfluence selects samples with an average length nearly twice that of those selected by the FineWeb-Edu classifier. A qualitative inspection of these samples (see [Appendix I](#)) reveals that, in the Python-Edu domain, AttentionInfluence prefers documents containing not only more complex code but also richer textual context, such as detailed problem statements. In the OpenWebMath domain, the selected samples demonstrate more elaborate formula-based reasoning. **These findings suggest that AttentionInfluence effectively identifies data with more comprehensive and complex reasoning structures.**

6.2 Diversity of Selected Data by AttentionInfluence

6.2.1 Word Frequency Analysis

Ranking (%)	Static Method	Data Source			
		FineWeb-Edu-Dedup	Cosmopedia-v2	Python-Edu	OpenWebMath
10	TF	0.84	0.73	0.29	0.57
	TF-IDF	0.82	0.72	0.38	0.52
20	TF	0.88	0.81	0.41	0.67
	TF-IDF	0.87	0.80	0.43	0.63
50	TF	0.95	0.91	0.67	0.79
	TF-IDF	0.92	0.90	0.66	0.78

Table 4 Word overlap by ranking threshold and frequency-based statistical method

We separately select the top 10%, 20%, and 50% of samples ranked by AttentionInfluence and the FineWeb-Edu classifier, and compute the overlap of high-frequency words using multiple statistical approaches.

As shown in [Table 4](#), we derive several key insights: 1) AttentionInfluence exhibits a high degree of overlap with the FineWeb-Edu Classifier, highlighting the **reliability of the samples selected by AttentionInfluence**. 2) **AttentionInfluence and the FineWeb-Edu Classifier demonstrate a degree of complementarity**. We observe notable domain-specific variations. Specifically, in the FineWeb-Edu-Dedup and Cosmopedia-v2 domains, the overlap exceeds 70%, whereas in the Python-Edu and OpenWebMath domains, it falls below 60%. To further examine the differences between AttentionInfluence and FineWeb-Edu Classifier in specific domains, we sample representative examples from the Python-Edu and OpenWebMath domains, as shown in [Appendix I](#). These cases reveal that although the two methods display different preferences across domains, both yield reasonable selections."

As shown in [Table 11 of Appendix L](#), **AttentionInfluence places greater emphasis on method-related terminology, while FineWeb-Edu Classifier is more sensitive to numerical expressions**. We identify two distinctive high-frequency terms: "19th" from subset selected by FineWeb-Edu Classifier and "sklearn" from AttentionInfluence's subset. We then retrieve representative documents from the original corpus containing these terms. The sample containing "19th" is related to historical topics, whereas the one with "sklearn" discusses K-Nearest Neighbors Classifier and Hyperparameter Tuning. **This suggests that AttentionInfluence prefers samples containing hands-on coding or procedural mathematical reasoning.**

6.2.2 Clustering-Based Distribution Analysis

To better understand the distribution of samples selected by different methods (i.e., AttentionInfluence and the FineWeb-Edu Classifier), we cluster the selected subsets and employ GPT-4o to annotate the resulting clusters. The clustering procedure is detailed in [Appendix H](#).

We derive the following insights:

- 1) **AttentionInfluence produces a more balanced distribution across data categories.** As illustrated in [Figure 4](#), both methods achieve broad coverage of the top-level categories. However, the distribution resulting from AttentionInfluence is noticeably more balanced.
- 2) **AttentionInfluence selects a highly diverse set of samples.** We examine two clusters from the AttentionInfluence subset that exhibit large embedding distances. As demonstrated by the examples from the

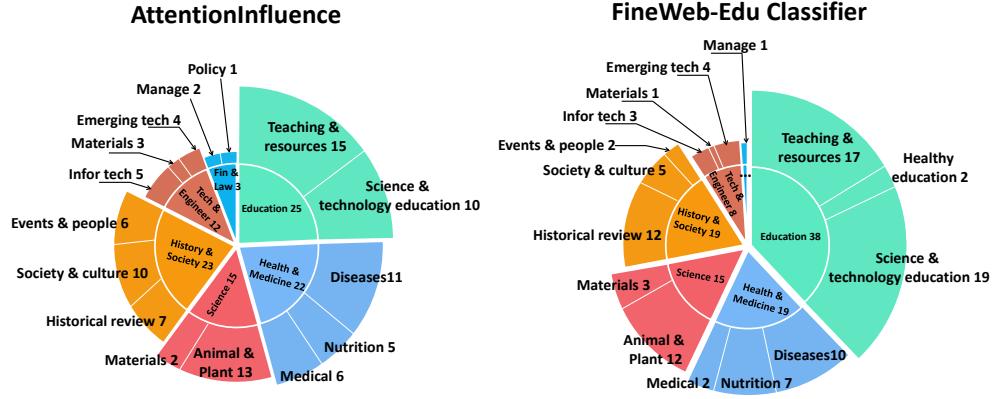


Figure 4 The statistics of clustering. The left is the clustering result of AttentionInfluence, the right part is that of FineWeb-Edu Classifier.

Health Guidelines & Nutrition and *Information Technology* clusters in Appendix J, the selected samples differ substantially in both content and style. This lack of semantic overlap underscores the effectiveness of the clustering and enhances the interpretability of the annotated categories.

6.2.3 The Visualization of Data Distribution

To provide an intuitive illustration of the relationship between the two selection methods, we apply Principal Component Analysis (PCA) to reduce the dimensionality of the document embeddings and visualize their distributions in two-dimensional space.

As shown in Figure 5, AttentionInfluence selects samples with broader and more balanced coverage. **By directly leveraging the attention mechanisms of pretrained language models, it facilitates more effective selection of general and diverse training data than the FineWeb-Edu classifier.**

In addition, the selected samples from the two methods exhibit complementary coverage. We further examine the distinctive regions identified by AttentionInfluence and FineWeb-Edu Classifier. For example, the samples in Zone1 are related to Health Education, while most samples in Zone2 fall under the theme of Emerging Technologies. This suggests that the samples selected by the two methods can be complementary. How to effectively integrate the strengths of both selection strategies could be a promising direction for future exploration.

6.3 Scalability of AttentionInfluence

We compare the samples selected by the AttentionInfluence method using 1.3B and 7B pretrained language models. We obtain the following insights:

AttentionInfluence based on a larger LLM selects higher quality data. Similar to the setting in the section 6.1, we use GPT4o to evaluate selected samples. As shown in Table 5, across all domains, the samples selected by the 7B model exhibit high edu scores that are comparable to those selected by the 1.3B model, with a slight overall advantage. Regarding reasoning scores, the 7B model consistently outperforms the 1.3B model across all four domains, achieving a particularly notable improvement of 9% in the FineWeb-Edu-Dedup domain.

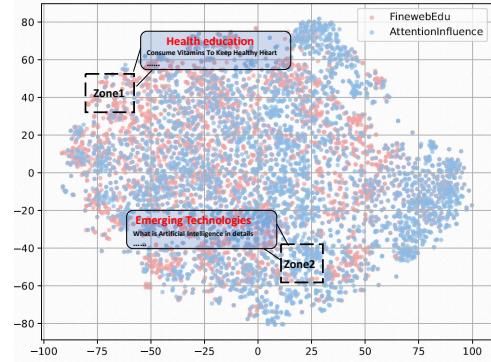


Figure 5 Visualization of data selected by AttentionInfluence and FineWeb-Edu Classifier.

Domain	1.3B			7B		
	Edu Score	Reasoning Score	Token Len	Edu Score	Reasoning Score	Token Len
FineWeb-Edu-Dedup	0.99	0.49	1895.7	0.97	0.58	3488.8
Cosmopedia-V2	1.0	0.80	2774.6	1.0	0.82	2984.1
Python-Edu	0.97	0.87	909.3	0.98	0.91	1657.2
OpenWebMath	0.96	0.88	2138.6	0.96	0.93	5550.4

Table 5 The quality score of the data selected by AttentionInfluence using 1.3B and 7B models, respectively.

These results suggest that larger models are more effective at identifying reasoning-intensive samples.

AttentionInfluence based on a larger LLM is more generalizable

AttentionInfluence based on a larger LLM is more generalizable. As shown in the Figure 6, we compare the sample distributions selected by the 1.3B and 7B models. We observe that the samples selected by the 7B model are more widely distributed in the space, covering many areas that the 1.3B model fails to reach. Notably, regions underrepresented by the 1.3B model are densely populated with specific categories of samples, which are predominantly captured by the 7B model. For instance, Zone1 corresponds to cooking, Zone2 is related to code, and Zone3 mainly focuses on the economy. This suggests that even without additional training, the samples selected by larger models are more balanced and diverse, capturing a broader range of information. Moreover, we also trained a 7B model on the data selected by AttentionInfluence-7B. As shown in the appendix (see Table 10 and Figure 7), this model achieves better performance than AttentionInfluence-1.3B in the middle and later stages of training, where the average accuracy excludes the result of the C-Eval[19] evaluation task. However, the gap narrows during the final learning rate annealing phase, which is likely due to saturation in the SmoLLM corpus and training setup referring to the comparisons with SmoLLM[3] and SmoLLM2[1]. Importantly, the selected evaluation benchmarks may not fully capture the generalization benefits of AttentionInfluence-7B. For example, while the SmoLLM corpus is predominantly English with minimal Chinese contents, we observe that AttentionInfluence-7B significantly outperforms AttentionInfluence-1.3B on the Chinese C-Eval benchmark which is shown in Figure 10, reflecting a broader and more robust generalization ability that remains underexplored under the current evaluation settings.

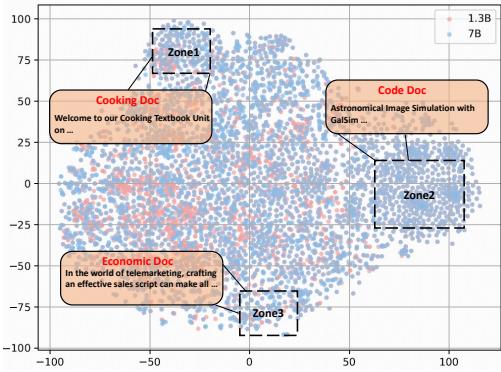


Figure 6 The visualization of the samples selected by AttentionInfluence using 1.3B and 7B models.

7 Conclusion

In this paper, we propose AttentionInfluence, a training-free method for selecting high-quality pretraining data by leveraging the activation patterns of attention heads in pretrained LLMs. Unlike traditional classifier-based approaches, our method exploits intrinsic model signals to identify reasoning-intensive data, requiring no additional supervision or manual curation. Experimental results on SmoLLM-Corpus demonstrate that AttentionInfluence consistently improves downstream performance, selects longer and more diverse high-quality data, and aligns well with existing classifier-based patterns—while offering better weak-to-strong generalization. Our findings suggest that internal model mechanisms can serve as reliable indicators of data quality, offering a scalable and efficient pathway for LLM pretraining.

References

- [1] Loubna Ben Allal, Anton Lozhkov, Elie Bakouch, Gabriel Martín Blázquez, Guilherme Penedo, Lewis Tunstall, Andrés Marafioti, Hynek Kydlíček, Agustín Piqueres Lajaráin, Vaibhav Srivastav, et al. Smollm2: When smol goes big-data-centric training of a small language model. [arXiv preprint arXiv:2502.02737](#), 2025.
- [2] Zachary Ankner, Cody Blakeney, Kartik Sreenivasan, Max Marion, Matthew L Leavitt, and Mansheej Paul. Perplexed by perplexity: Perplexity-based data pruning with small reference models. [arXiv preprint arXiv:2405.20541](#), 2024.
- [3] Loubna Ben Allal, Anton Lozhkov, Guilherme Penedo, Thomas Wolf, and Leandro von Werra. Smollm-corpus, 2024. URL <https://huggingface.co/datasets/HuggingFaceTB/smollm-corpus>.
- [4] Yonatan Bisk, Rowan Zellers, Jianfeng Gao, Yejin Choi, et al. Piqa: Reasoning about physical commonsense in natural language. In [Proceedings of the AAAI conference on artificial intelligence](#), volume 34, pages 7432–7439, 2020.
- [5] Trenton Bricken, Adly Templeton, Joshua Batson, Brian Chen, Adam Jermyn, Tom Conerly, Nicholas L Turner, Cem Anil, Carson Denison, Amanda Askell, Robert Lasenby, Yifan Wu, and et al. Towards monosematicity: Decomposing language models with dictionary learning. <https://transformer-circuits.pub/2023-monosemantic-features/index.html>, 2023. Accessed: 2023-10-04.
- [6] Iñigo Casanueva, Tadas Temčinas, Daniela Gerz, Matthew Henderson, and Ivan Vulić. Efficient intent detection with dual sentence encoders. [arXiv preprint arXiv:2003.04807](#), 2020.
- [7] Mark Chen, Jerry Tworek, Heewoo Jun, Qiming Yuan, Henrique Ponde de Oliveira Pinto, Jared Kaplan, Harri Edwards, Yuri Burda, Nicholas Joseph, Greg Brockman, et al. Evaluating large language models trained on code. [arXiv preprint arXiv:2107.03374](#), 2021.
- [8] Peter Clark, Isaac Cowhey, Oren Etzioni, Tushar Khot, Ashish Sabharwal, Carissa Schoenick, and Oyvind Tafjord. Think you have solved question answering? try arc, the ai2 reasoning challenge. [arXiv preprint arXiv:1803.05457](#), 2018.
- [9] Karl Cobbe, Vineet Kosaraju, Mohammad Bavarian, Mark Chen, Heewoo Jun, Lukasz Kaiser, Matthias Plappert, Jerry Tworek, Jacob Hilton, Reiichiro Nakano, et al. Training verifiers to solve math word problems. [arXiv preprint arXiv:2110.14168](#), 2021.
- [10] Dheeru Dua, Yizhong Wang, Pradeep Dasigi, Gabriel Stanovsky, Sameer Singh, and Matt Gardner. Drop: A reading comprehension benchmark requiring discrete reasoning over paragraphs. [arXiv preprint arXiv:1903.00161](#), 2019.
- [11] Yu Fu, Zefan Cai, Abdelkadir Asi, Wayne Xiong, Yue Dong, and Wen Xiao. Not all heads matter: A head-level kv cache compression method with integrated retrieval and reasoning. [arXiv preprint arXiv:2410.19258](#), 2024.
- [12] Mor Geva, Roei Schuster, Jonathan Berant, and Omer Levy. Transformer feed-forward layers are key-value memories. [arXiv preprint arXiv:2012.14913](#), 2020.
- [13] Aaron Grattafiori, Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Alex Vaughan, et al. The llama 3 herd of models. [arXiv preprint arXiv:2407.21783](#), 2024.
- [14] William Held, Bhargavi Paranjape, Punit Singh Koura, Mike Lewis, Frank Zhang, and Todor Mihaylov. Optimizing pretraining data mixtures with llm-estimated utility. [arXiv preprint arXiv:2501.11747](#), 2025.
- [15] Dan Hendrycks, Collin Burns, Steven Basart, Andy Zou, Mantas Mazeika, Dawn Song, and Jacob Steinhardt. Measuring massive multitask language understanding. [arXiv preprint arXiv:2009.03300](#), 2020.
- [16] Dan Hendrycks, Collin Burns, Saurav Kadavath, Akul Arora, Steven Basart, Eric Tang, Dawn Song, and Jacob Steinhardt. Measuring mathematical problem solving with the math dataset. [arXiv preprint arXiv:2103.03874](#), 2021.
- [17] Geoffrey Hinton, Oriol Vinyals, and Jeff Dean. Distilling the knowledge in a neural network. [arXiv preprint arXiv:1503.02531](#), 2015.

- [18] Shengding Hu, Yuge Tu, Xu Han, Chaoqun He, Ganqu Cui, Xiang Long, Zhi Zheng, Yewei Fang, Yuxiang Huang, Weilin Zhao, et al. Minicpm: Unveiling the potential of small language models with scalable training strategies. *arXiv preprint arXiv:2404.06395*, 2024.
- [19] Yuzhen Huang, Yuzhuo Bai, Zhihao Zhu, Junlei Zhang, Jinghan Zhang, Tangjun Su, Junteng Liu, Chuancheng Lv, Yikai Zhang, Yao Fu, et al. C-eval: A multi-level multi-discipline chinese evaluation suite for foundation models. *Advances in Neural Information Processing Systems*, 36:62991–63010, 2023.
- [20] Mandar Joshi, Eunsol Choi, Daniel S Weld, and Luke Zettlemoyer. Triviaqa: A large scale distantly supervised challenge dataset for reading comprehension. *arXiv preprint arXiv:1705.03551*, 2017.
- [21] Armand Joulin, Edouard Grave, Piotr Bojanowski, Matthijs Douze, Hervé Jégou, and Tomas Mikolov. Fasttext. zip: Compressing text classification models. *arXiv preprint arXiv:1612.03651*, 2016.
- [22] Myeongseob Ko, Feiyang Kang, Weiyan Shi, Ming Jin, Zhou Yu, and Ruoxi Jia. The mirrored influence hypothesis: Efficient data influence estimation by harnessing forward passes. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 26286–26295, 2024.
- [23] Guokun Lai, Qizhe Xie, Hanxiao Liu, Yiming Yang, and Eduard Hovy. Race: Large-scale reading comprehension dataset from examinations, 2017. URL <https://arxiv.org/abs/1704.04683>.
- [24] Jeffrey Li, Alex Fang, Georgios Smyrnis, Maor Ivgi, Matt Jordan, Samir Yitzhak Gadre, Hritik Bansal, Etash Guha, Sedrick Scott Keh, Kushal Arora, et al. Datacomp-lm: In search of the next generation of training sets for language models. *Advances in Neural Information Processing Systems*, 37:14200–14282, 2024.
- [25] Ruihang Li, Yixuan Wei, Miaozen Zhang, Nenghai Yu, Han Hu, and Houwen Peng. Scalingfilter: Assessing data quality through inverse utilization of scaling laws. *arXiv preprint arXiv:2408.08310*, 2024.
- [26] Zhenghao Lin, Zhibin Gou, Yeyun Gong, Xiao Liu, Yelong Shen, Ruochen Xu, Chen Lin, Yujiu Yang, Jian Jiao, Nan Duan, et al. Rho-1: Not all tokens are what you need. *arXiv preprint arXiv:2404.07965*, 2024.
- [27] Qian Liu, Xiaosen Zheng, Niklas Muennighoff, Guangtao Zeng, Longxu Dou, Tianyu Pang, Jing Jiang, and Min Lin. Regmix: Data mixture as regression for language model pre-training. *arXiv preprint arXiv:2407.01492*, 2024.
- [28] Ang Lv, Yuhan Chen, Kaiyi Zhang, Yulong Wang, Lifeng Liu, Ji-Rong Wen, Jian Xie, and Rui Yan. Interpreting key mechanisms of factual recall in transformer-based language models. *arXiv preprint arXiv:2403.19521*, 2024.
- [29] Todor Mihaylov, Peter Clark, Tushar Khot, and Ashish Sabharwal. Can a suit of armor conduct electricity? a new dataset for open book question answering. *arXiv preprint arXiv:1809.02789*, 2018.
- [30] Team OLMo, Pete Walsh, Luca Soldaini, Dirk Groeneveld, Kyle Lo, Shane Arora, Akshita Bhagia, Yuling Gu, Shengyi Huang, Matt Jordan, et al. 2 olmo 2 furious. *arXiv preprint arXiv:2501.00656*, 2024.
- [31] Catherine Olsson, Nelson Elhage, Neel Nanda, Nicholas Joseph, Nova DasSarma, Tom Henighan, Ben Mann, Amanda Askell, Yuntao Bai, Anna Chen, et al. In-context learning and induction heads. *arXiv preprint arXiv:2209.11895*, 2022.
- [32] Guilherme Penedo, Hynek Kydlíček, Anton Lozhkov, Margaret Mitchell, Colin A Raffel, Leandro Von Werra, Thomas Wolf, et al. The fineweb datasets: Decanting the web for the finest text data at scale. *Advances in Neural Information Processing Systems*, 37:30811–30849, 2024.
- [33] Ru Peng, Kexin Yang, Yawen Zeng, Junyang Lin, Dayiheng Liu, and Junbo Zhao. Dataman: Data manager for pre-training large language models. *arXiv preprint arXiv:2502.19363*, 2025.
- [34] Zexuan Qiu, Jingjing Li, Shijue Huang, Xiaoqi Jiao, Wanjun Zhong, and Irwin King. Clongeval: A chinese benchmark for evaluating long-context large language models. *arXiv preprint arXiv:2403.03514*, 2024.
- [35] Jack W Rae, Sebastian Borgeaud, Trevor Cai, Katie Millican, Jordan Hoffmann, Francis Song, John Aslanides, Sarah Henderson, Roman Ring, Susannah Young, et al. Scaling language models: Methods, analysis & insights from training gopher. *arXiv preprint arXiv:2112.11446*, 2021.
- [36] Nils Reimers and Iryna Gurevych. Sentence-bert: Sentence embeddings using siamese bert-networks. *arXiv preprint arXiv:1908.10084*, 2019.
- [37] David Rein, Betty Li Hou, Asa Cooper Stickland, Jackson Petty, Richard Yuanzhe Pang, Julien Dirani, Julian Michael, and Samuel R. Bowman. Gpqa: A graduate-level google-proof q&a benchmark, 2023. URL <https://arxiv.org/abs/2311.12022>.

- [38] Laura Ruis, Maximilian Mozes, Juhan Bae, Siddhartha Rao Kamalakara, Dwarak Talupuru, Acyr Locatelli, Robert Kirk, Tim Rocktäschel, Edward Grefenstette, and Max Bartolo. Procedural knowledge in pretraining drives reasoning in large language models. [arXiv preprint arXiv:2411.12580](#), 2024.
- [39] Keisuke Sakaguchi, Ronan Le Bras, Chandra Bhagavatula, and Yejin Choi. Winogrande: An adversarial winograd schema challenge at scale. [Communications of the ACM](#), 64(9):99–106, 2021.
- [40] Dan Su, Kezhi Kong, Ying Lin, Joseph Jennings, Brandon Norick, Markus Kliegl, Mostafa Patwary, Mohammad Shoeybi, and Bryan Catanzaro. Nemotron-cc: Transforming common crawl into a refined long-horizon pretraining dataset. [arXiv preprint arXiv:2412.02595](#), 2024.
- [41] Mirac Suzgun, Nathan Scales, Nathanael Schärli, Sebastian Gehrmann, Yi Tay, Hyung Won Chung, Aakanksha Chowdhery, Quoc V Le, Ed H Chi, Denny Zhou, et al. Challenging big-bench tasks and whether chain-of-thought can solve them. [arXiv preprint arXiv:2210.09261](#), 2022.
- [42] Alon Talmor, Jonathan Herzig, Nicholas Lourie, and Jonathan Berant. Commonsenseqa: A question answering challenge targeting commonsense knowledge. [arXiv preprint arXiv:1811.00937](#), 2018.
- [43] Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, et al. Llama: Open and efficient foundation language models. [arXiv preprint arXiv:2302.13971](#), 2023.
- [44] Yubo Wang, Xueguang Ma, Ge Zhang, Yuansheng Ni, Abhranil Chandra, Shiguang Guo, Weiming Ren, Aaran Arulraj, Xuan He, Ziyan Jiang, et al. Mmlu-pro: A more robust and challenging multi-task language understanding benchmark. In [The Thirty-eight Conference on Neural Information Processing Systems Datasets and Benchmarks Track](#), 2024.
- [45] Alexander Wettig, Aatmik Gupta, Saumya Malik, and Danqi Chen. Qurating: Selecting high-quality data for training language models. [arXiv preprint arXiv:2402.09739](#), 2024.
- [46] Alexander Wettig, Kyle Lo, Sewon Min, Hannaneh Hajishirzi, Danqi Chen, and Luca Soldaini. Organize the web: Constructing domains enhances pre-training data curation. [arXiv preprint arXiv:2502.10341](#), 2025.
- [47] Wenhao Wu, Yizhong Wang, Guangxuan Xiao, Hao Peng, and Yao Fu. Retrieval head mechanistically explains long-context factuality. [arXiv preprint arXiv:2404.15574](#), 2024.
- [48] Sang Michael Xie, Hieu Pham, Xuanyi Dong, Nan Du, Hanxiao Liu, Yifeng Lu, Percy S Liang, Quoc V Le, Tengyu Ma, and Adams Wei Yu. Doremi: Optimizing data mixtures speeds up language model pretraining. [Advances in Neural Information Processing Systems](#), 36:69798–69818, 2023.
- [49] Jiasheng Ye, Peiju Liu, Tianxiang Sun, Jun Zhan, Yunhua Zhou, and Xipeng Qiu. Data mixing laws: Optimizing data mixtures by predicting language modeling performance. [arXiv preprint arXiv:2403.16952](#), 2024.
- [50] Zichun Yu, Spandan Das, and Chenyan Xiong. Mates: Model-aware data selection for efficient pretraining with data influence models. [Advances in Neural Information Processing Systems](#), 37:108735–108759, 2024.
- [51] Rowan Zellers, Ari Holtzman, Yonatan Bisk, Ali Farhadi, and Yejin Choi. Hellaswag: Can a machine really finish your sentence? [arXiv preprint arXiv:1905.07830](#), 2019.
- [52] Ranchi Zhao, Zhen Leng Thai, Yifan Zhang, Shengding Hu, Yunqi Ba, Jie Zhou, Jie Cai, Zhiyuan Liu, and Maosong Sun. Decoratelm: Data engineering through corpus rating, tagging, and editing with language models. [arXiv preprint arXiv:2410.05639](#), 2024.
- [53] Zifan Zheng, Yezhaohui Wang, Yuxin Huang, Shichao Song, Mingchuan Yang, Bo Tang, Feiyu Xiong, and Zhiyu Li. Attention heads of large language models: A survey. [arXiv preprint arXiv:2409.03752](#), 2024.
- [54] Wanjun Zhong, Ruixiang Cui, Yiduo Guo, Yaobo Liang, Shuai Lu, Yanlin Wang, Amin Saied, Weizhu Chen, and Nan Duan. Agieval: A human-centric benchmark for evaluating foundation models. [arXiv preprint arXiv:2304.06364](#), 2023.
- [55] Youxiang Zhu, Ruochen Li, Danqing Wang, Daniel Haehn, and Xiaohui Liang. Focus directions make your language models pay more attention to relevant contexts. [arXiv preprint arXiv:2503.23306](#), 2025.

Appendix

A Synthetic Test Sample

```
model input:  
Please extract the value corresponding to the specified key from the  
following JSON object. Output only the value of the corresponding key  
and nothing else. The JSON data is as follows:  
{context}  
  
{question-shot1}  
{answer-shot1}  
{question-shot2}  
{answer-shot2}  
{question-shot3}  
{answer-shot3}  
{question}  
  
answer:  
{answer}
```

B Evolution of Retrieval Heads in Pretrained Models

We apply the method described in Section 4.1 to identify retrieval heads at six checkpoints of the pretrained 1.3B-parameter model. These checkpoints correspond to training progress at 5B, 307B, 608B, 898B, 1200B, and 1499B tokens, respectively.

C Masking Operation

The "mask" operation is to set the attention weights provided by the specific attention heads to equal weights. And if the length of the sequence is L , the attention weight of each token should be set to $\frac{1}{L}$.

D Effect of Masking Retrieval Heads vs. Random Non-Retrieval Heads

Model	Benchmarks					
	Hellaswag	WinoGrande	MMLU	MMLU-Pro	AGIEval-en	GPQA
1.3B	0.5715	0.6062	0.4258	0.1290	0.2047	0.2203
	DROP	BBH	GSM8K	HumanEval	Banking77-en-ICL	
	0.2344	0.3166	0.1820	0.1707	0.4148	
1.3B (Random Masked, Non-Retrieval Heads)	Hellaswag	WinoGrande	MMLU	MMLU-Pro	AGIEval-en	GPQA
	0.5518	0.6069	0.4165	0.1275	0.2072	0.2071
	DROP	BBH	GSM8K	HumanEval	Banking77-en-ICL	
1.3B (Masked, Retrieval Heads)	0.2190	0.3005	0.1274	0.1159	0.3840	
	Hellaswag	WinoGrande	MMLU	MMLU-Pro	AGIEval-en	GPQA
	0.5493	0.5801	0.3089	0.0305	0.1298	0.1827
1.3B (Masked, Retrieval Heads)	DROP	BBH	GSM8K	HumanEval	Banking77-en-ICL	
	0.1141	0.0429	0.0068	0.1098	0.0001	

Table 6 Effect of Masking Retrieval Heads vs. Random Non-Retrieval Heads on Reasoning and In-Context Learning

Banking77-en-ICL is an internal evaluation task for assessing a model's in-context learning ability. It requires models to perform many-shot classification on the Banking77-en dataset[6]. Here, "Masked, Retrieval Heads" refers to masking attention heads ranked in the top 5% by retrieval score, while "Random Masked, Non-Retrieval Heads" refers to randomly masking heads ranked between the top 5% and top 100% (i.e., the remaining 95%) by retrieval score. We conduct the experiments using the models shown in the Table 7 and

find that masking retrieval heads significantly impairs the model’s reasoning performance, while masking random non-retrieval heads has only a minor effect—consistent with the findings of Wu et al. [47]. In addition, we find that retrieval heads also play an essential role in the model’s in-context learning ability.

E Experiment Setting

Pretraining Data To ensure reproducibility, we use SmoLLM-Corpus[3] as the pretraining dataset. The composition of the SmoLLM-Corpus dataset is shown in the [Table 8](#). We sample 100 million tokens from SmoLLM-Corpus as the validation dataset.

Pretrained models used by AttentionInfluence In this work, AttentionInfluence employs internal pretrained models based on the LLaMA2-alike architecture. The hyperparameters of the models are detailed in [Table 7](#).

model size	pretraining tokens	vocab size	hidden size	ffn inner	num heads	num layers	shared q_head	seq len	tie emb
1.3B	1.5TB	155136	2,560	10,240	20	16	2	4,096	true
7B	9TB	155136	4,096	16,384	32	32	2	8,192	true

Table 7 Hyperparams of the Pretrained Models Used by AttentionInfluence.

Model trained in the experiment The hyperparameters are presented in [Table 9](#), and tokenizer used for training and computing token counts is the same as SmoLLM³ with a vocab size of 49,152.

Pretraining setting Referring to SmoLLM[3], our experiments are conducted with WSD learning rate scheduler [18] with 0.1% warmup steps, 75% stable phase, and a final 25% decay phase. The amount of training tokens is 1 TB. The training is distributed across 32 machines, each equipped with eight H100-80GB GPUs.

Dataset	FineWeb-Edu-Dedup	Cosmopedia-V2	Python-Edu	OpenWebMath
# Tokens (billions)	193.3	27.9	3.8	13.3

Table 8 Composition of the SmoLLM Corpus Dataset.

model size	batch size	learning rate	hidden size	ffn inner	num heads	num layers	shared q_head	seq len	tie emb	total params
7B	1,024	4e-4	4,096	8,192	32	32	4	8,192	false	6.98B

Table 9 Hyperparams of the Model Trained in the Experiment.

Evaluation details To ensure that all demonstrations, along with the question and the generated prediction, fit within the 8192-token context window, we use a different number of few-shot examples per evaluation task. Specifically, we use the following numbers of demonstrations (in parentheses): MATH (`minerva_math`) (4), DROP (3), BBH (3), and 5 for all other tasks. We report accuracy for most tasks, with the following exceptions: `exact_match` for MMLU-Pro, TriviaQA, and BBH; `flexible-extract` for GSM8K; `math_verify` for MATH; and F1 score for DROP. When available, we use the normalized accuracy (`acc_norm`) metric provided by the lm-evaluation-harness. ARC(C+E) denotes the average accuracy over ARC-Challenge (ARC-C) and ARC-Easy (ARC-E). For specific tasks, we adopt the following exceptions:

- For AGIEval, we conduct the official few-shot evaluation using the official AGIEval repository⁴.

³<https://huggingface.co/HuggingFaceTB/cosmo2-tokenizer>

⁴<https://github.com/ruixiangcui/AGIEval/tree/main>

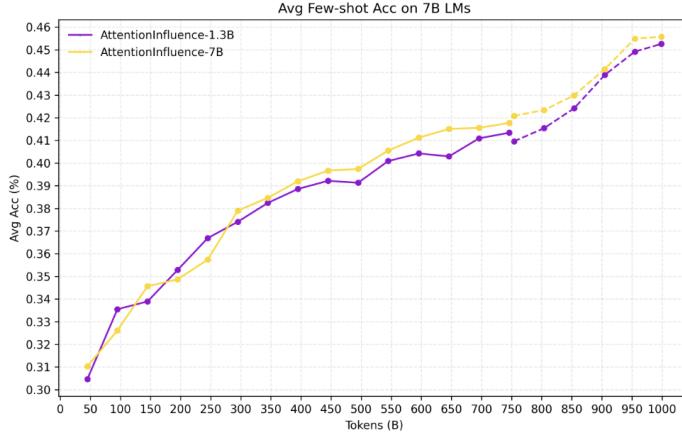


Figure 7 Performance evolution on comprehensive benchmark evaluations during pretraining. The first 750 billion tokens correspond to the pretraining phase, represented by solid lines, while the subsequent 250 billion tokens represent the learning rate annealing phase, represented by dashed lines, using the same dataset. After around 100 billion tokens, AttentionInfluence-1.3B consistently outperforms the baseline across a wide range of tasks on average, including the annealing phase.

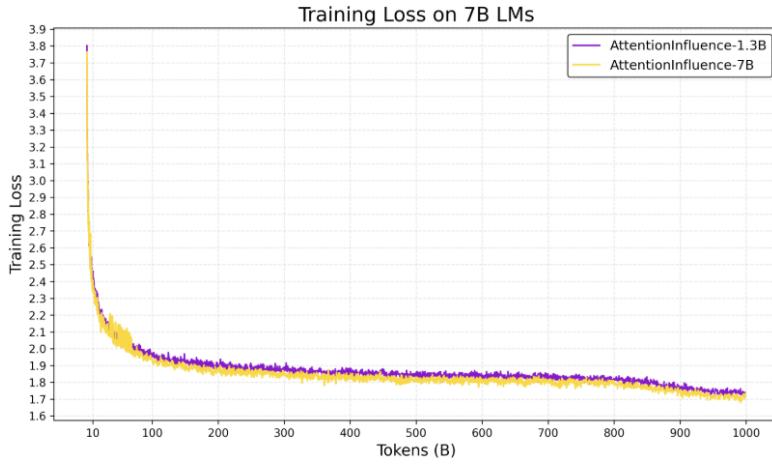


Figure 8 Training loss

- For HumanEval, we conduct zero-shot evaluation using the BigCode evaluation harness⁵ and report pass@1 using the following generation settings, which are the same as those used in SmolLM[3]: temperature = 0.2, top-p = 0.95, n_samples = 20, and max_length_generation = 1024.
- For DROP, we fix a known bug in the lm-evaluation-harness implementation, following the discussion⁶.

F Detailed Performance Evolution During Pretraining

As shown in Figure 7, Figure 9, and Figure 10, we illustrate how the performance of the baseline, the 1.3B method, and the 7B method evolves across different benchmarks as the number of training tokens increases. In addition, panel (b) of Figure 1 and Figure 8 present the training loss comparison among baseline, AttentionInfluence-1.3B, and AttentionInfluence-7B. Furthermore, we report the evaluation results of LLMs trained on data selected by AttentionInfluence-1.3B and AttentionInfluence-7B, as shown in Table 10.

⁵<https://github.com/bigcode-project/bigcode-evaluation-harness>

⁶<https://github.com/EleutherAI/lm-evaluation-harness/issues/2137>

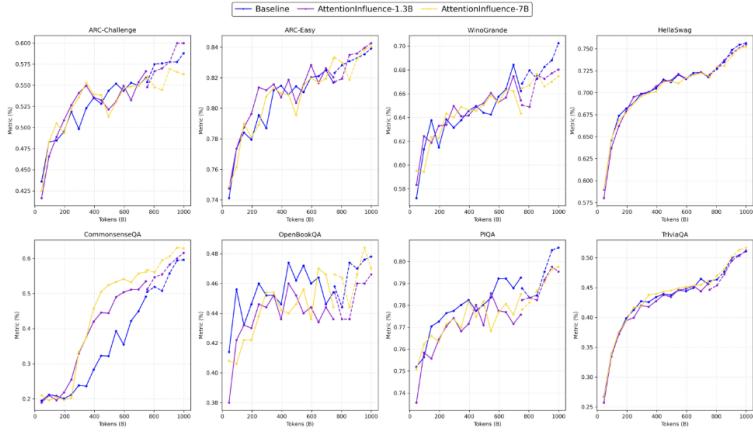


Figure 9 The performance evolution during pretraining on relatively simple benchmarks (i.e., ARC-Challenge, ARC-Easy, Winogrande, HellaSwag, CommonsenseQA, OpenBookQA, Piqa, TrivialQA). The first 750 billion tokens correspond to the standard pretraining phase (solid lines), followed by 250 billion tokens under learning rate annealing (dashed lines). Curves with the same color (solid and dashed) indicate training on the same dataset. After approximately 100 billion tokens, AttentionInfluence-1.3B consistently outperforms the baseline across a broad range of tasks, including during the annealing phase.

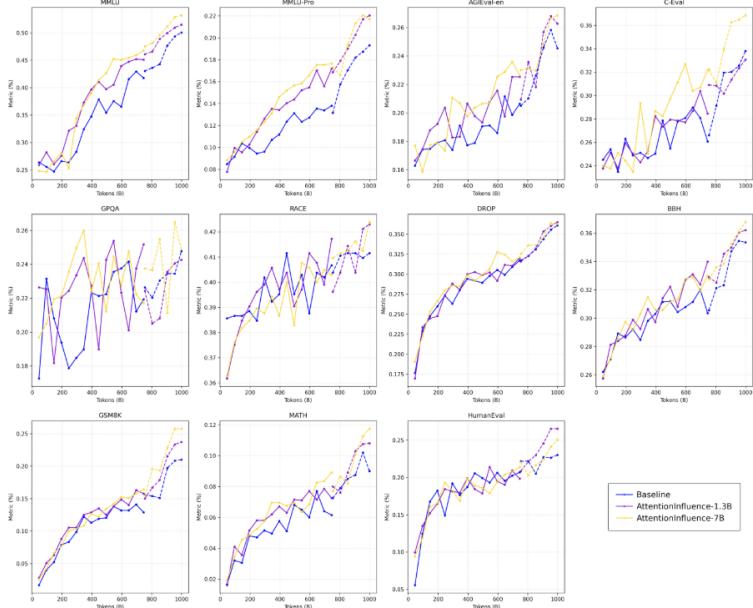


Figure 10 The performance evolution during pretraining on knowledge-intensive and reasoning-heavy benchmarks (i.e., MMLU, MMLU-Pro, AGIEval-en, C-Eval, GPQA, RACE, DROP, BBH, GSM8K, MATH, and HumanEval). The first 750 billion tokens correspond to the standard pretraining phase (solid lines), followed by 250 billion tokens under learning rate annealing (dashed lines). Curves with the same color (solid and dashed) indicate training on the same dataset. After around 100 billion tokens, AttentionInfluence-1.3B consistently outperforms the baseline across a wide range of tasks on average, including the annealing phase.

G LLM-As-A-Judge Experiment Details

We use GPT-4o to evaluate the performance of different data selection methods on the FineWeb-Edu-Dedup domain. On one hand, since most of the data in FineWeb-Edu-Dedup is related to education, we aim for the selected high-quality data to be highly relevant to this domain. Therefore, we design an Education Score based

Model	#Tokens	Avg.	Metrics						
			ARC-C	ARC-E	ARC(C+E)	Wino.	Hella.	CSQA	OpenBookQA
AttentionInfluence-1.3B w/o LRD	746B	41.34	56.66	82.03	69.35	65.43	71.90	53.48	43.60
			PIQA	TriviaQA	MMLU	MMLU-Pro	AGIEval-en	RACE	DROP
			77.58	45.68	45.10	17.19	22.50	41.72	32.03
			BBH	GSM8K	MATH	HumanEval	C-Eval	GPQA	
AttentionInfluence-7B w/o LRD	746B	41.77	33.99	15.77	7.25	19.85	28.45	25.18	
			ARC-C	ARC-E	ARC(C+E)	Wino.	Hella.	CSQA	OpenBookQA
			55.80	83.25	69.53	64.33	71.94	56.18	44.40
			PIQA	TriviaQA	MMLU	MMLU-Pro	AGIEval-en	RACE	DROP
AttentionInfluence-1.3B w/ LRD	1T	45.26	78.51	46.14	46.77	17.64	22.64	40.29	32.09
			BBH	GSM8K	MATH	HumanEval	C-Eval	GPQA	
			33.02	16.45	6.78	21.40	32.17	21.73	
			ARC-C	ARC-E	ARC(C+E)	Wino.	Hella.	CSQA	OpenBookQA
AttentionInfluence-7B w/ LRD	1T	45.57	59.98	84.26	72.12	68.03	75.49	61.59	46.60
			PIQA	TriviaQA	MMLU	MMLU-Pro	AGIEval-en	RACE	DROP
			79.54	51.20	51.48	22.03	26.30	42.30	36.52
			BBH	GSM8K	MATH	HumanEval	C-Eval	GPQA	
			36.22	23.73	10.80	26.55	33.06	24.26	
			ARC-C	ARC-E	ARC(C+E)	Wino.	Hella.	CSQA	OpenBookQA
			56.31	84.05	70.18	67.48	75.24	62.90	47.00
			PIQA	TriviaQA	MMLU	MMLU-Pro	AGIEval-en	RACE	DROP
			79.76	51.68	53.18	21.70	26.85	42.39	36.25
			BBH	GSM8K	MATH	HumanEval	C-Eval	GPQA	
			36.80	25.78	11.75	25.06	36.85	24.87	

Table 10 The ablation results on various benchmarks. The LRD denotes learning rate decay.

on whether the selected sample content is education-related. On the other hand, we want the selected samples to contain more complex, reasoning-intensive knowledge. Based on this criterion, we design a Reasoning Score.

In summary, we use the following prompt to have GPT-4o score the selected samples:

🔗 LLM-As-A-Judge

Prompt:

Given a piece of text: **<Selected Sample>**. Determine whether the text has educational value. If it does, respond with 1; if not, respond with 0. Then, determine whether the text is reasoning-intensive — that is, whether it contains explicit or implicit logical reasoning chains. If it does, respond with 1; if not, respond with 0. Respond in the following format:

```
\#\#Educational Value Score
<educational value score>
```

```
\#\#Reasoning Intensive Score
<reasoning intensive score>
```

Although GPT-4o can also be used for scoring pretraining data, different domains require specially designed prompts. Moreover, the computational cost of using GPT-4o for scoring is very high, whereas AttentionInfluence-1.3B has a much lower computational overhead.

H Details of Clustering

We obtain document embeddings using Sentence-BERT[36] and apply K-means clustering with $k = 100$. For each cluster, we sample representative documents near the cluster center and use GPT-4o to generate descriptive fine-grained (i.e., secondary) category labels, such as *Education-Teaching & Resources*.

We manually group these secondary labels into six primary categories and report the number of samples falling into each high-level category for both selection methods.

Method	Ranking	Words
AttentionInfluence	0%- 1%	frac, len, sklearn, append, pyplot, browser, pre, mathbf, 3d, employee, __init__
	1%- 10%	well, part, movement, children, appreciation, involve, remember, family growth, treatment, principles, business, b, long, work
	10%- 50%	maximize, paintings, independence, therefore, expenses, regulatory, recall square, protocols, monitoring, integrity, consistent, channels, inspiring, width
	50%- 100%	driver, flying, humble, fourier, smoother, longstanding, owl personnel, lawyers, entrenched, beach, brother, oils, wow, desk
FineWeb-Edu Classifier	0%- 1%	dimensional, student, 3d, 19th, eco, anti israelite, bmatrix, voter, socio, linspace
	1%- 10%	creative, based, would, sources, do, system, compared, someone studies, delve, true, turn, only, elements, ultimately
	10%- 50%	argument, bright, rising, excessive, governments, friendships, complicated, discipline constitutes, hearing, consequences, institutional, match, meets, holocaust
	50%- 100%	peek, manifest, reciprocity, obligations, toilet, customized, olive validity, enriching, profits, presentations, twelve, originating, arithmetic, nazi

Table 11 The high-frequency words of different methods.

Rank: top 0.24% (AttentionInfluence-1.3B)	Rank: top 0.74% (FineWebEdu Classifier)
<p>Consider a board similar to the one below\\</p> <pre> 7 8 9 10 \\ 6 1 2 11 \\ 5 4 3 \\ </pre> <p>However, imagine it as being infinite. A die is initially placed at 1 and can only move to the next consecutive number (e.g 1 to 2, 2 to 3...) Prompts the user for a natural number N at least equal to 1, and outputs the numbers at the top, the front and the right after the die has been moved to cell N.</p> <p>Written by Benny Hwang 13/08/2017</p> <pre> import math def move_right(Current_faces): Top_old = Current_faces[0] Right_old = Current_faces[2] Bottom_old = Current_faces[3] Left_old = Current_faces[5] ... if __name__ == '__main__': N = False while N == False: ... </pre>	<pre> # Chapter 01 # Exercise 04 # Write a method to replace all spaces in a string with '%20' # Pretty basic for Python def main(): test_string = """This is a test string""" print spaces(test_string) def spaces_easy(input): return input.replace(' ', '%20') if __name__ == "__main__": main() </pre>

Figure 11 The sample in Python-Edu domain ranked within the top 20% according to AttentionInfluence-1.3B (**left**) an FineWeb-Edu Classifier (**right**).

I Case Study

In this section, we present the cases selected by FineWeb-Edu Classifier and AttentionInfluence-1.3B.

Rank: top 0.24% (AttentionInfluence-1.3B)	Rank: top 0.74% (FineWebEdu Classifier)
<p>"17Calculus - Vector Cross Product Application - Triple Scalar Product</p> <p>17Calculus</p> <p>The triple scalar product is a result of combining the dot product with the cross product. Some other names for the triple scalar product are scalar triple product, mixed product and box product. First, let's define what it is and then discuss a couple of properties.</p> <p>Definition and Notation</p> <p>If we have three vectors in space $\vec{u} = u_x \hat{i} + u_y \hat{j} + u_z \hat{k}$, $\vec{v} = v_x \hat{i} + v_y \hat{j} + v_z \hat{k}$ and $\vec{w} = w_x \hat{i} + w_y \hat{j} + w_z \hat{k}$, then the triple scalar product is defined to be $\vec{u} \cdot (\vec{v} \times \vec{w})$. The calculation of this can be done as follows:</p> $\begin{vmatrix} \vec{u}_x & \vec{u}_y & \vec{u}_z \\ \vec{v}_x & \vec{v}_y & \vec{v}_z \\ \vec{w}_x & \vec{w}_y & \vec{w}_z \end{vmatrix}$ <p>Let's look at where this comes from.</p> <p>Theorem: Triple Scalar Product</p> <p>If we have three vectors in space, $\vec{u} = u_x \hat{i} + u_y \hat{j} + u_z \hat{k}$, $\vec{v} = v_x \hat{i} + v_y \hat{j} + v_z \hat{k}$ and $\vec{w} = w_x \hat{i} + w_y \hat{j} + w_z \hat{k}$, ...</p>	<p># Compressibility</p> <p>(Redirected from Incompressible) "Incompressible" redirects here. For the property of vector fields, see Solenoidal vector field. For the topological property, see Incompressible surface.</p> <p>In thermodynamics and fluid mechanics, compressibility is a measure of the relative volume change of a fluid or solid as a response to a pressure (or mean stress) change.</p> <p>$\beta = -\frac{1}{V} \frac{\partial V}{\partial p}$ where V is volume and p is pressure.</p> <p>## Definition ...</p> <p>## Fluid dynamics</p> <p>The degree of compressibility of a fluid has strong implications for its dynamics. Most notably, the propagation of sound is dependent on the compressibility of the medium.</p> <p>### Aeronautical dynamics</p> <p>Compressibility is an important factor in aerodynamics.</p>

Figure 12 The sample in OpenWebMath domain ranked within the top 20% according to AttentionInfluence-1.3B (**left**) and FineWeb-Edu Classifier (**right**).

Sample1 (Tag: Health & Medicine - Health Guidelines & Nutrition)	Sample2 (Tag: Technology and Engineering - Information Technology)
<p>Type 2 diabetes is a chronic illness costing over \$300 billion per year in the United States with an estimated 100 million individuals with diabetes or pre-diabetes. Complications due to diabetes place individuals at increased risk for heart attack, stroke, amputations, blindness, kidney failure, disability, and early death. Education has been shown to be effective in improving health behaviors that decrease complications due to diabetes. Common risk factors for development of diabetes are modifiable behaviors such as sedentary lifestyle and obesity. A peer-led approach to diabetes education has the potential to overcome multiple barriers to receiving education. Peer-led diabetes education can provide education at low or no cost in communities where individuals feel welcomed and travel is minimized. Diabetes education has the potential to decrease disability, early death, and the economic costs of diabetes.</p> <p>The purpose of the study was to determine if peer-led sessions on diabetes self-management impacted health behaviors, empowerment, and knowledge of diabetes. Four topic-driven educational sessions were provided for participants in Northeast Arkansas who had either a diagnosis of pre-diabetes or diabetes. Pre and post-questionnaires were used to assess changes in knowledge using the Revised Diabetes Knowledge Test, empowerment using the Diabetes Empowerment Scale - Short Form, and health behaviors.</p> <p>A statistically significant difference was found in the empowerment scale with an increase in mean scores from 31.23 to 36.04. A paired samples t-test found a statistically significant difference in scores on Diabetes Knowledge Test, ($t(25) = -2.54$, $p < .05$). Significant changes in health behaviors were found for knowledge of A1C levels, the frequency of foot exams, and days of exercise per week. Focus groups following intervention provided qualitative results indicating satisfaction with the peer-led model. In order to implement peer-led education, there is a need to develop improved strategies for recruitment. A peer-led model for diabetes education has potential to provide needed education.</p> <p>[Committee: Guffey, James S., Hall, John, Nichols, Joseph, Nix, Elizabeth School: Arkansas State University School Location: United States – Arkansas] Source: DAI-A 80/09(E), Dissertation Abstracts International Subjects: Educational leadership, Public Health Education, Nutrition Keywords: Community, Diabetes, Education, Peer-led]</p> <p>Copyright in each Dissertation and Thesis is retained by the author. All Rights ReservedThe supplemental file or files you are about to download were provided to ProQuest by the author as part of dissertation or thesis. The supplemental files are provided "AS IS" without warranty. ProQuest is not responsible for the content, format or impact on the supplemental file(s) on our system. In some cases, the file type may be unknown or may be a .exe file. We recommend caution as you open such files.</p>	<p>Bitcoin mining is a process of verifying transactions and recording them on the blockchain ledger. The blockchain is a decentralized public ledger that keeps a record of all Bitcoin transactions. Mining involves solving complex mathematical problems using specialized software and hardware. Explore quemasai.io for further information.</p> <p>The Bitcoin network rewards miners for successfully verifying transactions by giving them newly created Bitcoins. The mining process involves adding a new block of transactions to the blockchain every 10 minutes. Miners compete against each other to add the next block to the chain.</p> <p>To participate in Bitcoin mining, one needs to have a powerful hardware setup and specialized mining software. The hardware required is called an ASIC miner, which is specially designed to solve the mathematical problems required to add a block to the blockchain.</p> <p>The Bitcoin network is designed to gradually decrease the mining reward over time. As the number of Bitcoins in circulation increases, the mining reward decreases. This is done to maintain the scarcity and value of Bitcoin. Bitcoin mining requires a significant amount of energy, which has led to concerns about its environmental impact. However, many miners are taking steps to use renewable energy sources to power their mining operations. In summary, Bitcoin mining is a competitive process that involves verifying transactions and adding them to the blockchain ledger. It requires specialized hardware and software and rewards miners with newly created Bitcoins. Although it consumes a significant amount of energy, advances in renewable energy are making Bitcoin mining more sustainable. What exactly is Bitcoin mining?</p> <p>Bitcoin mining is the process of adding new transactions to the blockchain and verifying them. It's done by solving complex mathematical problems and recording those transactions on a public ledger known as the blockchain. The miners who successfully solve these problems are rewarded with newly generated bitcoins. The mining process involves many miners around the world competing to solve these problems, and the first one to do so earns the reward, which is currently 6.25 bitcoins. This reward is then divided among the miners who participated in the process. But mining bitcoin requires a lot of computing power, which means it requires a lot of energy. In fact, according to the Cambridge Bitcoin Electricity Consumption Index, bitcoin mining now consumes as much energy as Switzerland, a country with a population of 8 million. Despite its energy consumption, Bitcoin mining is essential to the functioning of the currency. Without mining, there would be no way to ensure the integrity of the transactions, and the decentralized nature of the currency would be undermined. In recent years, some critics have raised concerns about the environmental impact of Bitcoin mining. However, efforts are being made to reduce the energy consumption associated with the process, ...</p>

Figure 13 The samples of a clustering in data in the Cosmopedia-V2 domain ranked within 20% according to AttentionInfluence.

J Clustering Case

As shown in Figure 13, we present the two clustering cases in the Cosmopedia-V2 domain.

1.3B (Top 0.10%)	1.3B (Top 97.95%)
<pre>## Modeling Dynamic Systems in Python\n\nIn this section, we will explore how to model dynamic systems using Python. We will focus on a specific example involving the equations of motion for an aircraft, but the concepts and techniques we cover will be applicable to a wide range of dynamic systems.\n\n### Equations of Motion\n\nThe equations of motion for an aircraft can be quite complex, as they involve multiple coordinate systems and take into account various forces and moments acting on the aircraft. However, we can simplify the problem by considering a specific set of equations known as the "flat Earth equations." These equations assume that the Earth is flat and non-rotating, which is a reasonable approximation for many applications.\n\nThe flat Earth equations can be written in the following form:\n```\npython\nnot = (q * sin(phi) + r * cos(phi)) / cos(theta)\n```\nwhere `ot` is the "out-of-track" error, which represents the lateral deviation of the aircraft from its intended course. The variables `q`, `r`, `phi`, and `theta` are related to the aircraft's angular rates and orientation.\n\n### Moment Equations\n\nThe moment equations describe how the angular rates of the aircraft change over time. These equations take into account the moments generated by the aircraft's control surfaces, as well as any external disturbances such as wind gusts.\n\nThe moment equations can be written in the following form:\n```\npython\nnp_dot = (j_xz * (j_x - j_y + j_z) * p * q - j_z * (j_z - j_y) + j_xz ** 2) * q * r + j_z * roll + j_xz * yaw) / gamma\nq_dot = ((j_z - j_x) * p * r - j_xz * (p ** 2 - r ** 2) + pitch) / j_y\nnr_dot = (((j_x - j_y) * j_x + j_xz ** 2) * p * q - j_xz * (j_x - j_y + j_z) * q * r + j_xz * roll + j_x * yaw) / gamma\n```\n`p_dot`, `q_dot`, and `r_dot` are the time derivatives of the angular rates; `j_x`, `j_y`, and `j_z` are ...</pre>	<p>I remember watching this indie film last year that really got me thinking about the way society treats certain racial and ethnic groups. It was called "Beyond Skin Deep" and told the story of a young African American woman named Tasha who moves to a small, predominantly white town in the Midwest. Throughout the movie, we see how Tasha faces subtle (and not-so-subtle) racism from her neighbors, coworkers, and even some friends. But what struck me most were the scenes showing how she struggled to fit in and find a sense of belonging in a community that seemed to reject her at every turn. One scene in particular has stuck with me. Tasha is at a local bar with some colleagues after work, trying to make conversation and connect with them. But instead of engaging with her, they talk over her, ignore her contributions to the conversation, and eventually leave without inviting her along. As she watches them go, tears well up in her eyes and she looks around the now-empty bar, feeling completely alone and isolated. What made this film so powerful, in my opinion, was the way it used depictions of race and ethnicity to shed light on broader societal frustrations. By focusing on one character's experience, it highlighted the systemic issues that many people of color face on a daily basis - things like microaggressions, implicit bias, and exclusion. But just when you think you know where the story is going, there's an unexpected plot twist. It turns out that Tasha isn't actually African American - she's Middle Eastern, but had been passing as black because she felt more accepted in that community than in her own. This revelation forces us to reevaluate everything we thought we knew about Tasha's struggles, and challenges us to consider the ways in which our assumptions and prejudices can blind us to the true complexities</p>
7B (Top 0.10%)	7B (Top 97.95%)
<pre>## Understanding Dictionaries and Lists in Python\n\nPython is a powerful programming language that allows us to work with different types of data. In this unit, we will explore two essential data structures: dictionaries and lists. We will also learn how to manipulate and analyze data using these structures.\n\n### Dictionaries\n\nA dictionary in Python is a collection of key-value pairs. It is an unordered collection, meaning that the items do not have a specific order. Each key-value pair is called an item. The syntax for creating a dictionary is as follows:\n```\npython\nmy_dict = {\n "key1": "value1",\n "key2": "value2",\n "key3": "value3"\n}\n```\nYou can access the values in a dictionary using their corresponding keys:\n```\npython\nprint(my_dict["key1"]) # Output: "value1"\n```\n\n### Lists\n\nA list in Python is an ordered collection of items. It is similar to an array in other programming languages. The syntax for creating a list is as follows:\n```\npython\nmy_list = ["item1", "item2", "item3"]\n```\nYou can access the items in a list using their index, which starts at 0:\n```\npython\nprint(my_list[0]) # Output: "item1"\n```\n\n### Analyzing Data with Dictionaries and Lists\n\nNow that we have a basic understanding of dictionaries and lists, let's explore how we can use them to analyze data. We will use a code snippet from a Python tutorial as an example.\n\nThe Code Snippet\n\nHere is the code snippet we will be analyzing:\n```\npython\nresult[track.name] = {\n "cues": firstK, # Cues candidates\n "cuesFeature": {\n features: len([t for t in signal.times if t in firstK]) / len(firstK) if len(firstK) else 0\n }\n}\nfor j, signal in enumerate(peakSignals):\n if any(gttracks):\n gtCues += gttracks[i].features["boundaries"]\n result[track.name][gtCues] = gttracks[i].features["boundaries"]\n # Cues annotated\n result[track.name][gtCuesFeature] = {\n features: len([\n ...</pre>	<p>In today's digital age, businesses rely heavily on complex computer networks to connect their operations, communicate with clients, and store vast amounts of data. At the heart of these networks lies the work of skilled networking professionals who design, implement, and maintain these critical systems. If you are interested in pursuing a career in this field, obtaining a CCNA (Cisco Certified Network Associate) certification can serve as an excellent starting point. In particular, gaining expertise in CCNA (Cisco Certified Architect) architecture can set you apart as a true leader in network design and strategy. Before diving into the specifics of CCNA architecture, it's essential to understand the foundational principles that underpin all networking technologies. At its core, networking involves connecting multiple devices—such as computers, servers, and smartphones—to enable communication and resource sharing. To accomplish this goal, networks employ various layers of hardware and software components working together to transmit information between nodes efficiently and securely. These layers follow well-defined standards and protocols, ensuring seamless interoperability across different vendors and platforms. As a leading provider of networking equipment and solutions, Cisco has established itself as a dominant force within the industry. With a diverse range of products catering to organizations of all sizes, Cisco offers numerous certifications designed to validate the skills and knowledge of networking professionals at every stage of their careers. Among them, the CCNA stands out as an ideal entry point for those new to the field, providing a solid foundation in networking fundamentals while also serving as a stepping stone toward more advanced credentials like the CCNA. Obtaining a CCNA certification requires passing a single exam, known as</p>

Figure 14 The cases of AttentionInfluence in Cosmopeida-V2 domain.

K Cases of AttentionInfluence based on 1.3B and 7B Models

As shown in Figure 14, Figure 15, Figure 16, and Figure 17, we present some cases with different score levels.

<p>1.3B (Top 0.10%)</p> <p>Excel is a popular tool for data analysis, and its usage has increased significantly in recent years. It provides numerous features that make managing data easier. One such feature is the 'Save As' function that helps users create a copy of an existing Excel file with a new name and file format. In this article, we will discuss the 'Save As' function and the keyboard shortcut used for it.\nWhat is the 'Save As' function in Excel?\nThe 'Save As' function in Excel allows users to create a copy of an existing file and save it with a new name or file format. This function is useful when you want to make a copy of an Excel file as a backup, create a new version of the file, or save the file in a different format that is compatible with other applications or systems.\nWhy is the 'Save As' function important?\nThe 'Save As' function is essential because it helps users avoid overwriting their original files accidentally. When you save an Excel file using the 'Save As' function, a new copy of the file is created, and the original file remains unchanged. This way, you can always revert to the original file if necessary.\nWhat is the keyboard shortcut for the 'Save As' function in Excel?\nThe keyboard shortcut for the 'Save As' function in Excel is 'F12'. Pressing the 'F12' key brings up the 'Save As' dialog box, where you can choose the location, name, and file format for the new copy of the file.\nHow to use the 'Save As' function using the keyboard shortcut?\nUsing the 'Save As' function using the keyboard shortcut is easy. Follow the steps below:\n- Open the Excel file you want to save as a new copy\n- Press 'F12' on your keyboard\n- The 'Save As' dialog box will appear\n- Choose the location where you want to save the new copy of the file\n- Enter a new name for the file in the 'File name' field\n- Select the file format you want to use from the 'Save as type' dropdown menu\n- Click the 'Save' button\nWhat are the benefits of using the keyboard shortcut.....</p>	<p>1.3B (Top 97.95%)</p> <p>- Nano Fish Limnophila hippuridoides is originally from Asia and the stalks grow to be 20-50 cm high and 6-10 cm wide – often with beautiful outwards crooked shoot tips. A simple plant, able to adjust to various conditions. The leaves are green with a red-violet underside, and the whole leaf turns red-violet under ideal growth conditions. A vigorously growing plant that willingly creates new, solid shoots from the base. Thinning of the oldest and longest shoots is recommended, in order to make room for such new shoots. Replant the cut-offs, they will soon grow new roots. If either stem or leaves are damaged, a strong scent is emitted. Growth rate: Medium Height: 20 - 30+ Light demand: Medium CO2 : Low</p>
<p>7B (Top 0.10%)</p> <p>Understanding the Three Common Causes of Sensor Failure\nIn today's technologically advanced world, sensors play a crucial role in various industries, from automotive to healthcare. These devices are designed to detect and measure physical properties, enabling machines and systems to operate efficiently. However, like any other piece of technology, sensors are not immune to failure. Understanding the common causes behind sensor failure is essential for businesses and individuals relying on these devices to ensure smooth operations and prevent costly disruptions.\nOne of the primary causes of sensor failure is environmental factors. Sensors are often exposed to harsh conditions, such as extreme temperatures, humidity, or corrosive substances. Over time, these factors can degrade the sensor's components, leading to inaccurate readings or complete malfunction. For instance, in industrial settings where sensors are exposed to high temperatures or corrosive chemicals, the lifespan of the sensor may be significantly reduced. It is crucial to select sensors that are specifically designed to withstand the environmental conditions they will be exposed to, ensuring their longevity and reliability.\nAnother common cause of sensor failure is mechanical stress. Sensors are often subjected to physical forces, such as vibrations, shocks, or excessive pressure. These external forces can damage the delicate internal components of the sensor, resulting in inaccurate measurements or complete failure. For example, in automotive applications, sensors may be exposed to constant vibrations or sudden impacts, which can lead to premature failure if not properly protected. Employing appropriate mounting techniques and using protective measures, such as shock absorbers or vibration dampeners, can help mitigate the risk of mechanical stress-induced sensor failure.\nElectrical issues also contribute significantly to sensor failure. Power surges, voltage spikes,</p>	<p>7B (Top 97.95%)</p> <p>An eye-opening look at the life and legacy of Jackie Robinson, the man who broke the color barrier in Major League Baseball and became an American hero. Baseball, basketball, football — no matter the game, Jackie Robinson excelled. His talents would have easily landed another man a career in pro sports, but such opportunities were closed to athletes like Jackie for one reason: his skin was the wrong color. Settling for playing baseball in the Negro Leagues, Jackie chafed at the inability to prove himself where it mattered most: the major leagues. Then in 1946, Branch Rickey, manager of the Brooklyn Dodgers, recruited Jackie Robinson. Jackie faced cruel and sometimes violent hatred and discrimination, but he proved himself again and again, exhibiting courage, determination, restraint, and a phenomenal ability to play the game. In this compelling biography, award-winning author Doreen Rappaport chronicles the extraordinary life of Jackie Robinson and how his achievements won over — and changed — a segregated nation.</p> <p>Potentially Sensitive Areas: Violence, Racism and racist language Booklist (September 1, 2017 (Vol. 114, No. 1)) Grades 5-7. Early on, young Jackie Robinson was taught to fight back when faced with racial slurs and prejudice, and he did, first as one of the few black kids in his neighborhood and later as one of the few black officers on his army base. But those injustices and the indignities he endured while playing for Negro league baseball were dwarfed by the hostility shown by many white players and fans when he broke the color barrier in Major League Baseball. While children's</p>

Figure 15 The cases of AttentionInfluence in FineWeb-Edu-Dedup domain.

<p>1.3B (Top 0.10%)</p> <p>The Associative Property of Addition is one of four basic properties that students will learn in early addition lessons and use later in multiplication and pre-algebra. Remembering the formula for commutative property of addition is $a + b = b + a$ and you are good to go! The commutative property is a fundamental building block of math, but it only works for addition and multiplication. By non-commutative, we mean the switching of the order will give different results. Example 1: $2 + 4 = 4 + 2 = 6$. What is the Commutative Property? The mathematical operations, subtraction and division are the two non-commutative operations. You can find them all at the bottom of this page. The commutative property for any two numbers, X and Y, is $X \# Y = Y \# X$ where $\#$ can stand for addition or multiplication. The commutative property of addition essentially states that no matter what order the addends are in within a particular number sentence, the sums will be the same. The product of any number and 0 is 0 For example: $874 \times 0 = 0$ Identity Property of Addition & ... Subtraction (Not Commutative) Subtraction is probably an example that you know, intuitively, is not commutative. $16y + 0 = 16y$ Associate Property of Addition Zero Property of Multiplication Commutative Property of Addition Identity Property of Addition 2. $d \bullet r = r \bullet d$ Commutative Property of Multiplication Identity Property of . This rule just says that, when you are doing addition, it doesn't matter which order the numbers are in. Just enter the inputs, the commutative property of addition calculator will update you the result. Addition and multiplication are both commutative. The properties include the commutative, identity, and distributive properties--all of which I cover in other math lessons. The commutative property of addition also applies to variables similarly. Commutative Property Of Addition:</p>	<p>1.3B (Top 97.95%)</p> <p>Article Impact Of Fading Correlation And Unequal Branch Gains On The Capacity Of Diversity Systems Dept. of Electr. Eng., California Inst. of Technol., Pasadena, CA Vehicular Technology Conference, 1988, IEEE 38th 11/2001; DOI:10.1109/VETEC.1999.778436 In proceeding of: Vehicular Technology Conference, 1999 IEEE 49th, Volume: 3 Source: IEEE Xplore ABSTRACT We investigate the effect of fading correlation and branch gain imbalance on the Shannon capacity of diversity systems in conjunction with adaptive transmission techniques. This capacity provides the theoretical upper bound for the spectral efficiency of adaptive transmission schemes. We obtain closed-form expressions for this capacity for Rayleigh fading channels under four adaptation policies: optimal power and rate adaptation (opra), optimal rate adaptation with constant power (ora), truncated channel inversion with fixed rate (tifr), and complete channel inversion with fixed rate (cifr). We give numerical examples illustrating the main trends and offer comparisons on the behavior of opra, ora, tifr, and cifr under variation of different parameters. 1. 0 0 0 Bookmarks 22 Views • Source Article: Capacity of Rayleigh fading channels under different adaptive transmission and diversity-combining techniques [hide abstract] IEEE Transactions on Vehicular Technology 08/1999; · 2.06 Impact Factor • Source Article: Capacity of fading channels with channel side information [hide abstract] ABSTRACT: ...</p>
<p>7B (Top 0.10%)</p> <p>## Sunday, February 8, 2009\n\n6. How Euler Derived the Continuity Equation\n[Previous Article: The Reynolds Transport Theorem]\n\nI thought that it would be interesting to present Euler's derivation of the continuity equation for incompressible flows. Although d'Alembert, in 1752, had already presented an equivalent equation in his <i>Essai d'une nouvelle théorie de la résistance des fluides</i> (which he had already submitted to the Academy of Sciences of Berlin in 1749), the one proposed by Euler in 1756 (written 1752) is considered to be the most rigorous.\n\nEuler's contribution to Fluid Mechanics goes beyond what a scientist may imagine, and was mostly due to four manuscripts published between 1752 and 1755. These are\n\n1. <i>Principia Motus Fluidorum</i> (1756) [pdf]\n2. <i>Principes généraux de l'état d'équilibre des fluides</i> (1755) [pdf]\n3. <i>Principes généraux du mouvement des fluides</i> (1755) [pdf]\n4. <i>Continuation des recherches sur la théorie de mouvement des fluides</i> (1755) [pdf]\n\nThe final thing I would like to point out is that Euler's genius lies partly in his ability to synthesize and introduce world class notation. In this way, he was able to supersede all his predecessors.\n\nEuler starts by saying: "... I shall posit that the fluid cannot be compressed into a smaller space, and its continuity cannot be interrupted. I stipulate without qualification that, in the course of the motion within the fluid, no empty space is left by the fluid, but it always maintains continuity in this motion..." [Paragraph 6, <i>Principia Motus Fluidorum</i>, Translated by Enlin Pan]\n\nHe then argues that if one considers any part of a fluid of this type (i.e. incompressible), then each individual particles fill the same amount of space as they move around. He then infers that if this happens for particles, it should happen to the fluid as a whole (which was his assumption of incompressibility). One is now able to consider an arbitrary fluid element and then track its instantaneous changes "to determine</p>	<p>7B (Top 97.95%)</p> <p>Gitlab-runner (docker-machine) concurrency and request-concurrency? Can anyone tell me how to set on gitlab-runner (docker-machine) parameters: --limit --request-concurrency --machine-idle-nodes concurrency (cannot be set from CLI) ? Is --request-concurrency same as concurrency param but just for docker-machine executor ? I would like to have 2 idle nodes, 3 parallel jobs per node and max limit of nodes 10. I am getting WARN message: WARNING: Specified limit (10) larger then current concurrent limit (1). Concurrent limit will not be enlarged. Thanks EDIT: concurrency should be number of cores + 1 ? and also concurrency=request-concurrency ?</p>

Figure 16 The cases of AttentionInfluence in OpenWebMath domain.

Figure 17 The cases of AttentionInfluence in Python-Edu domain.

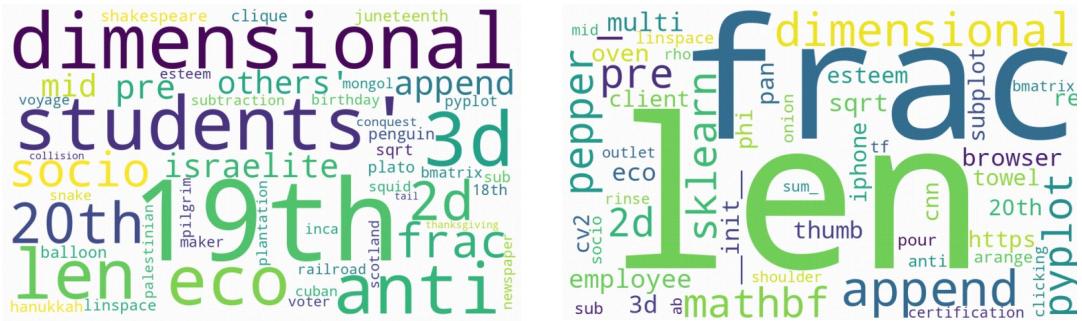


Figure 18 The cloud maps of the data selected by AttentionInfluence and FineWeb-Edu Classifier, respectively. The left part is the cloud map of FineWeb-Edu Classifier, the right part is that of AttentionInfluence.

Specific Word: sklearn	Specific Word: 19th
<p>K-Nearest Neighbors Classifier and Hyperparameter Tuning</p>	
<p>In this chapter, we will explore the K-Nearest Neighbors (KNN) classifier, a fundamental machine learning algorithm, and learn how to optimize its performance by tuning hyperparameters. We will use Python, along with popular libraries such as pandas, NumPy, scikit-learn, and matplotlib.</p>	
<p>K-Nearest Neighbors Classifier</p>	
<p>The KNN classifier is a simple yet powerful algorithm used for both classification and regression tasks. It is a type of instance-based learning, ...</p>	
<p>First, let's import the necessary libraries:</p>	
<pre>\begin{verbatim} import pandas as pd import numpy as np from sklearn.model_selection import train_test_split from sklearn.neighbors import KNeighborsClassifier import matplotlib.pyplot as plt \end{verbatim}</pre>	
<p>Next, we will load our dataset, which is a pandas DataFrame df containing the columns 'cases' and 'date'. We will use only these two columns for our analysis: ...</p>	<p>Chapter Title: Discovering Sacred Solo Voices in MusicImagine walking into a grand cathedral, dimly lit with tall stained glass windows casting colorful patterns on the cool stone floors. As you take a deep breath, a single voice fills the air, resonating off the walls and ceilings. This soloist sings sacred music – songs written specifically for worship services or religious ceremonies. Through this chapter, we'll embark on an adventure exploring different types of sacred solo voices in various cultures and time periods.</p> <p>Section 1: Gregorian Chant - Monks and Nuns Singing Prayers---In medieval Europe (around 500–1400 AD), monks and nuns created simple yet powerful chants called Gregorian chants. These were sung during Catholic Masses as they believed singing was praying twice! They used only one melody line, which meant everyone sang together in unison. Listen to an example here: <https://www.youtube.com/watch?v=zgYQE7jxx28>. How does it make you feel?</p> <p>Section 2: Indian Classical Music - Exploring Ragas---Let's travel across continents to explore India's rich tradition of classical music. Unlike Western music, Indian classical music focuses heavily on improvisation within specific rules. One popular form is called khayal, where a singer performs a rag (melodic framework) accompanied by a drone instrument like the tanpura. Over centuries, many great singers have developed unique styles passed down generations. Check out this captivating clip featuring renowned vocalist Kishori Amonkar performing a raga based on love.</p> <p>Section 3: Spirituals \& Gospel - From Slaves to Freedom Fighters---During the dark period of slavery in America (16th–19th centuries), enslaved Africans preserved their heritage through secretive gatherings filled with song and dance. Their spirituals often contained ...</p>

Figure 19 The sample of a doc containing the specific word selected by AttentionInfluence-1.3B (**left**) and FineWeb-Edu Classifier (**right**).

L High Frequency Words

As illustrated in Figure 18, we visualize the respective word clouds of AttentionInfluence-1.3B and the FineWeb-Edu Classifier after removing overlapping high-frequency words in the Cosmopeida-V2 domain. The resulting word clouds clearly highlight their distinct focal points, indicating a complementary relationship between the two models. To gain deeper insights, we further examine representative samples corresponding to the key terms in each word cloud.

M Limitations and Opportunities

While our experimental results demonstrate the effectiveness of AttentionInfluence, several important aspects warrant further investigation. We identify five key areas for future research:

- Our current experiments demonstrate the effectiveness of AttentionInfluence up to 7B parameters and 1,000B tokens of training budget. Extending this approach to long-horizon training and larger-scale models requires a highly expensive computational cost, and we leave it for future research.
- Due to limited manpower, we do not investigate the effects of selected data by AttentionInfluence on the final performance of models, followed by post-training. We hypothesize that reinforcement learning will amplify the good effects of selected data by AttentionInfluence.
- While this work focuses on selecting data from short texts, AttentionInfluence can be readily extended to long texts to identify high-quality samples characterized by long-range dependencies.
- We conduct experiments with alternative approaches for identifying important attention heads, such as the method proposed by Fu et al. [11], which produces a partially overlapping yet distinct set of heads compared to ours. Training LLMs based on the data selected by these heads can achieve comparable downstream evaluation performance. More recently, Zhu et al. [55] introduces another compatible method that can be incorporated into our framework.

These results demonstrate that AttentionInfluence serves as a flexible and general framework: by defining an appropriate proxy task, one can identify task-relevant attention heads and select associated data via masking. The entire pipeline operates without any supervision signals and is modular by design, allowing the proxy task to be easily replaced depending on the target domain or task. Moreover, the framework is effective even when applied to small pretrained language models, making it practical and scalable for a wide range of data selection scenarios.

More comprehensive proxy tasks can also be designed to better capture specific types of data within the AttentionInfluence framework, further expanding its applicability and customization potential.

- The combined effect of multiple heads remains unknown. Moreover, this work does not involve research on the MLP. Substantially more in-depth research endeavors are required to unearth the more fundamental and intrinsic mechanisms underpinning language models.