

Spectrum Projection Score: Aligning Retrieved Summaries with Reader Models in Retrieval-Augmented Generation

Zhanghao Hu¹, Qinglin Zhu¹, Siya Qi¹, Yulan He^{1,2}, Hanqi Yan¹, Lin Gui¹

¹King’s College London ²The Alan Turing Institute

{zhanghao.hu, qinglin.l.zhu, siya.qi}@kcl.ac.uk

{yulan.he, hanqi.yan, lin.l.gui}@kcl.ac.uk

Abstract

Large Language Models (LLMs) have shown improved generation performance through retrieval-augmented generation (RAG) following the retriever-reader paradigm, which supplements model inputs with externally retrieved knowledge. However, prior work often evaluates RAG holistically, assessing the retriever and reader jointly, making it difficult to isolate the true contribution of retrieval, particularly given the prompt sensitivity of LLMs used as readers. We introduce Spectrum Projection Score (SPS), a lightweight, supervision-free metric that allows the reader to gauge the semantic alignment of a retrieved summary with its hidden representation by comparing the area formed by generated tokens from the summary, and the principal directions of subspace in the reader and to measure the relevance. Building on SPS we present xCompress, an inference-time controller framework that dynamically samples, ranks, and compresses retrieval summary candidates. Extensive experiments on five QA benchmarks with four open source LLMs show that SPS not only enhances performance across a range of tasks but also provides a principled perspective on the interaction between retrieval and generation.

1 Introduction

Large-context Retrieval-Augmented Generation (RAG) has demonstrated promising capabilities in addressing open-domain question answering tasks (Wang et al. 2024; Izacard et al. 2023). In the standard pipeline, a retriever locates and compresses external evidence with a compressor language model, and a reader generates the final answer from the compressed summary (Mialon et al. 2023). A central challenge is to evaluate whether a given summary will actually help the reader answer the question (Shi et al. 2023), particularly given the reader’s sensitivity to summary variations.

Therefore, a metric that measures the compatibility of the reader with different input summaries is crucial for better compressed summary generation. Existing measurements, such as token-level perplexity and its long-context variants, or on embedding similarity computed with mean pooling (Zhang et al. 2025; Liu et al. 2025; Chen et al. 2024), primarily assess how typical a token sequence is under a language model. To demonstrate the effects of these measurements, we scatter all tokens from one summary in the reader’s embedding space with t-SNE (Figure 1). For this summary, we

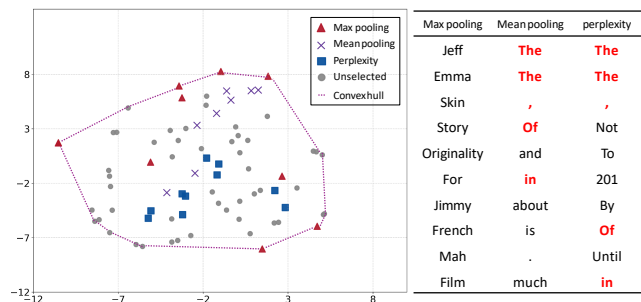


Figure 1: Token selection in the reader’s embedding space. We project summary’s token embeddings with t-SNE and compare three selections: nearest to the mean-pooled vector, highest predictive probability, and contributors to max pooling. Mean pooling and perplexity concentrate near the center and favour syntactically frequent tokens. Max pooling emphasises boundary tokens near the convex hull that carry salient semantics.

highlight token embeddings with different selection methods: tokens nearest to the mean-pooled vector, tokens chosen by perplexity-based scoring with the highest predictive probability, and tokens that contribute to max pooling as per-dimension maxima. Perplexity and mean pooling favour centrally clustered, low-content tokens (e.g., *the*, *of*, *“”*); by contrast, max pooling surfaces boundary tokens near the convex hull with substantive meaning (e.g., *Jeff*, *Emma*, *French*). This evidence suggests that representing a sentence by a single centroid or by the most probable tokens fails to capture the shape of the information carried by the sequence.

Given the observation above, we argue that these token-level predictive probability-related embeddings not be able to fully represent the semantic meaning of a text segment, which aims to project text into only one single point in the space and ignore the shape of the distribution. Instead, it should be captured by the collective “area” covered by the embeddings of all tokens in the sequence. However, two primary challenges arise when adopting this “area-base” approach: 1) *Defining the Area*: Formally defining the semantic “area” in high-dimensional embedding space is non-trivial. The embedding space is vast, and computing structures like the convex hull that encapsulates all token em-

beddings can be computationally expensive. 2) *Measuring the Area*: Even if a well-defined area is obtained, measuring its shape and size accurately, which is defined in high-dimensional space, is complex due to potential non-convexity and irregular geometry.

To address these challenges, we propose a novel evaluation metric, the **Spectrum Projection Score (SPS)**, starting from the concepts of convex hulls and partial order theory, which have been insufficiently explored despite widespread study of max-pooling. SPS leverages max-pooling across token embeddings from the retriever to approximate a semantic “area” and applies PCA to identify principal semantic directions (spectrum directions). By aligning this semantic area from the retriever with directions derived from the reader’s internal embedding space, even when retriever and reader models differ, SPS quantifies the alignment between the summary embeddings and the reader’s representation, offering a principled measure of semantic confidence.

Building upon SPS, we introduce **xCompress**, an effective framework that incorporates SPS into test-time sampling strategies. The framework adaptively selects text summaries or embedding summaries optimally aligned with the reader’s parameter space and improves retrieval utility. Furthermore, we enhance efficiency through an adaptive norm-guided filtering strategy, dynamically determining the necessity of sampling for each query, thus maintaining generation quality while reducing computational overhead. We evaluate SPS on five Open Question Answer (Open-QA) datasets using four different large language models. Experimental results show that SPS consistently outperforms existing evaluation baselines across most settings. The main contributions of this paper are as follows:

- We propose the Spectrum Projection Score (SPS), a training-free metric that measures summary–reader alignment by projecting a max-pooled envelope of token embeddings onto the reader’s principal subspace and using the residual norm as the score.
- We present xCompress, an inference-time controller that samples candidate summaries, ranks them with SPS to select those best aligned with the reader, and applies an adaptive norm-guided filter to control computation.
- We empirically validate SPS across five datasets and four state-of-the-art open-sourced LLMs, demonstrating superior performance over established baselines.

2 Related Work

Summary Compression in RAG. Recent work on RAG summary focuses on condensing retrieved content into query-relevant representations, primarily through text-to-text summarisation or text-to-embedding conversion (Wang et al. 2023; Li et al. 2024b; Ke et al. 2024). Text-to-text approaches generate concise summaries through models trained to distil knowledge from larger language models (Yoon et al. 2024; Xu, Shi, and Choi 2024). Alternatively, text-to-embedding methods, such as xRAG (Cheng et al. 2024a), directly convert retrieved passages into embeddings, concatenating them with query embeddings before processing by the reader. These approaches often yield

suboptimal alignment between compressed contexts and the downstream reader’s internal representations due to inherent discrepancies in model-specific embedding spaces. In contrast, our framework xCompress introduces an inference-time, training-free strategy that aligns retrieval summaries with the reader’s semantic space via SPS.

Perplexity-based Metrics for Text Assessment. Existing evaluation methods for retrieval-based generation predominantly utilise entropy- or perplexity-based metrics, assessing how well a language model predicts tokens given their preceding context (Liu et al. 2025; Yu et al. 2025; Xu et al. 2025). Despite their intuitive appeal, these metrics exhibit fundamental limitations. Primarily, perplexity is sensitive to sequence length (Wang et al. 2022), often emphasising predictable but semantically trivial tokens. Consequently, perplexity-based methods inadequately capture the semantic coherence and relevance critical to retrieval-based question-answering (Agarwal et al. 2024; Li et al. 2024a; Fang et al. 2025). In contrast, SPS evaluates summaries by aligning their semantic distribution with the reader’s embedding space using max pooling and convex hull theory, thereby emphasising boundary tokens that carry greater semantic relevance over trivial ones.

3 Preliminary: Summarise retrieval passages to align with the reader

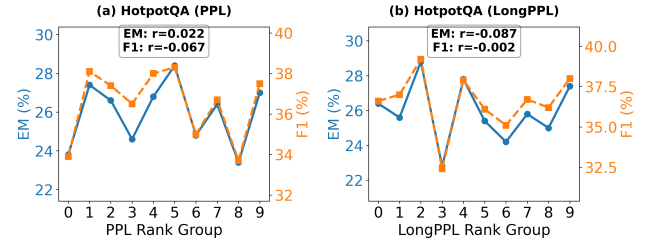


Figure 2: RAG task performances (measured by EM and F1) when feeding summaries with varying PPL (left) and LongPPL (right) to the Reader on the HotpotQA dataset. The low Pearson correlation coefficients (r) indicate that both PPL and LongPPL fail to identify a good summary.

Retrieval-based generation typically follows a retriever–reader pipeline: documents are retrieved, summarised, and then provided to a generative reader for answer production (Lewis et al. 2020; Izacard et al. 2022a). While summarisation condenses input and highlights salient content (Yoon et al. 2024; Cheng et al. 2024b), prior work has largely overlooked how well a summary *aligns with the reader model’s internal representation space*. We therefore focus on the following question: *how can we measure the quality of retrieved summaries from the perspective of their compatibility with the reader?*

A natural proxy is perplexity (PPL), a monotonic transform of sequence likelihood under the reader. Lower PPL indicates that the reader deems a sequence more “typical”

within its internal language space, suggesting better compatibility. Formally, for a summary token sequence $x = (x_1, \dots, x_n)$ and a reader parameterised by P_θ , we compute:

$$\text{PPL}_\theta(x) = \exp\left(-\frac{1}{n} \sum_{i=1}^n \log P_\theta(x_i | x_{<i})\right). \quad (1)$$

However, our analyses (Figure 2a) show a weak association between PPL and downstream QA performance when PPL is used to rank summaries. A likely cause is that PPL is highly sensitive to length and stylistic factors (Wang et al. 2022), which becomes particularly problematic in the long-context settings common in retrieval-based generation (Agarwal et al. 2024; Li et al. 2024a). We further evaluate a strengthened likelihood-based metric, LongPPL (Fang et al. 2025), which discounts length and emphasises key tokens in extended contexts. Although LongPPL mitigates some issues, it still exhibits limited correlation with QA performance (Figure 2b).

These observations suggest a structural misalignment: token-level log-likelihood primarily captures *typicality* rather than whether a summary’s *salient semantics* map onto directions that the reader readily encodes and can exploit for answering the query. Related efforts like SePer (Dai et al. 2025a) seek to assess retrieval utility but rely on human preference signals and do not explicitly model representation-level alignment to task performance.

In contrast to log-likelihood-based metrics like perplexity, we propose the Spectrum Projection Score, a representation-level measure that directly targets *summary–reader compatibility*. Rather than judging a summary by how “typical” it appears token-by-token, Spectrum Projection Score evaluates how the summary’s salient representation aligns with the reader’s internal geometry: we form a **bounder** vector from max-pooled token states (to retain boundary, content-bearing features) and assess its alignment to the reader’s principal subspace derived from the model’s parameters or hidden representations. The motivation is that: 1) this bounder vector aims to estimate the **Essential Supremum**, 2) the bounder vector converges to a distribution-specific property. 3) Thus, this property of consistent estimation allows the boundary vector to be used as a robust tool to determine if the boundary areas of two different generators, for example, the retriever and the reader in this task, are aligned¹. In general, this combines sentence-level projection with salient token-level cues, yielding a simple, training-free score that better reflects the utility of retrieved summaries for generation (formalised in Section 4.2).

4 Methodology

4.1 Overview and Problem Description

Overview. We propose xCompress, a retrieval-time controller that *evaluates and compresses* retrieved content so it better aligns with the reader LLM (Figure 3). In Section 4.2, we introduce Spectrum Projection Score, a sim-

¹Due to the length limitation, we detailed the theoretical analysis, including the formal definition and convergence analysis, in Appendix B.

ple, training-free metric that replaces perplexity-style scoring by assessing summary quality inside the reader’s representation space. In Section 4.3, we describe a lightweight test-time sampling that explores both *text-to-text* and *text-to-embedding* compression, ranks candidates with Spectrum Projection Score, and uses an adaptive filtering to decide whether further sampling is needed.

Problem Description. We consider the retriever–reader pipeline (Chen et al. 2017; Lee, Chang, and Toutanova 2019) for retrieval-based generation. Given a query q and a corpus \mathcal{D} , the retriever returns relevant passages \mathcal{B} . The top- N passages are compressed into summaries, either in textual (text-to-text) or embedding (text-to-embedding). Each summary candidate is embedded via the reader’s penultimate-layer representations, which are subsequently max-pooled. A norm-guided filter then determines if further candidate sampling is necessary. If sampling is triggered, additional summary candidates are generated. Each summary is evaluated by the Spectrum Projection Score (SPS), which measures alignment between the summary’s representation and the reader’s principal embedding subspace. Finally, the candidate with the lowest SPS, or the initial summary if sampling is skipped, is provided to the reader LLM for answer generation.

4.2 Spectrum Projection Score: Measuring Alignment with the Reader LLM

Retrieval-based generation compresses retrieved documents before feeding them into a reader LLM to reduce context length and foreground salient content (Lewis et al. 2020; Izacard et al. 2022a; Yoon et al. 2024; Cheng et al. 2024b). However, entropy- or perplexity-based evaluations are length-biased and only weakly correlated with downstream performance (Wang et al. 2022; Agarwal et al. 2024; Li et al. 2024a; Dai et al. 2025a), as they capture *typicality* rather than whether a summary’s salient semantics are well represented by the reader’s internal geometry. We therefore seek a representation-level metric that scores a summary by its *compatibility* with the reader.

We introduce the *Spectrum Projection Score* (SPS), which quantifies how well a compressed summary aligns with the reader’s principal representational directions. Let the reader’s representation space be characterised by a matrix $W \in \mathbb{R}^{D \times M}$ (e.g., the input embedding matrix or a bank of hidden states collected from the reader, where $D \times M$ is the corresponding matrix size). We first identify the principal subspace of W through PCA. Specifically, we apply Singular Value Decomposition (SVD): $W = U\Sigma V^\top$, where U and V are left and right singular matrices, $\|\cdot\|^\top$ is the matrix transpose operation, and Σ is the singular value matrix. To retain the principal components by selecting the top 95% eigenvalues in Σ , noted as Σ_p , and reconstruct the projection by $P = U\Sigma_p V^\top$, to obtain the reader’s core subspace.

Given a retrieval summary, we pass it through the reader and obtain token representations from the penultimate layer; we then apply elementwise max pooling over tokens to form a *salient* summary vector $\mathbf{x} \in \mathbb{R}^D$ that preserves boundary features (entities, fact-bearing nouns/adjectives) rather than

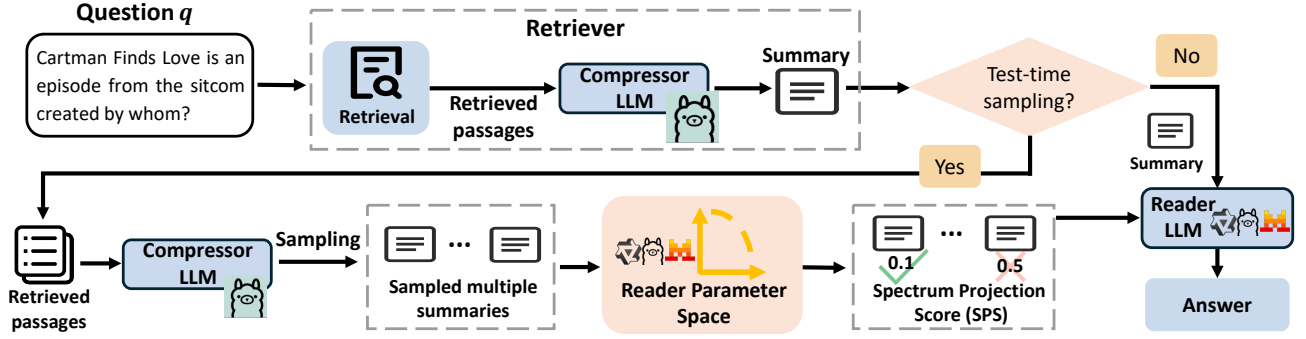


Figure 3: Overview of the xCompress framework. Retrieved passages are first compressed into summaries. An adaptive norm-guided filtering mechanism determines whether additional test-time sampling is necessary. If required, multiple summaries are sampled from the compressor LLM and evaluated using the Spectrum Projection Score (SPS). These summaries are first embedded via max-pooling, then projected onto the reader’s principal subspace of its parameter. The summary with the lowest SPS is selected as input to the reader; otherwise, the initial summary is used directly for answer generation.

averaging them away. We define

$$\text{SPS}(\mathbf{x}) = \|(I - P)\mathbf{x}\|_2, \quad (2)$$

where I is an identity matrix.

Intuitively, this measures how much of the salient summary vector \mathbf{x} is captured by the reader’s principal subspace: $P\mathbf{x}$ is the in-subspace component, and the residual $(I - P)\mathbf{x}$ quantifies what lies outside. Hence, a smaller $\text{SPS}(\mathbf{x})$ indicates stronger alignment between the summary and the reader’s core representational geometry, making the summary easier for the reader to generate.

4.3 Test-time Sampling with Spectrum Projection Score

Retrieval-based generation typically follows a sequential retriever–reader pipeline (Hu et al. 2025), where an auxiliary LLM compresses retrieved passages into summaries, either textual (text-to-text) (Yoon et al. 2024) or embedding-based (text-to-embedding) (Cheng et al. 2024a), which are then provided to the reader for answer generation. A limitation of this unidirectional flow is that compression is performed without regard to the reader’s internal representational geometry. We instead use the reader’s own representation space to guide compression at test time: we generate a set of summary candidate compressions and score each with Spectrum Projection Score (SPS), selecting the summary that best aligns with the reader.

Test-time Sampling in Text-to-text Compression: In the text-to-text compression paradigm, summaries are typically generated by a compressor LLM before being passed to the reader. To better align these compressed summaries with the reader’s embedding space, we propose leveraging the reader’s parameters within our SPS metric. Specifically, considering output diversity while maintaining coherence, we adopt stochastic decoding, rather than deterministic methods like greedy or beam search, to produce K diverse summary candidates for each query. Each candidate summary is

then evaluated by computing its SPS using the reader’s embedding parameters. Finally, the summary with the lowest SPS, indicating optimal alignment with the reader’s internal representation, is selected as input to the reader LLM for downstream answer generation.

Test-time Sampling in Text-to-embedding Compression.

Embedding-level compression maps retrieved passages (and the query) directly to a summary embedding via a trained projector (Cheng et al. 2024a). Because this mapping is deterministic, it offers no native mechanism for sampling diverse candidates. Inspired by soft reasoning with injected noise (Hu et al. 2025), we introduce *probe-based* stochasticity at test time. Concretely, we sample N small Gaussian probe vectors $\{\mathbf{e}_r\}_{r=1}^N$ and append each to the summary–query embedding before passing the fusion representation through the reader LLM. For each probe, we extract the reader’s penultimate-layer hidden state at the probe position, denoted \mathbf{h}_r , and compute a simple diversity score following (Hu et al. 2025):

$$S_{\text{probe}} = \sum_{i=1}^p (\Delta_{(i)})^2, \quad (3)$$

where $\Delta_{(i)}$ is the gap between the i -th and $(i+1)$ -th largest elements of \mathbf{h}_r . Smaller S_{probe} indicates stronger semantic deviation from the existing summary–query signal. We retain the M probes with the smallest scores and form $M+1$ candidate embedding summaries (the original plus M probed variants). As in the text-to-text case, each candidate is scored with Spectrum Projection Score using the reader’s representation space, and the embedding summary with the lowest SPS is selected for answer generation.

Adaptive Sampling via Norm-guided Filtering. Sampling multiple candidates improves alignment but is expensive if applied universally. We therefore add a lightweight filter that decides whether further sampling is needed. For the initial summary (text-to-text or text-to-embedding), we compute two proxies in the reader’s latent space: the L2

norm of the *mean-pooled* representation $L2_{mean}$ (captures overall mass) and the L1 norm of the max-pooled representation $L1_{max}$ (captures salient peaks). Their ratio $L2_{mean}/L1_{max}$ serves as a concentration and stability indicator: higher values indicate that the information follows a more skewed distribution, suggesting that the summary is less likely to benefit from additional sampling. In contrast, lower values reflect a more sparse distribution, where further exploration of the alternatives summary may yield additional value². We estimate a threshold on the validation set through full sampling of all data. At inference, if the ratio exceeds the threshold, we accept the initial summary; otherwise, we perform the sampling-and-selection procedure guided by Spectrum Projection Score. This preserves most of the accuracy gains while substantially reducing computation.

5 Experiment

5.1 Setups.

Evaluation Dataset. We evaluate our framework on five retrieval-based QA benchmarks: HotpotQA (Yang et al. 2018), 2WikiMulti-hopQA (2Wiki) (Ho et al. 2020), Natural Questions (NQ) (Kwiatkowski et al. 2019), TriviaQA (TQA) (Joshi et al. 2017), and Musique. Evaluations are conducted on the development sets, except for TQA, which uses the test set. For NQ, we adopt the original test split with the 21M English Wikipedia dump (Karpukhin et al. 2020) as the retrieval corpus. Across all datasets, we follow the data splits and associated document corpora released by Kim et al. (2024) and Yoon et al. (2024).

Metrics. Following prior work (Chen et al. 2024; Dai et al. 2025b), we use the Area Under the Receiver Operating Characteristic curve (AUROC) and Pearson Correlation Coefficient (PCC) to assess the effectiveness of evaluation metrics. AUROC is widely applied to evaluate the measure of uncertainty estimation (Chen et al. 2024), with higher values indicating better discriminative ability. For retrieval-based generation task performance, we focus on open-domain question answering and report Exact Match (EM) and F1 scores. Following Rajpurkar et al. (2016), all predictions and gold answers are normalised by lowercasing and removing punctuation to ensure consistency.

Baseline and Models. We compare Spectrum Projection Score with perplexity, the most common uncertainty-based evaluation metric for large language model predictions (Ren et al. 2023), along with its variant, LongPPL, specifically designed to improve performance with long contexts (Fang et al. 2025). For retrieval summary compression in retrieval-based generation, we select one recent method from each compression paradigm: the text-to-text method CompAct (Yoon et al. 2024) and the text-to-embedding method xRag (Cheng et al. 2024a). Additionally, we evaluate various retrieval strategies:

- (1) *Raw Document*, which directly concatenates the top- k retrieved passages;

- (2) *Long-Context LLM Summary*, which uses LLMs to summarise retrieved passages before answer generation, following recent practices (Yoon et al. 2024).

Backbone models. we utilise four open-source LLMs: LLaMA-3.1-8B-Instruct (Grattafiori et al. 2024), Gemma3-12B-Instruct (Team et al. 2025), and Qwen3-8B (Yang et al. 2025) for the text-to-text paradigm and select Mistral 7B (Jiang et al. 2024) for the text-to-embedding paradigm since the reader model in the text-to-embedding method xRag (Cheng et al. 2024a) is specifically trained alongside its retriever projector, we directly adopt its original reader LLM.

Implementation Details. For retrieval, we adopt Contriever (Izacard et al. 2022b) via the BEIR toolkit (Thakur et al. 2021). Following Yoon et al. (2024), we retrieve the top-30 documents for fair comparison. In test-time sampling, we set the temperature to 1.0, apply a repetition penalty of 1.2, and generate five summaries per question to balance diversity and efficiency. For reader LLM generation, we use greedy decoding (temperature = 0.0) to eliminate randomness and ensure reproducibility (Sun et al. 2023)³. For norm-guided filtering, we empirically set the threshold as the top-30% value within the validation set.

5.2 Main Experiment

Dataset	Metric	PPL	LongPPL	SPS
HotpotQA	PCC (EM)	0.022	-0.087	0.643
	PCC (F1)	-0.067	-0.002	0.753
	AUROC	0.504	0.495	0.553
2Wiki	PCC (EM)	-0.318	-0.065	0.557
	PCC (F1)	0.295	0.269	0.503
	AUROC	0.487	0.482	0.565
NQ	PCC (EM)	0.202	0.281	0.650
	PCC (F1)	0.452	0.498	0.628
	AUROC	0.508	0.500	0.525
TQA	PCC (EM)	0.244	-0.083	0.563
	PCC (F1)	0.127	-0.210	0.432
	AUROC	0.497	0.478	0.531
Musique	PCC (EM)	0.182	-0.186	0.508
	PCC (F1)	0.094	0.008	0.505
	AUROC	0.443	0.488	0.504

Table 1: Pearson correlation coefficients (PCC) and AUROC for PPL, LongPPL, and SPS on RAG tasks across five datasets using the LLAMA-3.1-8B-Instruct model as the backbone. SPS consistently achieves the highest correlation with answer quality across all datasets, demonstrating its effectiveness in identifying high-quality summaries.

Effectiveness of Spectrum Projection Score. To evaluate the effectiveness of Spectrum Projection Score in correlating with performance on retrieval-based generation tasks, we conducted experiments comparing it against standard metrics, PPL and LongPPL. Specifically, we generated ten distinct summaries per query using fixed decoding parameters.

²Detailed theoretical discussion is in Appendix B.

³Prompt details in Appendix A.

Model	Method/Dataset	HotpotQA	2WikiMQA	Musique	NQ	TriviaQA
Text-to-Text						
Llama 3.1 8b Ins	Retrieval direct	19.6 / 28.85	9.4 / 18.22	2.0 / 7.57	17.4 / 29.58	49.4 / 57.64
	Compact	34.0 / 43.17	27.2 / 31.83	6.6 / 13.76	35.2 / 47.49	62.4 / 71.25
	xCompress + PPL	33.0 / 43.52	25.0 / 29.58	7.6 / 14.49	35.0 / 46.53	62.6 / 71.76
	xCompress + LongPPL	32.0 / 42.48	24.2 / 28.16	6.8 / 14.71	35.6 / 46.78	62.4 / 71.72
	xCompress + SPS(ours)	37.6 / 47.87	29.6 / 34.21	9.0 / 17.63	39.4 / 51.18	65.4 / 73.11
Qwen3 8b	Retrieval direct	21.4 / 32.29	12.2 / 22.19	3.8 / 12.49	17.8 / 27.70	51.8 / 59.64
	Compact	26.8 / 38.84	22.2 / 26.69	6.0 / 13.36	25.8 / 36.50	55.4 / 63.22
	xCompress + PPL	29.6 / 40.86	21.8 / 25.63	5.0 / 13.48	30.0 / 45.88	54.8 / 63.57
	xCompress + LongPPL	22.8 / 31.90	20.8 / 26.33	7.0 / 15.40	25.0 / 36.44	57.8 / 68.02
	xCompress + SPS(ours)	28.8 / 41.84	25.6 / 30.71	8.6 / 17.07	28.0 / 38.75	59.6 / 68.84
Gemma 3 12b Ins	Retrieval direct	10.8 / 16.47	3.4 / 6.49	1.2 / 3.92	16.6 / 26.15	27.4 / 37.27
	Compact	19.2 / 29.69	23.8 / 28.97	5.4 / 12.78	27.6 / 40.86	52.8 / 64.44
	xCompress + PPL	19.4 / 31.56	23.0 / 28.46	5.4 / 13.93	28.8 / 39.07	52.0 / 64.10
	xCompress + LongPPL	22.6 / 33.82	18.2 / 23.70	3.8 / 10.93	29.4 / 40.90	53.0 / 64.06
	xCompress + SPS(ours)	25.2 / 35.66	25.0 / 29.31	6.4 / 14.60	31.6 / 42.27	57.4 / 65.39
Text-to-Embedding						
Mistral-7b	Retrieval direct	1.0 / 8.23	1.2 / 11.73	0.2 / 3.41	1.0 / 5.53	2.0 / 14.91
	xRAG	5.2 / 16.63	2.2 / 14.09	0.4 / 5.69	3.0 / 13.56	16.0 / 40.09
	xCompress + SPS(ours)	7.6 / 20.06	2.8 / 15.82	0.6 / 6.15	3.8 / 17.70	29.2 / 46.76

Table 2: EM / F1 (%) scores across five QA benchmarks using different retrieval and summarisation strategies. xCompress with SPS consistently achieves the best performance across models and datasets, demonstrating its effectiveness over perplexity-based metrics and baseline methods in both text-to-text and text-to-embedding paradigms.

Each summary was then independently scored using PPL, LongPPL, and our Spectrum Projection Score. To ensure fair comparison, summaries for each query were ranked according to these metric scores, and subsequently grouped into ten ordered bins. We then measured the downstream retrieval-based generation task performance (Exact Match [EM] and F1) for each bin. The correlation between bin rankings and corresponding task performance was quantified using the Pearson Correlation Coefficient (PCC).

Additionally, to further quantify metric discriminative capability, we computed the AUROC scores based on binary correctness (EM=1 as positive, EM=0 as negative). Each pairwise comparison between positive and negative summaries for a given query was used to evaluate whether the metrics correctly identified the better summary or not.

Table 1 demonstrates that both PPL and LongPPL have poor correlation with downstream task performance. Conversely, our Spectrum Projection Score consistently shows significantly stronger correlations, indicating superior effectiveness in distinguishing summary quality relevant to retrieval-based generation.

Effectiveness of xCompress. Table 2 summarises the effectiveness of our proposed xCompress framework across five QA datasets and two retrieval-based generation paradigms (text-to-text and text-to-embedding). Results demonstrate that incorporating our Spectrum Projection Score (SPS) consistently improves downstream Exact Match (EM) and F1 scores over the baseline compression meth-

ods. For instance, on the NQ dataset using the LLAMA 3.1 model, SPS improves performance from 35.2/47.49 (EM/F1) to 39.4/51.18, highlighting SPS’s effectiveness in selecting retrieval summaries that align better with the reader model’s internal representations. Our framework consistently outperforms perplexity-based (PPL and LongPPL) methods, except in two cases involving the Qwen3 model on the HotpotQA and NQ datasets. In these instances, selecting summaries based on PPL yielded slightly better performance than SPS. We attribute this to the Qwen model’s overconfidence on these datasets, possibly due to substantial overlap between its pretraining data and these evaluation datasets, a hypothesis supported by recent literature suggesting dataset contamination or repeated exposure during training (Wu et al. 2025). Further, empirical analysis revealed significantly lower entropy, approximately three times lower, in Qwen’s predictions on HotpotQA and NQ compared to LLAMA 3.1, reinforcing that Qwen likely recalls answers directly from memorised content⁴. Consequently, summaries chosen via perplexity reflect familiar, memorised content rather than optimal alignment, paradoxically enhancing performance but reducing generalisability.

6 Analysis

6.1 Why Max Pooling Yields Superior Results?

We investigate the impact of different sentence embedding extraction methods, max pooling, mean pooling, and last-

⁴Experiment result details are in Appendix C.

token pooling, on our Spectrum Projection Score (SPS) performance. Table 3 shows that max pooling consistently achieves superior EM/F1 scores across multiple retrieval-based generation datasets. This suggests max pooling effectively captures salient semantic tokens, whereas mean pooling dilutes semantic signals by averaging, and last-token pooling disproportionately emphasises sentence-end tokens. Hence, max pooling yields embeddings with richer semantic content and better alignment with the LLM’s embedding space, ultimately enhancing downstream task performance.

Dataset	Max Pooling	Mean Pooling	Last Token
HotpotQA	37.6 / 47.87	36.2 / 47.65	33.6 / 43.37
2WikiMQA	29.8 / 34.21	28.2 / 33.63	23.6 / 28.10
Musique	9.2 / 17.63	7.8 / 15.63	5.8 / 10.49
NQ	39.6 / 51.18	37.2 / 49.36	35.8 / 47.58
TriviaQA	65.6 / 73.11	64.2 / 72.62	63.0 / 72.24

Table 3: EM / F1 scores (%) of different pooling strategies for sentence embedding extraction with LLAMA 3.1 across datasets. Max pooling consistently achieves the best.

6.2 How Spectrum Projection Score Performs with Different Sentence Embeddings?

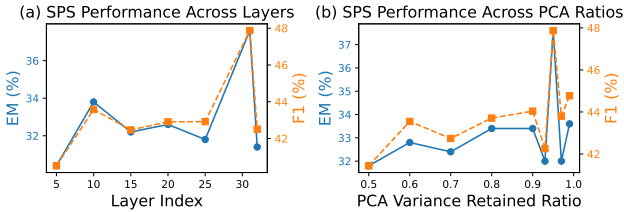


Figure 4: SPS performance under (a) Across LLM layers. (b) Varying PCA retained variance ratios. Optimal results are achieved using embeddings from the penultimate layer and a PCA variance ratio of 0.95.

Different Layers. Building on the use of max-pooled sentence embeddings, we examine how SPS behaves across different model layers. As illustrated in Figure 4 (a), embeddings derived from the penultimate layer consistently yield superior downstream performance compared to embeddings from shallower or the last layers. Specifically, embeddings from earlier layers lack the high-level semantic abstraction necessary for effectively aligning summaries with the model’s embedding space, whereas embeddings from the final layer tend to be overly specialised toward token prediction, diminishing their general semantic representativeness. These results empirically confirm that embeddings from the penultimate layer optimally balance semantic abstraction and contextual generalisation, enhancing retrieval quality assessment and thus improving downstream retrieval-based generation performance.

Different PCA Ratio. We further examine how the variance ratio retained in PCA affects the performance of our

Spectrum Projection Score (SPS). As depicted in Figure 4 (b), performance (EM and F1 scores) peaks when the retained PCA variance ratio is set to 0.95. Lower variance ratios (e.g., 0.50–0.90) fail to preserve sufficient semantic information, resulting in degraded downstream performance. Conversely, excessively high variance ratios (e.g., 0.99) tend to include redundant or noisy dimensions, slightly diluting the semantic representativeness crucial for effective retrieval-summary alignment. Empirically, retaining 95% variance achieves an optimal balance between semantic richness and dimensional efficiency.

6.3 Number of Generation Influence.

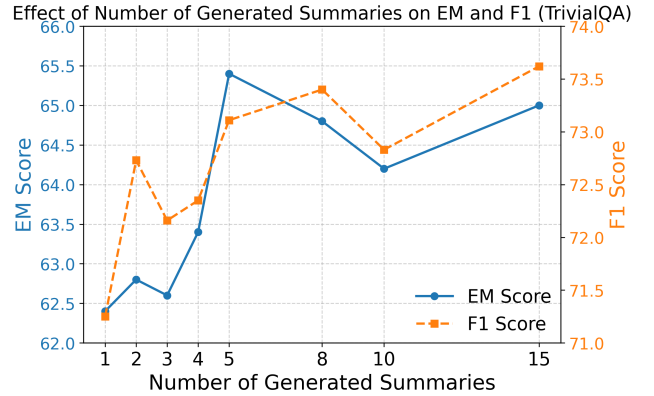


Figure 5: Impact of the number of generated summaries on EM and F1 scores TrivialQA. Performance saturates at five summaries, providing an optimal balance between effectiveness and computational efficiency.

We analyse how the number of generated summaries influences the effectiveness of our proposed Spectrum Projection Score (SPS). As shown in Figure 5, xCompress surpasses the Compact baseline (Yoon et al. 2024) (EM 62.4, F1 71.25) by generating just two summaries, highlighting immediate gains from summary diversity. Performance improves notably up to five summaries (EM 65.4, F1 73.11), beyond which additional summaries yield marginal benefits. Thus, we select five summaries as the optimal balance between performance and computational efficiency.

7 Conclusion

We propose the Spectrum Projection Score (SPS), a training-free, representation-level metric that evaluates the semantic alignment between retrieved summaries and the reader model’s internal geometry. Building on SPS, we introduce xCompress, an inference-time controller that guides summary selection through test-time sampling and adaptive filtering. Extensive experiments across five QA benchmarks and multiple LLMs demonstrate that SPS outperforms perplexity-based baselines in both correlation with answer quality and downstream task performance.

References

- Agarwal, R.; Singh, A.; Zhang, L.; Bohnet, B.; Rosias, L.; Chan, S.; Zhang, B.; Anand, A.; Abbas, Z.; Nova, A.; et al. 2024. Many-shot in-context learning. *Advances in Neural Information Processing Systems*, 37: 76930–76966.
- Chen, C.; Liu, K.; Chen, Z.; Gu, Y.; Wu, Y.; Tao, M.; Fu, Z.; and Ye, J. 2024. INSIDE: LLMs’ Internal States Retain the Power of Hallucination Detection. In *The Twelfth International Conference on Learning Representations*.
- Chen, D.; Fisch, A.; Weston, J.; and Bordes, A. 2017. Reading Wikipedia to Answer Open-Domain Questions. In Barzilay, R.; and Kan, M.-Y., eds., *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, 1870–1879. Vancouver, Canada: Association for Computational Linguistics.
- Cheng, X.; Wang, X.; Zhang, X.; Ge, T.; Chen, S.-Q.; Wei, F.; Zhang, H.; and Zhao, D. 2024a. xRAG: Extreme Context Compression for Retrieval-augmented Generation with One Token. In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*.
- Cheng, X.; Wang, X.; Zhang, X.; Ge, T.; Chen, S.-Q.; Wei, F.; Zhang, H.; and Zhao, D. 2024b. xRAG: Extreme Context Compression for Retrieval-augmented Generation with One Token. In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*.
- Dai, L.; Xu, Y.; Ye, J.; Liu, H.; and Xiong, H. 2025a. SePer: Measure Retrieval Utility Through The Lens Of Semantic Perplexity Reduction. In *The Thirteenth International Conference on Learning Representations*.
- Dai, L.; Xu, Y.; Ye, J.; Liu, H.; and Xiong, H. 2025b. SePer: Measure Retrieval Utility Through The Lens Of Semantic Perplexity Reduction. In *The Thirteenth International Conference on Learning Representations*.
- Fang, L.; Wang, Y.; Liu, Z.; Zhang, C.; Jegelka, S.; Gao, J.; Ding, B.; and Wang, Y. 2025. What is Wrong with Perplexity for Long-context Language Modeling? In *The Thirteenth International Conference on Learning Representations*.
- Grattafiori, A.; Dubey, A.; Jauhri, A.; Pandey, A.; Kadian, A.; Al-Dahle, A.; Letman, A.; Mathur, A.; Schelten, A.; Vaughan, A.; et al. 2024. The llama 3 herd of models. *arXiv preprint arXiv:2407.21783*.
- Ho, X.; Duong Nguyen, A.-K.; Sugawara, S.; and Aizawa, A. 2020. Constructing A Multi-hop QA Dataset for Comprehensive Evaluation of Reasoning Steps. In Scott, D.; Bel, N.; and Zong, C., eds., *Proceedings of the 28th International Conference on Computational Linguistics*, 6609–6625. Barcelona, Spain (Online): International Committee on Computational Linguistics.
- Hu, Z.; Yan, H.; Zhu, Q.; Shen, Z.; He, Y.; and Gui, L. 2025. Beyond Prompting: An Efficient Embedding Framework for Open-Domain Question Answering. *arXiv preprint arXiv:2503.01606*.
- Izacard, G.; Caron, M.; Hosseini, L.; Riedel, S.; Bojanowski, P.; Joulin, A.; and Grave, E. 2022a. Unsupervised Dense Information Retrieval with Contrastive Learning. *Transactions on Machine Learning Research*.
- Izacard, G.; Caron, M.; Hosseini, L.; Riedel, S.; Bojanowski, P.; Joulin, A.; and Grave, E. 2022b. Unsupervised Dense Information Retrieval with Contrastive Learning. *Transactions on Machine Learning Research*.
- Izacard, G.; Lewis, P.; Lomeli, M.; Hosseini, L.; Petroni, F.; Schick, T.; Dwivedi-Yu, J.; Joulin, A.; Riedel, S.; and Grave, E. 2023. Atlas: Few-shot Learning with Retrieval Augmented Language Models. *Journal of Machine Learning Research*, 24(251): 1–43.
- Jiang, A. Q.; Sablayrolles, A.; Roux, A.; Mensch, A.; Savary, B.; Bamford, C.; Chaplot, D. S.; Casas, D. d. l.; Hanna, E. B.; Bressand, F.; et al. 2024. Mixtral of experts. *arXiv preprint arXiv:2401.04088*.
- Joshi, M.; Choi, E.; Weld, D.; and Zettlemoyer, L. 2017. TriviaQA: A Large Scale Distantly Supervised Challenge Dataset for Reading Comprehension. In Barzilay, R.; and Kan, M.-Y., eds., *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, 1601–1611. Vancouver, Canada: Association for Computational Linguistics.
- Karpukhin, V.; Oguz, B.; Min, S.; Lewis, P.; Wu, L.; Edunov, S.; Chen, D.; and Yih, W.-t. 2020. Dense Passage Retrieval for Open-Domain Question Answering. In Webber, B.; Cohn, T.; He, Y.; and Liu, Y., eds., *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 6769–6781. Online: Association for Computational Linguistics.
- Karvonen, A.; Rager, C.; Lin, J.; Tigges, C.; Bloom, J.; Chanin, D.; Lau, Y.-T.; Farrell, E.; McDougall, C.; Ayonrinde, K.; et al. 2025. Saebench: A comprehensive benchmark for sparse autoencoders in language model interpretability. *arXiv preprint arXiv:2503.09532*.
- Ke, Z.; Kong, W.; Li, C.; Zhang, M.; Mei, Q.; and Bendersky, M. 2024. Bridging the Preference Gap between Retrievers and LLMs. In Ku, L.-W.; Martins, A.; and Srikumar, V., eds., *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, 10438–10451. Bangkok, Thailand: Association for Computational Linguistics.
- Kim, J.; Nam, J.; Mo, S.; Park, J.; Lee, S.-W.; Seo, M.; Ha, J.-W.; and Shin, J. 2024. SuRe: Summarizing Retrievals using Answer Candidates for Open-domain QA of LLMs. In *The Twelfth International Conference on Learning Representations*.
- Kwiatkowski, T.; Palomaki, J.; Redfield, O.; Collins, M.; Parikh, A.; Alberti, C.; Epstein, D.; Polosukhin, I.; Devlin, J.; Lee, K.; Toutanova, K.; Jones, L.; Kelcey, M.; Chang, M.-W.; Dai, A. M.; Uszkoreit, J.; Le, Q.; and Petrov, S. 2019. Natural Questions: A Benchmark for Question Answering Research. *Transactions of the Association for Computational Linguistics*, 7: 452–466.
- Lee, K.; Chang, M.-W.; and Toutanova, K. 2019. Latent Retrieval for Weakly Supervised Open Domain Question Answering. In Korhonen, A.; Traum, D.; and Màrquez, L., eds., *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, 6086–6096. Florence, Italy: Association for Computational Linguistics.

- Lewis, P.; Perez, E.; Piktus, A.; Petroni, F.; Karpukhin, V.; Goyal, N.; Küttler, H.; Lewis, M.; Yih, W.-t.; Rocktäschel, T.; et al. 2020. Retrieval-augmented generation for knowledge-intensive nlp tasks. *Advances in neural information processing systems*, 33: 9459–9474.
- Li, T.; Zhang, G.; Do, Q. D.; Yue, X.; and Chen, W. 2024a. Long-context llms struggle with long in-context learning. *arXiv preprint arXiv:2404.02060*.
- Li, Z.; Hu, X.; Liu, A.; Zheng, K.; Huang, S.; and Xiong, H. 2024b. *Refiner*: Restructure Retrieved Content Efficiently to Advance Question-Answering Capabilities. In Al-Onaizan, Y.; Bansal, M.; and Chen, Y.-N., eds., *Findings of the Association for Computational Linguistics: EMNLP 2024*, 8548–8572. Miami, Florida, USA: Association for Computational Linguistics.
- Liu, W.; Qi, S.; Wang, X.; Qian, C.; Du, Y.; and He, Y. 2025. NOVER: Incentive Training for Language Models via Verifier-Free Reinforcement Learning. *arXiv preprint arXiv:2505.16022*.
- Mialon, G.; Dessi, R.; Lomeli, M.; Nalmpantis, C.; Pasunuru, R.; Raileanu, R.; Roziere, B.; Schick, T.; Dwivedi-Yu, J.; Celikyilmaz, A.; Grave, E.; LeCun, Y.; and Scialom, T. 2023. Augmented Language Models: a Survey. *Transactions on Machine Learning Research*. Survey Certification.
- Ng, A.; et al. 2011. Sparse autoencoder. *CS294A Lecture notes*, 72(2011): 1–19.
- Rajpurkar, P.; Zhang, J.; Lopyrev, K.; and Liang, P. 2016. SQuAD: 100,000+ Questions for Machine Comprehension of Text. In Su, J.; Duh, K.; and Carreras, X., eds., *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, 2383–2392. Austin, Texas: Association for Computational Linguistics.
- Ren, J.; Luo, J.; Zhao, Y.; Krishna, K.; Saleh, M.; Lakshminarayanan, B.; and Liu, P. J. 2023. Out-of-Distribution Detection and Selective Generation for Conditional Language Models. In *The Eleventh International Conference on Learning Representations*.
- Shi, F.; Chen, X.; Misra, K.; Scales, N.; Dohan, D.; Chi, E. H.; Schärli, N.; and Zhou, D. 2023. Large Language Models Can Be Easily Distracted by Irrelevant Context. In Krause, A.; Brunskill, E.; Cho, K.; Engelhardt, B.; Sabato, S.; and Scarlett, J., eds., *Proceedings of the 40th International Conference on Machine Learning*, volume 202 of *Proceedings of Machine Learning Research*, 31210–31227. PMLR.
- Sun, W.; Yan, L.; Ma, X.; Wang, S.; Ren, P.; Chen, Z.; Yin, D.; and Ren, Z. 2023. Is ChatGPT Good at Search? Investigating Large Language Models as Re-Ranking Agents. In Bouamor, H.; Pino, J.; and Bali, K., eds., *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, 14918–14937. Singapore: Association for Computational Linguistics.
- Team, G.; Kamath, A.; Ferret, J.; Pathak, S.; Vieillard, N.; Merhej, R.; Perrin, S.; Matejovicova, T.; Ramé, A.; Rivière, M.; et al. 2025. Gemma 3 technical report. *arXiv preprint arXiv:2503.19786*.
- Thakur, N.; Reimers, N.; Rücklé, A.; Srivastava, A.; and Gurevych, I. 2021. BEIR: A Heterogeneous Benchmark for Zero-shot Evaluation of Information Retrieval Models. In *Thirty-fifth Conference on Neural Information Processing Systems Datasets and Benchmarks Track (Round 2)*.
- Wang, X.; Wang, Z.; Gao, X.; Zhang, F.; Wu, Y.; Xu, Z.; Shi, T.; Wang, Z.; Li, S.; Qian, Q.; Yin, R.; Lv, C.; Zheng, X.; and Huang, X. 2024. Searching for Best Practices in Retrieval-Augmented Generation. In Al-Onaizan, Y.; Bansal, M.; and Chen, Y.-N., eds., *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, 17716–17736. Miami, Florida, USA: Association for Computational Linguistics.
- Wang, Y.; Deng, J.; Sun, A.; and Meng, X. 2022. Perplexity from plm is unreliable for evaluating text quality. *arXiv preprint arXiv:2210.05892*.
- Wang, Z.; Araki, J.; Jiang, Z.; Parvez, M. R.; and Neubig, G. 2023. Learning to filter context for retrieval-augmented generation. *arXiv preprint arXiv:2311.08377*.
- Wu, M.; Zhang, Z.; Dong, Q.; Xi, Z.; Zhao, J.; Jin, S.; Fan, X.; Zhou, Y.; Fu, Y.; Liu, Q.; et al. 2025. Reasoning or Memorization? Unreliable Results of Reinforcement Learning Due to Data Contamination. *arXiv preprint arXiv:2507.10532*.
- Xu, F.; Shi, W.; and Choi, E. 2024. RECOMP: Improving Retrieval-Augmented LMs with Context Compression and Selective Augmentation. In *The Twelfth International Conference on Learning Representations*.
- Xu, Y.; Chakraborty, T.; Sharma, S.; Nunes, L.; Kıcıman, E.; Lu, S.; and Chandra, R. 2025. Direct reasoning optimization: LLMs can reward and refine their own reasoning for open-ended tasks. *arXiv preprint arXiv:2506.13351*.
- Yang, A.; Li, A.; Yang, B.; Zhang, B.; Hui, B.; Zheng, B.; Yu, B.; Gao, C.; Huang, C.; Lv, C.; et al. 2025. Qwen3 technical report. *arXiv preprint arXiv:2505.09388*.
- Yang, Z.; Qi, P.; Zhang, S.; Bengio, Y.; Cohen, W.; Salakhutdinov, R.; and Manning, C. D. 2018. HotpotQA: A Dataset for Diverse, Explainable Multi-hop Question Answering. In Riloff, E.; Chiang, D.; Hockenmaier, J.; and Tsujii, J., eds., *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, 2369–2380. Brussels, Belgium: Association for Computational Linguistics.
- Yoon, C.; Lee, T.; Hwang, H.; Jeong, M.; and Kang, J. 2024. CompAct: Compressing Retrieved Documents Actively for Question Answering. In Al-Onaizan, Y.; Bansal, M.; and Chen, Y.-N., eds., *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, 21424–21439. Miami, Florida, USA: Association for Computational Linguistics.
- Yu, T.; Ji, B.; Wang, S.; Yao, S.; Wang, Z.; Cui, G.; Yuan, L.; Ding, N.; Yao, Y.; Liu, Z.; et al. 2025. RLPR: Extrapolating RLVR to General Domains without Verifiers. *arXiv preprint arXiv:2506.18254*.
- Zhang, H.; Wang, P.; Diao, S.; Lin, Y.; Pan, R.; Dong, H.; Zhang, D.; Molchanov, P.; and Zhang, T. 2025. Entropy-Regularized Process Reward Model. *Transactions on Machine Learning Research*.

A A. Prompt Design

In this section, we present our prompts used for the experiments in Section 5.1. In Listing 1, we present the prompt $p_{compressor}$, which is used to generate summaries from the given question and N retrieved passages.

Prompt for Compressor LLM

Instruct

1. Generate a summary of source documents to answer the question. Ensure the summary is under 200 words and does not include any pronouns. DO NOT make assumptions or attempt to answer the question; your job is to summarise only.
2. Evaluate the summary based solely on the information of it, without any additional background context.

Question: {question}

Source documents: {document_input}

Summary:

In Listing 2, we present the prompt p_{reader} , which is used to generate answers from the given question and summary.

Prompt for Reader LLM

Write a high-quality answer for the given question using only the provided search results (some of which might be irrelevant).

In Listing 3, we present the prompt p_{table1} , which is used to generate the summary from the given question for the table 1 experiment.

Prompt for generating the group of summary in Experiment Table 1

Instruct

Your job is to act as a professional writer. You will write a good-quality passage that can support the given prediction about the question only based on the information in the provided supporting passages. Now, let's start. After you write, please write [DONE] to indicate you are done. Do not write a prefix (e.g., "Response:") while writing a passage.

Question: example['question']

Source documents: document_input

Summary:"

B B. Theoretical discussion

In this section, we would like to discuss the theoretical intuition to support the methodology.

B.1 Bounder and the order

We first define the concept of a bounded vector and introduce the corresponding notion of hyper-rectangle order. We then prove that the hyper-rectangle order constitutes a partial order. Furthermore, we show that if one convex hull in a high-dimensional vector space encloses another, their corresponding bounded vectors also satisfy the hyper-rectangle order.

Definition 1. Bounder vector: Given a sequence $\mathbf{x} = (\mathbf{x}_1, \dots, \mathbf{x}_m)$, where each \mathbf{x}_i is an n -dimensional vector, a vector $\mathbf{M} \in \mathbb{R}^n$ is called a bounder vector of \mathbf{x} if, for every \mathbf{x}_i and for every dimension $k \in \{1, \dots, n\}$, it holds that $M^k \geq x_i^k$, where M^k and x_i^k denote the k -th components of \mathbf{M} and \mathbf{x}_i , respectively.

Obviously, for any token sequence, the max pooling result is a bounder vector of the sequence. Then, we can define a partial order based on the bounder vector.

Definition 2. Hyper-rectangle order: Given two sequences $\mathbf{x} = (\mathbf{x}_1, \dots, \mathbf{x}_m)$ and $\mathbf{y} = (\mathbf{y}_1, \dots, \mathbf{y}_l)$, where each \mathbf{x}_i or \mathbf{y}_j is an n -dimensional vector, with corresponding bounder vectors $\mathbf{M}_x, \mathbf{M}_y \in \mathbb{R}^n$, we say $\mathbf{x} \preceq \mathbf{y}$ if for every dimension $k \in \{1, \dots, n\}$, it holds that $M_x^k \leq M_y^k$, where M_x^k and M_y^k denote the k -th components of \mathbf{M}_x and \mathbf{M}_y , respectively. Here, \preceq is the hyper-rectangle order.

We refer to it as the hyper-rectangle order because a minimal enclosing hyper-rectangle can be constructed based on the bounder vector to cover all token embeddings of a given sentence. Next, we discuss the relationship between the hyper-rectangle order and the convex hull. In addition, analogous to **Definition 2**, we formally define the hyper-rectangle order \succeq .

Theorem 1. Given a sequence $\mathbf{x} = (\mathbf{x}_1, \dots, \mathbf{x}_m)$, for any subsequence of \mathbf{x} , denoted as \mathbf{x}_{sub} , it always holds that $\mathbf{x}_{sub} \preceq \mathbf{x}$.

Proof. For a given sequence $\mathbf{x} = (\mathbf{x}_1, \dots, \mathbf{x}_m)$, suppose there is a subsequence $\mathbf{x}' = (\mathbf{x}_{i_1}, \dots, \mathbf{x}_{i_t})$, where $\{i_1, \dots, i_t\} \subseteq \{1, \dots, m\}$ and $t \leq m$. Let \mathbf{M}_x and $\mathbf{M}_{x'}$ be the corresponding bounder vectors of \mathbf{x} and \mathbf{x}' , respectively.

By the definition of bounder vector, for any dimension k :

$$M_{x'}^k = \max_{j \in \{1, \dots, t\}} x_{i_j}^k \quad (4)$$

$$M_x^k = \max_{i \in \{1, \dots, m\}} x_i^k \quad (5)$$

Since $\{i_1, \dots, i_t\} \subseteq \{1, \dots, m\}$, we have:

$$\max_{j \in \{1, \dots, t\}} x_{i_j}^k \leq \max_{i \in \{1, \dots, m\}} x_i^k$$

Therefore, $M_{x'}^k \leq M_x^k$ for all k , which implies $\mathbf{x}' \preceq \mathbf{x}$. \square

Theorem 1 states that as the number of tokens increases, the corresponding bounder vector and its associated hyper-rectangle order also expand. This aligns with the intuition

of information gain as token length grows. An exception occurs when the newly added tokens are low-content or semantically insignificant, as illustrated in the example from the Introduction. In such cases, these tokens tend to lie near the center of the distribution and, by our definition, do not affect the bounder vector.

Theorem 2. *Given a sequence $x = (x_1, \dots, x_m)$ and corresponding convex hull C_x , for any sampled sequence from C_x , denoted as x_{sample} , it always holds that $x_{\text{sample}} \preceq x$.*

Proof. Since C_x is the convex hull of sequence $x = (x_1, \dots, x_m)$, for any $s \in C_x$, there exist $\lambda_i \geq 0$ with $\sum_{i=1}^m \lambda_i = 1$, such that $s = \sum_{i=1}^m \lambda_i x_i$.

Therefore, for any sequence $x_{\text{sample}} = (s_1, \dots, s_t)$ sampled from the convex hull C_x , let M_{sample} and M_x be the corresponding bounder vectors. For any dimension k and any $s_j \in x_{\text{sample}}$:

$$s_j^k = \sum_{i=1}^m \lambda_{ji} x_i^k \leq \sum_{i=1}^m \lambda_{ji} M_x^k = M_x^k$$

Thus $M_{\text{sample}}^k = \max_j s_j^k \leq M_x^k$ for all k , which implies $x_{\text{sample}} \preceq x$. \square

Theorem 2 states that, based on our definition, the convex hull of a given sequence is bounded by its corresponding bounder vector. This bounder vector can thus be used to approximate the "coverage" of the token sequence. The remaining question is: how can we effectively use it for such measurement?

B.2 Measuring the sampling

Here, we discuss how to use the bounded vector to measure the area covered by a given sampling result. This analysis is based on the following question: If two sampling results are drawn from the same distribution, will their corresponding bounded vectors converge as the sample size increases?

Theorem 3. *For any two sampling results x_a and x_b from the same distribution (generator), with corresponding bounder vectors M_a and M_b , for all $\epsilon > 0$, the difference between M_a and M_b converges to zero in probability as the sample size m approaches infinity, that is: $\lim_{m \rightarrow \infty} P(\|M_a - M_b\| \geq \epsilon) = 0$.*

Where m is the sampling size, P is the probability and $\|\cdot\|$ is the norm.

Proof. We focus on a single component $k \in \{1, \dots, n\}$, where n is the dimension of the vector space. Let M_a^k and M_b^k be the k -th components of the bounding vectors. We must show that $|M_a^k - M_b^k|$ converges to 0 in probability, i.e.,

$$\lim_{m \rightarrow \infty} P(|M_a^k - M_b^k| > \epsilon) = 0$$

Let $F_k(x)$ be the cumulative distribution function (CDF) of the k -th component. The CDF of the sample maximum for a sample of size m is given by:

$$F_{M_a^k}(x) = P(M_a^k \leq x) = [F_k(x)]^m$$

Let $s_k = \sup\{x : F_k(x) < 1\}$ be the essential supremum of the k -th component.

The event $\{|M_a^k - M_b^k| > \epsilon\}$ can be split into two events: $\{M_a^k > M_b^k + \epsilon\}$ and $\{M_b^k > M_a^k + \epsilon\}$. By symmetry, we analyze the first event:

$$\begin{aligned} P(M_a^k > M_b^k + \epsilon) &= \int_{-\infty}^{\infty} P(M_a^k > y + \epsilon) f_{M_b^k}(y) dy \\ &= \int_{-\infty}^{\infty} (1 - [F_k(y + \epsilon)]^m) f_{M_b^k}(y) dy \end{aligned}$$

As $m \rightarrow \infty$, the density $f_{M_b^k}(y)$ becomes increasingly concentrated near s_k . For any $y < s_k - \epsilon$, we have $F_k(y + \epsilon) < 1$, so $[F_k(y + \epsilon)]^m \rightarrow 0$. However, the mass of $f_{M_b^k}$ in this region vanishes. For y near s_k , if $F_k(y + \epsilon) = 1$, then $1 - [F_k(y + \epsilon)]^m = 0$. Thus, the integral vanishes as $m \rightarrow \infty$.

By symmetry, $P(M_b^k > M_a^k + \epsilon) \rightarrow 0$ as well, so $P(|M_a^k - M_b^k| > \epsilon) \rightarrow 0$.

The probability that the norm of the difference vector exceeds ϵ is bounded by:

$$P(\|M_a - M_b\| > \epsilon) \leq \sum_{k=1}^n P(|M_a^k - M_b^k| > \epsilon/\sqrt{n})$$

Since each term goes to zero as $m \rightarrow \infty$, the entire sum also goes to zero, completing the proof. \square

This theorem establishes a direct and rigorous link between the sample-based boundary vector and a fundamental property of the underlying distribution. The bounder vector, while a statistic derived from a finite sample, is not a random artifact. Instead, it is a consistent estimator of the distribution's true axis-aligned support.

The boundary vector captures the extremal properties of the distribution, specifically its range along each coordinate axis. As the sample size m increases, the probability that the sample bounding box deviates significantly from the true boundary vector of the distribution's support becomes vanishingly small.

B.3 Further Discussion

In our method, we use different models for the retriever and the reader. Our primary concern is whether the summary generated by the retriever can be aligned with or accepted by the reader. Given a retrieved summary sequence, we can easily compute its corresponding bounded vector from the reader's perspective. The key question is whether this bounded vector, if treated as a proxy for the reader's output, aligns with the actual token embeddings generated by the reader. To assess this, we compare the bounded vector with the spectrum projection direction of the reader to obtain a matching score.

However, it is important to consider under what conditions our proposed framework is necessary. We assume that the token embedding distribution is sparse, and there exists

a significant difference between the mean vector of the sequence $\mathbf{x} = (\mathbf{x}_1, \dots, \mathbf{x}_n)$ and its bounded vector. In real-world scenarios, however, token embedding distributions may vary considerably. When the distribution is heavily concentrated or dense, our assumption may not hold.

To analyse this further, we adopt the commonly used assumption that token embeddings follow a multi-dimensional Gaussian distribution (Ng et al. 2011; Karvonen et al. 2025). Under this assumption, we consider the ratio between the mean vector and the bounded vector, defined as:

$$\mathcal{R} = \frac{\|\bar{\mathbf{x}}\|}{\|\mathbf{M}_x\|}$$

In most LLMs, the mean vector is close to zero. By the Strong Law of Large Numbers, the sample mean converges almost surely to the true mean vector $\boldsymbol{\mu}$:

$$\lim_{n \rightarrow \infty} \bar{\mathbf{x}} = \boldsymbol{\mu} \quad (\text{almost surely})$$

In contrast, the bounded vector \mathbf{M}_x , which captures the extreme values in the sample, grows unbounded. Specifically, for a Gaussian distribution with standard deviation σ , the expected maximum of n samples increases approximately as $\sigma\sqrt{2\ln n}$. Thus:

$$\lim_{n \rightarrow \infty} \|\mathbf{M}_x\| = \infty$$

Combining the asymptotic behaviours of the numerator and denominator, we find that the ratio \mathcal{R} converges to zero:

$$\lim_{n \rightarrow \infty} \mathcal{R} = \lim_{n \rightarrow \infty} \frac{\|\bar{\mathbf{x}}\|}{\|\mathbf{M}_x\|} = \lim_{n \rightarrow \infty} \frac{\|\boldsymbol{\mu}\|}{\mathcal{O}(\sigma\sqrt{2\ln n})} = 0$$

This demonstrates that as the sample size increases, the mean vector becomes negligible in magnitude relative to the boundary vector.

However, in practice, the number of tokens in a summary is limited, so n is bounded and $\|\mathbf{M}_x\|$ remains controlled by a function of σ . We propose using the ratio \mathcal{R} as a filtering criterion to detect sparsity in token embeddings. Specifically, when σ is large (indicating high variance and sparse distribution), the bounded vector grows large relative to the mean, resulting in a small \mathcal{R} . We apply our method only to summaries with sufficiently low ratios, indicating sparse and informative distributions.

C C. Overconfidence Analysis of Qwen and LLaMa Model

We measured answer entropy across different models and datasets. On the HotpotQA and NQ datasets, the Qwen model tends to produce answers with very high confidence scores (i.e., predictive probabilities) compared with the LLAMA model. This observation is supported by the results in Table 4, where we measure the entropy of generated answers across datasets. Qwen consistently yield *lower entropy values*, indicating a strong tendency toward *overconfident predictions*.

Model	HotpotQA	NQ
LLaMA-3.1-8B-Instruct	1.49	1.16
Qwen3-8B	0.46	0.38

Table 4: Average answer entropy across datasets for different LLMs. Lower entropy indicates higher confidence in answering questions.