

# Don't Be Misled by Style: A Style-Adaptive Reranker for Capturing Effective Knowledge in Retrieval-Augmented Generation

Ruwen Zhang<sup>1</sup>, Bo Liu<sup>1,2,\*</sup>, Shengxiang Zhang<sup>1</sup>, Yida Chen<sup>1</sup>,  
Hantao Zhao<sup>1,2</sup>, Ding Ding<sup>1</sup>, Jiahui Jin<sup>1</sup>, Jiuxin Cao<sup>1,2</sup>

<sup>1</sup>Southeast University, Nanjing, China <sup>2</sup>Purple Mountain Laboratories, Nanjing, China

Correspondence: bliu@seu.edu.cn

## Abstract

Rerankers are critical in Retrieval-Augmented Generation (RAG) for filtering evidence that enhances the accurate generation of LLMs. With the extension to open-domain scenarios, rerankers are inevitably deployed on mixed-style corpora, whereas most existing rerankers are mainly trained on well-edited texts. A rarely explored issue lies in enabling rerankers to maximally capture the effective knowledge for downstream LLMs without being misled by stylistic features. To address this issue, we propose SARK (Style-Adaptive Reranker with Knowledge Prioritization), a style-augmented multi-task framework that prioritizes effective knowledge over stylistic perturbations. SARK performs multi-granular knowledge mining by using an LLM to derive passage-level supervision on whether a passage helps or harms answer correctness, and list-level relative ranking preferences over candidate passages. It then jointly optimizes the reranker model with passage-level classification and list-level ranking objectives via style-augmented multi-task learning, encouraging the model to focus on the information needed for answering under mixed-style scenarios. Extensive experiments demonstrate that SARK improves generation performance across multiple LLMs under mixed-style conditions.

## 1 Introduction

Large Language Models (LLMs) have demonstrated remarkable capabilities across diverse domains, from dialogue systems (Castillo-Bolado et al., 2024; Choi et al., 2024) to recommendation (Ren et al., 2024). However, inherent limitations in parametric knowledge often lead to hallucinations (Huang et al., 2025), motivating the widespread adoption of Retrieval-Augmented Generation to enhance the factual accuracy of LLM outputs.

A typical RAG system usually relies on a retriever-reranker pipeline (Gao et al., 2023). The

reranker is responsible for reordering candidate passages recalled from external corpora, determining which information is ultimately exposed to the LLM. In practice, real-world retrieval systems aggregate content from mixed-style data sources. As illustrated in Fig. 1, the retrieval corpora differ not only in topical coverage but also in writing style, including encyclopedic articles, news reports, or social media text. Ideally, a reranker should prioritize passages based solely on supportive evidence. Yet, existing models trained predominantly on single stylistic datasets tend to overfit to specific linguistic patterns rather than capturing underlying semantic relevance (Cao, 2025). As a result, their ranking results are biased by superficial stylistic cues, leading to the failure to identify valuable knowledge when it appears in informal formats.

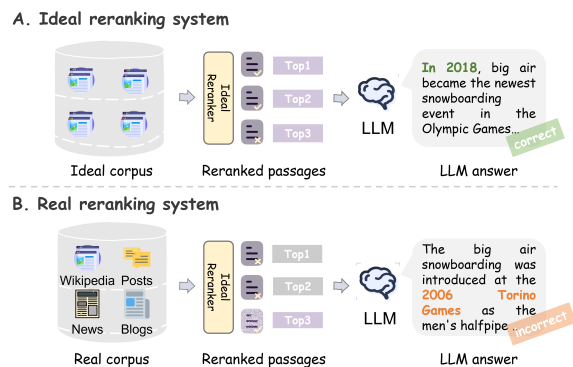


Figure 1: The comparison of the ideal and real reranking systems.

Recent reranking methods can be broadly divided into encoder-based approaches, which rely on representation learning for similarity estimation, and generative approaches, which exploit LLMs for ranking or supervision (Sun et al., 2023; Ma et al., 2023; Zhang et al., 2024b). Encoder-based models, such as ColBERT (Khattab and Zaharia, 2020) and BGE (Xiao et al., 2024), focus on fine-grained semantic matching but are not ex-

PLICITLY optimized to align with the deep reasoning of LLMs (Ren et al., 2025). Conversely, generative methods attempt to optimize rerankers using feedback signals from LLMs (Shi et al., 2024; Jiang et al., 2025; Dong et al., 2025). For instance, RAFT (Zhang et al., 2024b) embeds reranking logic directly into the generation process to filter irrelevant contexts, while DPA-RAG (Dong et al., 2025) derives preference signals from downstream answer accuracy. However, these methods overlook the issue of style sensitivity in reranking. Since rerankers are typically trained or optimized on formal text, they may implicitly rely on formal stylistic features as cues for relevance. Consequently, informal but semantically relevant evidence may be down-ranked, leading to inaccurate ranking decisions.

The reliance on single stylistic training data makes existing ranking models susceptible to stylistic variations. Though some research has treated style as a target for generation alignment (Zhang et al., 2024d), attempting to match output styles to human preferences. In the context of reranking, style merely acts as one of the forms of information, often distracting noise that obscures the underlying facts. The critical challenge is to **ensure that the ranking order is strictly driven by factual evidence, remaining effectiveness regardless of the source of candidates**, from a well-structured article to a social media post.

To tackle this challenge, we present SARK, a Style-Adaptive Reranker with Knowledge prioritization. SARK is trained on a multi-style dataset with formal and informal passage variants, prioritizing core evidence over surface-level presentation. To achieve this, SARK employs a **Multi-Granular Knowledge Extraction** module to mine both passage-level validity and list-level relative signals, effectively identifying query-relevant evidence from LLMs. Furthermore, SARK integrates **Style-Augmented Multi-task Learning**, which optimizes the model over the joint distribution of style-variant pairs. This joint optimization prevents SARK from overfitting to specific writing styles while retaining sensitivity to effective evidence. Extensive experiments demonstrate that SARK achieves superior robustness under style-diverse retrieval conditions. Importantly, SARK consistently improves performance across multiple LLMs, serving as a plug-and-play module for real-world RAG systems.

Our contributions are summarized as follows:

- We build a multi-style reranking dataset with paired formal and informal passages, annotated with passage-level supportive labels and list-level ranking orders, benchmarking rerankers for effective knowledge-aware ranking capabilities under multi-style corpora.
- We introduce SARK, a novel reranking framework that integrates multi-granular knowledge extraction with style-augmented multi-task learning, ensuring ranking is driven by effective knowledge while adopting stylistic diversity.
- We empirically uncover the impact of style variations on rerankers and analyze the effectiveness of SARK across diverse stylistic settings and multiple LLM backbones compared to existing methods.

## 2 Problem Definition

Given a query  $q \in \mathcal{Q}$  and a candidate set  $\mathcal{C}_q = \{p_1, p_2, \dots, p_n\}$  retrieved from an external corpus  $\mathcal{D}$ , the primary goal of a reranker is to determine a permutation  $\mathbf{R} = [p_1, p_2, \dots, p_n]$  of  $\mathcal{C}_q$ . This ranking is induced by a learnable scoring function  $f_\theta(q, p)$  that ranks passages based on their scores.

We formalize a passage  $p$  as a tuple  $\langle c, s \rangle$ , where  $c$  denotes the useful semantic content and  $s \in \mathcal{S}$  represents the linguistic style. Our optimization objective aims to learn a relevance score that depends solely on the content  $c$  against the influence of  $s$ , formally defined as  $f_\theta(q, \langle c, s_i \rangle) \approx f_\theta(q, \langle c, s_j \rangle)$  for any linguistic styles  $s_i, s_j \in \mathcal{S}$ .

## 3 Motivation Analysis

In this section, we formalize the linguistic styles and conduct preliminary experiments to quantify the reranker’s sensitivity to stylistic variations.

### 3.1 Style Formulation

In open-domain retrieval scenarios, candidate passages typically exhibit diverse linguistic styles, which may introduce unintended biases into the reranking process. To investigate this, we instantiate the style type  $s$  into two distinct categories following prior work on style transfer (Mukherjee and Dušek, 2024; Liu et al., 2022), including formal style  $s^{\text{fml}}$  and informal style  $s^{\text{inf}}$ . We construct style-variant pairs where a formal passage  $p^{\text{fml}} = \langle c, s^{\text{fml}} \rangle$  is contrasted with its informal passage  $p^{\text{inf}} = \langle c, s^{\text{inf}} \rangle$ , with the semantic content  $c$

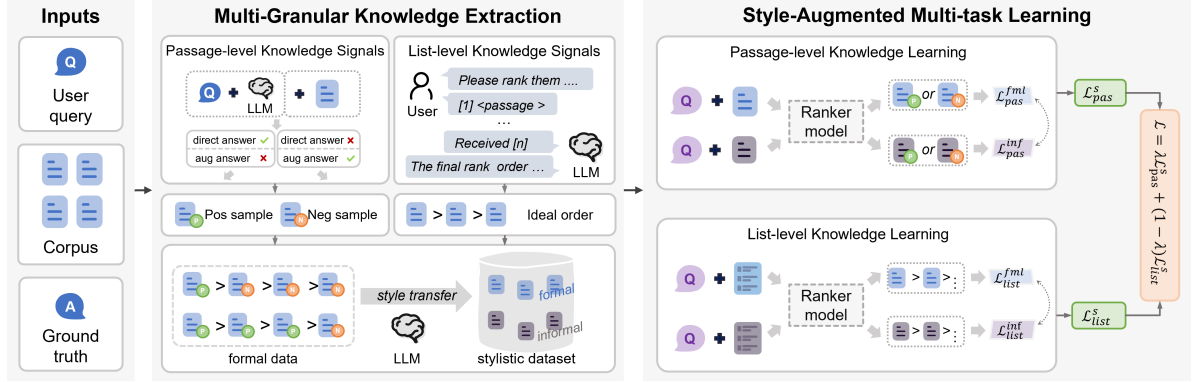


Figure 2: The overall architecture of the proposed SARK.

Table 1: Performance gaps between formal and informal data under style-related variations. Existing rerankers consistently degrade on informal text in terms of accuracy and BERTScore (Top-50 passages).

Model	Accuracy		BERTScore	
	Formal	$\Delta$	Formal	$\Delta$
BGE-Base	0.5377	4.65%↓	0.7476	2.54%↓
BGE-Large	0.5330	4.37%↓	0.7413	1.86%↓
MiniLM	0.5348	5.08%↓	0.7515	3.18%↓
mGTE	0.5424	5.16%↓	0.7514	3.00%↓
RankGPT	0.5397	5.02%↓	0.7476	3.03%↓
RankLlama	0.5504	5.64%↓	0.7575	3.40%↓
<b>Avg. Drop</b>	–	4.99%↓	–	2.84%↓

strictly preserved. This formulation allows us to isolate style as a variable to test rerankers.

### 3.2 Reranker Sensitivity to Style

As the training distribution of current rerankers is heavily trained on a fixed style  $s^{\text{fml}}$ , we hypothesize that they inevitably mistake the specific style for semantic validity.

To validate this hypothesis, we conduct an empirical analysis utilizing the Natural Questions (NQ) dataset (Kwiatkowski et al., 2019). We construct an evaluation set containing style-variant pairs  $(p^{\text{fml}}, p^{\text{inf}})$ , which share identical semantic content  $c$  but differ in style  $s$ , with  $s^{\text{fml}}$  representing standard encyclopedic text and  $s^{\text{inf}}$  representing colloquial variations. The details of this data construction process are provided in Appendix A.

Table 1 reports the performance of six representative reranking models on the Top-50 retrieved paragraphs. The results provide strong empirical evidence of style vulnerability. When the same effective knowledge is presented in an informal style, all evaluated models suffer significant performance

degradation, with LLM answer accuracy (ACC) dropping by up to 4.99% and BERTScore (Zhang et al., 2019) decreasing by 2.84%. Since the semantic content  $c$  remains unchanged, this degradation confirms that state-of-the-art rerankers are heavily biased on specific linguistic patterns instead of identifying the effective knowledge. This observation highlights the critical need for a reranker to meet the core knowledge required by LLMs against superficial stylistic cues, motivating the design of our proposed SARK framework.

## 4 Proposed Approach

### 4.1 Framework

To mitigate the misleading influence of stylistic features, we propose SARK. As illustrated in Fig. 2, SARK integrates two modules that prioritize useful knowledge under style augmentation via multi-task learning. The **Multi-Granular Knowledge Extraction** module treats the LLM as a reasoning verifier. It extracts effective knowledge signals at two granularities, comprising passage-level verification to identify factual support, and list-level ranking to capture the relative effectiveness of candidates for answer generation. **Style-Augmented Multi-task Learning** module leverages these signals to confer resilience against stylistic perturbations via a multi-task objective. It aggregates the passage-level and list-level losses across style-variant pairs, forcing the model to learn semantic features. Through the joint optimization of multiple tasks, SARK establishes a knowledge-centric ranking that is robust to stylistic variation.

### 4.2 Multi-Granular Knowledge Extraction

In this section, we extract effective knowledge directly from the target LLM, leveraging the LLM

as a reasoning verifier to discern the contribution of retrieved passages at distinct granularities.

#### 4.2.1 Passage-level Knowledge Signals

To quantify the validity of individual passages, we identify passages that provide the necessary factual support to correct the LLM’s reasoning by comparing the LLM’s answer accuracy with and without context augmentation.

Formally, let  $q$  be a query and  $y$  be the ground truth answer. For each candidate passage  $p_i \in \mathcal{C}_q$ , we generate a retrieval-augmented response  $a_i^{\text{aug}}$  via a prompt template  $\mathcal{T}(q, p_i)$ . Concurrently, we query the LLM without retrieval to obtain its intrinsic parametric answer  $a^{\text{dir}}$ , as defined in Eq. (1):

$$a_i^{\text{aug}} = \mathcal{M}(\mathcal{T}(q, p_i)), \quad a^{\text{dir}} = \mathcal{M}(q). \quad (1)$$

To assess factual correctness, we employ a hybrid verification  $\mathcal{E}(a, y) \in \{\text{Match}, \text{Mismatch}\}$ . It integrates the EM score for character-level match and an LLM-based judge to handle semantic variability in correct responses. We construct the passage-level ranking dataset  $\mathcal{D}_{\text{pas}}$  where the retrieved  $p_i$  explicitly alters the outcome. The labeling logic is formalized in Eq. (2):

$$\mathcal{D}_{\text{pas}} = \{(q, p_i, l_i) \mid l_i \in \{\text{pos}, \text{neg}\}\}, \quad (2)$$

where the label  $l_i$  is derived based on answer gain. A positive label  $l_i$  is assigned when the LLM lacks parametric knowledge ( $\mathcal{E}(a^{\text{dir}}, y) = \text{Mismatch}$ ) but successfully answers with augmentation ( $\mathcal{E}(a_i^{\text{aug}}, y) = \text{Match}$ ), indicating  $p_i$  provides the necessary knowledge. A hard negative label  $l_i$  is assigned when the LLM answers correctly via parametric knowledge but fails after incorporating the passage ( $\mathcal{E}(a^{\text{dir}}, y) = \text{Match} \wedge \mathcal{E}(a_i^{\text{aug}}, y) = \text{Mismatch}$ ), indicating  $p_i$  introduces misleading information.

#### 4.2.2 List-level Knowledge Signals

Passage-level signals offer fine-grained supervision, as LLMs may answer correctly regardless of context, obscuring the relative effectiveness of candidates (seen in Fig. 3). To capture relative effectiveness, we adopt a multi-turn conversational prompting strategy, which guides the LLM to rank passages based on their contribution to the ground truth answer.

Formally, for a query  $q$  and a sampled subset of passages, the LLM induces an optimal ranking order  $L = [p_1, p_2, \dots, p_n]$ , where  $p_i$  denotes the

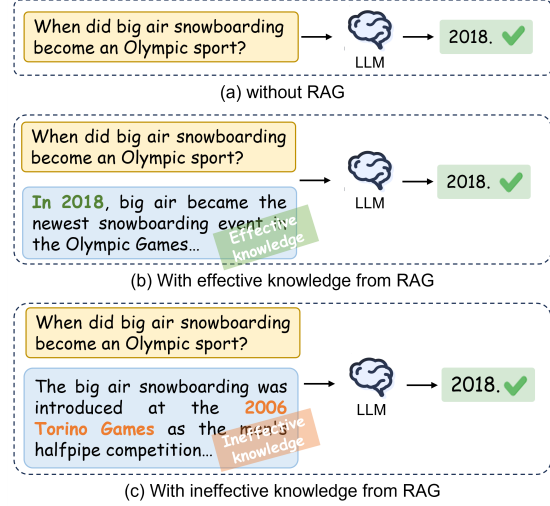


Figure 3: The LLM answers correctly regardless of retrieval, masking the effectiveness of retrieved passages.

passage at rank  $i$ . From this global ordering, we extract pairwise ranking constraints to construct the list-level dataset  $\mathcal{D}_{\text{list}}$ , as formalized in Eq. (3):

$$\mathcal{D}_{\text{list}} = \{(q, p_i, p_j) \mid p_i \succ p_j \in L\}, \quad (3)$$

where  $p_i \succ p_j$  indicates that  $p_i$  provides superior support compared to  $p_j$ . The final training dataset is  $\mathcal{D} = \{\mathcal{D}_{\text{pas}}, \mathcal{D}_{\text{list}}\}$ .

#### 4.2.3 Knowledge Style Transfer

To enhance SARK with the capability to capture core semantics from linguistic presentation, we augment the corpus with style-variant passage pairs. We treat the original passages from original  $\mathcal{D}$  as the standard formal anchors, denoted as  $p^{\text{fml}}$ . Subsequently, we leverage an LLM to perform the style transfer, generating corresponding informal variants  $p^{\text{inf}}$ .

Specifically, we design a prompt template  $\mathcal{T}_{\text{trans}}$  that instructs the LLM to rewrite the input text into a colloquial tone while strictly preserving the factual evidence, as detailed in Appendix A. This process yields a set of semantically equivalent pairs  $\{(p^{\text{fml}}, p^{\text{inf}})\}$  as  $\mathcal{D}'$ , which serves as the foundation for our style-augmented joint optimization.

#### 4.3 Style-Augmented Multi-task Learning

Building on the extracted multi-granular signals, SARK adopts a multi-task learning framework. It integrates passage-level classification to ensure factual correctness, while simultaneously optimizing list-level ranking to capture the relative order of evidence.

### 4.3.1 Passage-level Knowledge Learning

We first feed the  $(q, p_i)$  pair into the backbone reranker to obtain the feature embedding  $\mathbf{h}_i$  corresponding to the [CLS] token. Subsequently, the scoring function  $f_\theta(q, p_i)$  is implemented as a two-layer feed-forward neural network with non-linear activation and dropout regularization, which maps  $\mathbf{h}_i$  to the final output  $z_i$  as Eq. (4):

$$z_i = \mathbf{W}_2 \cdot \text{Dropout}(\phi(\mathbf{W}_1 \mathbf{h}_i + \mathbf{b}_1)) + \mathbf{b}_2, \quad (4)$$

where  $\mathbf{W}_1, \mathbf{W}_2, \mathbf{b}_1, \mathbf{b}_2$  are trainable parameters, and  $\phi(\cdot)$  denotes the activation function.

Consequently, this score is mapped to a probability via the sigmoid function  $\hat{y}_i = \sigma(z_i)$ , representing the likelihood that passage  $p_i$  contains critical evidential support. We formulate the passage-level verification as a binary classification task, minimizing the cross-entropy loss over the passage data  $\mathcal{D}'_{pas}$ , as Eq. (5):

$$\mathcal{L}_{pas} = -\frac{1}{|\mathcal{D}'|} \sum_{i=1}^{|\mathcal{D}'|} \left[ y_i \log \hat{y}_i + (1 - y_i) \log(1 - \hat{y}_i) \right]. \quad (5)$$

By minimizing  $\mathcal{L}_{pas}$ , the reranker explicitly learns to distinguish factual evidence from irrelevant noise.

### 4.3.2 List-level Knowledge Learning

To enhance the relative order of valid passages, we employ pairwise optimization to boost relative ranking logic. The probability that passage  $p_i$  provides stronger evidential support than  $p_j$  is represented as a sigmoid function of their score difference in  $[0, 1]$ . Let  $z_i$  be the predicted relevance score, the pairwise probability is defined as Eq. (6):

$$P(p_i \succ p_j) = \sigma(z_i - z_j), \quad (6)$$

where a higher  $P(p_i \succ p_j)$  indicates a higher probability that  $p_i$  is preferred over  $p_j$ .

To align SARK’s scoring with the LLM’s intrinsic reasoning, we minimize the RankNet loss over the set of pairwise constraints  $\mathcal{D}'_{list}$  in Eq. (7):

$$\mathcal{L}_{list} = \frac{1}{|\mathcal{D}'|} \sum_{(q, p_i, p_j) \in \mathcal{D}'} \log \left( 1 + e^{-(z_i - z_j)} \right). \quad (7)$$

By minimizing  $\mathcal{L}_{list}$ , the model learns to discern which evidence is comparatively more effective.

### 4.3.3 Style-Augmented Joint Training

The style-augmented joint training mechanism leverages multi-task learning to enhance prediction consistency to stylistic differences.

Specifically, we define the style-augmented passage loss  $\mathcal{L}_{pas}^s$  and list loss  $\mathcal{L}_{list}^s$  by aggregating losses from both stylistic contexts, as formulated in Eq. (8) and (9):

$$\mathcal{L}_{pas}^s = \text{MEAN} \left( \mathcal{L}_{pas}(p^{fml}) + \mathcal{L}_{pas}(p^{inf}) \right), \quad (8)$$

$$\mathcal{L}_{list}^s = \text{MEAN} \left( \mathcal{L}_{list}(\mathcal{C}_q^{fml}) + \mathcal{L}_{list}(\mathcal{C}_q^{inf}) \right). \quad (9)$$

The overall training objective is a weighted integration of these multi-task losses. To balance the local precision and global ordering, we formulate the total loss  $\mathcal{L}$  as shown in Eq. (10):

$$\mathcal{L} = \lambda \mathcal{L}_{pas}^s + (1 - \lambda) \mathcal{L}_{list}^s, \quad (10)$$

where  $\lambda \in [0, 1]$  is a dynamic weighting factor adaptively adjusted at each training step  $t$  to balance the optimization gradients of the two tasks.

## 5 Experiments

In this section, we evaluate SARK guided by five key research questions. Q1: How effective is SARK compared with existing rerankers on stylistic data? Q2: How effective is SARK as the size of the recalled candidate set increases? Q3: How do different style mix ratios within the candidate set impact SARK? Q4: Is SARK generalizable across various LLM sizes and types? Q5: What are the practical inference costs of deploying SARK?

### 5.1 Experimental Setting

**Dataset.** We use the open-domain English question-answering dataset NQ (Kwiatkowski et al., 2019) as the training corpus for our reranking model. The candidate passage pool is constructed from the Wikipedia dump (Dec. 20, 2018) (Lee et al., 2019) as the primary knowledge source. The statistics details on dataset construction are provided in Appendix A.

**Evaluation Metrics.** We use Qwen-14B<sup>1</sup> as the backbone model to generate answers for evaluation. Ranking performance is assessed using Accuracy and Exact Match (EM) (Rajpurkar et al., 2016). To further measure answer quality and similarity to the ground truth, we further adopt BERTScore and ROUGE (Lin, 2004). Additional details on the experimental setup are provided in Appendix B.

<sup>1</sup><https://huggingface.co/Qwen/Qwen2.5-14B-Instruct>

Table 2: Top-5 performance comparison across reranking models.

Model	Accuracy	EM	BERTScore	ROUGE
BGE-Base	0.4745	0.3020	0.7055	0.5327
BGE-Large	0.4843	0.3066	0.7104	0.5516
MiniLM	0.4945	0.3210	0.7224	0.5669
mGTE	0.5057	0.3253	0.7275	0.5785
RankGPT	0.4805	0.3040	0.7085	0.5465
RankLlama	0.5163	0.3330	0.7279	0.5819
<b>SARK</b>	<b>0.5360</b>	<b>0.3460</b>	<b>0.7350</b>	<b>0.5919</b>
w/o SAT	0.5213	0.3350	0.7320	0.5858

## 5.2 Main Results (Q1)

We compare SARK in a mixed-style retrieval setting, where the candidates contain 50% formal and 50% informal passages. As summarized in Table 2, SARK achieves the best performance across all metrics, demonstrating strong robustness against stylistic perturbations.

We observe that BGE family performs the weakest, while BGE-Base achieves only 0.4745 accuracy and 0.3020 EM. This underperformance likely stems from their reliance on surface-level semantic matching, which is easily disrupted by the lexical gap introduced by informal styles. In contrast, generative baselines like RankLLaMA capture deeper semantics, reaching a BERTScore of 0.7279. However, SARK outperforms RankLLaMA with a notable gain of 1.97% in accuracy and 1.00% in ROUGE. These gains indicate that while RankLLaMA may capture general relevance, SARK more effectively captures core knowledge. By learning factual evidence against stylistic attributes, SARK ensures that the retrieved context actually aids the LLM’s reasoning, rather than merely maximizing semantic similarity.

To validate the contribution of our Style-Augmented Joint Training (SAT), we evaluate a variant removing the SAT. Performance drops noticeably, with accuracy dropping from 0.5360 to 0.5213 and ROUGE declining from 0.5919 to 0.5858. This performance drop confirms that the SAT module is essential for projecting diverse variants into a knowledge-centric feature space.

## 5.3 Top-K Ranking Analysis (Q2)

To evaluate performance under different ranking depths, we analyze results across Top- $K$  values from 5 to 50. Fig. 4 presents the results on both formal and informal test sets.

Overall, SARK consistently outperforms all baselines, demonstrating superior stability across

depths. On the formal test set, the relative ordering of rerankers remains largely unchanged, while SARK maintains EM scores between 0.3729 and 0.3839. A clearer divergence appears on the informal test set, where larger  $K$  introduces more stylistic noise. SARK is less sensitive to stylistic variations at every Top-K level, outperforming strong baselines such as mGTE and RankLlama by approximately 3.03% and 1.94% in accuracy at Top-10. Additionally, RankGPT exhibits improved performance as the number of candidate documents increases. This trend is likely attributed to its sliding-window mechanism, which is better suited for longer contexts. Unlike RankGPT that require a larger candidates pool to rerank correct answers, SARK effectively focuses solely on the density of effective knowledge when surrounded by linguistic interference due to the SAT training, thereby maximizing LLM answer accuracy under limited context windows.

## 5.4 Style Robust Analysis (Q3)

We evaluate Top-5 performance across varying mixtures of formal and informal passages, ranging from 20% to 80% formal, as Fig. 5 reports. Overall, SARK maintains stable performance despite significant variations in stylistic noise.

In general, across all methods, accuracy scores improve as the proportion of formal text increases, indicating that informal passages introduce stylistic noise and pose challenges for semantic matching. Nevertheless, SARK effectively bridges stylistic gaps even when informal passages dominate. Under the most challenging formal-informal split (20% formal), it surpasses RankLlama by 2.3%, with the margin slightly decreasing to 2.0% and 2.1% under the 40% and 80% settings, respectively. The BGE variants exhibit volatility across ratios, likely due to their heavy reliance on lexical features. In contrast, the performance of SARK val-

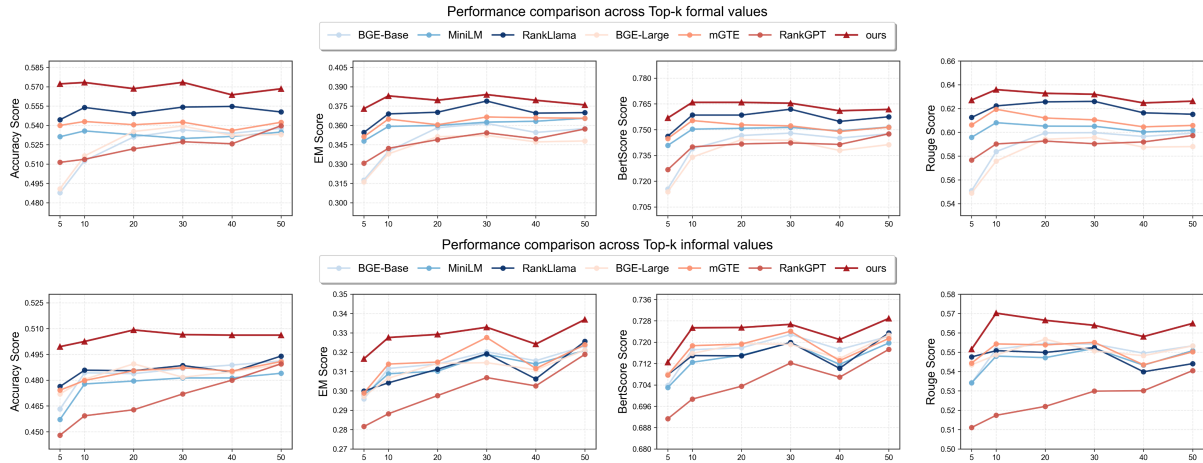


Figure 4: The comparison across Top-K passages on distinct styles.

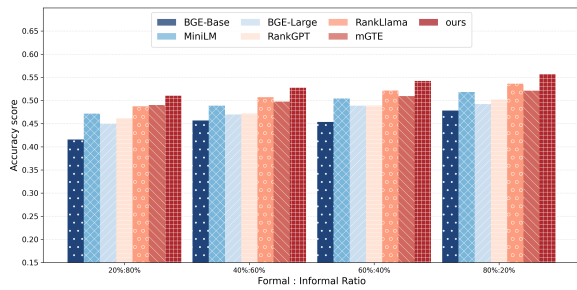


Figure 5: Accuracy Comparison across different style rates.

indates that it can lock onto effective knowledge regardless of the diverse style distributions.

### 5.5 Effectiveness across Different LLMs (Q4)

We deploy our reranker across four distinct four families, including Llama<sup>2</sup>, gemma<sup>3</sup>, Qwen2.5<sup>4</sup>, and Qwen3<sup>5</sup>, using a 50% mixed-style retrieval setting. Table 3 summarizes the performance gains.

We evaluate cross-model generalization to examine whether SARK overfits to the reasoning patterns of the source LLM (Qwen2.5-14B). As shown in Table 3, consistent gains across heterogeneous architectures suggest that SARK does not encode model-specific reasoning logic. Notably, the Qwen series exhibits the most dramatic performance surges, with accuracy improvements exceeding 31.57% and 28.97% for the Qwen3-4B and Qwen2.5-7B variants, respectively. We attribute this to multi-granular knowledge learning, since

<sup>2</sup><https://huggingface.co/meta-llama/Llama-3.1-8B-Instruct>

<sup>3</sup><https://huggingface.co/google/gemma-3-12b-it>

<sup>4</sup><https://huggingface.co/Qwen/Qwen2.5-7B-Instruct>

<sup>5</sup><https://huggingface.co/Qwen/Qwen3-4B-Instruct-2507>

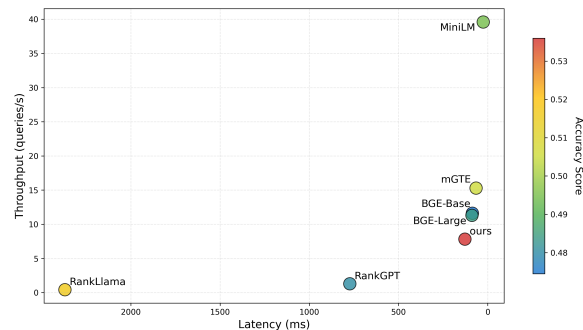


Figure 6: Efficiency comparison across different baselines.

supervision signals are derived from Qwen2.5, SARK has implicitly learned the internal reasoning patterns optimal for the Qwen family. This benefit also transfers significantly to non-Qwen models. For instance, Llama3.1-8B sees an EM increase of nearly 9%, and Gemma3-12B achieves a 12.95% boost in BERTScore. While SARK aligns best with its target LLM, the learned knowledge generalizes benefits diverse LLMs, acting as a plug-and-play module for mixed-style RAG systems.

### 5.6 Efficiency Analysis (Q5)

To assess the inference efficiency, we measure end-to-end inference latency and throughput at the query level, as illustrated in Fig. 6. We deploy RankGPT using the vLLM engine, whereas other models process query-passage pairs in batches. This setup accounts for RankGPT’s listwise prompting, which necessitates individual query processing. Among all methods, RankLlama exhibits the highest latency with 2369.24 ms/query, as it requires multiple forward passes

Table 3: Performance comparison on reranker-enhanced LLMs.

Type	Model	Accuracy	EM	BERTScore	ROUGE
Llama3.1	8B-Instruct	0.2603	0.1213	0.5748	0.3167
	8B-Instruct with SARK	0.4813	0.2110	0.6776	0.5564
Gemma3	4b-it	0.1823	0.1003	0.5469	0.2203
	4b-it + SARK	0.4667	0.2683	0.6855	0.5039
	12b-it	0.2487	0.1627	0.5983	0.3383
	12b-it + SARK	0.4657	0.3243	0.7278	0.5713
Qwen2.5	7B-Instruct	0.1850	0.1213	0.5627	0.2508
	7B-Instruct + SARK	0.4747	0.3283	0.7253	0.5588
	14B-Instruct	0.2517	0.1643	0.5898	0.3129
	14B-Instruct + SARK	0.5360	0.3460	0.7350	0.5919
Qwen3	4B-Instruct	0.2010	0.0170	0.4075	0.0999
	4B-Instruct with SARK	0.5167	0.2480	0.6674	0.4991

over candidate passages. Despite its strong effectiveness, such latency makes it less suitable for latency-sensitive applications. In contrast, MiniLM achieves the highest throughput of 39.59 queries/sec, but often at the cost of its semantic reasoning. Our SARK achieves a favorable balance between efficiency and effectiveness, attaining a latency of 127.89 ms/query and a throughput of 7.82 queries/sec. Notably, SARK achieves an efficiency level comparable to BGE-Large, while being over  $10\times$  faster than RankLLaMA. These results demonstrate that our method bridges the gap between encoder-based and generative rerankers.

## 6 Related work

The primary objective of reranking is to estimate the relevance between a query and candidate documents (Askari et al., 2023; Liu et al., 2025). Based on model architectures, existing approaches can be broadly categorized into encoder-based and generative rerankers.

Encoder-based methods rely on representation learning to computing relevance scores, either by encoding queries and documents independently in a dual-encoder architecture or by jointly encoding concatenated pairs. For instance, BGE Reranker (Xiao et al., 2024) concatenate the query and document to jointly model fine-grained interactions. Lightweight models like MiniLM (Wang et al., 2020) leverage knowledge distillation provide a favorable balance between efficiency and effectiveness. Despite their strong semantic matching, these methods define relevance based on surface-level semantic similarity. As shown in our analysis, they are vulnerable to stylistic noise, leading to the degradation of downstream application.

More recently, the text comprehension and reasoning capabilities of LLMs have driven the development of generative rerankers. On the one hand, approaches like RankT5 (Zhuang et al., 2023), ListT5 (Yoon et al., 2024), and RankLLaMA (Ma et al., 2024) adapt generative architectures to directly output relevance scores based on their sequence generation capabilities. Similarly, RankGPT (Sun et al., 2023) leverages LLMs to perform listwise ranking directly via prompting. On the other hand, to bridge the gap between retriever optimization and generator needs, some works focus on aligning rerankers with downstream feedback (Ren et al., 2025; Jiang et al., 2025; Zhang et al., 2024b). RePlug (Shi et al., 2024) leverages answer perplexity as a supervision signal, while Ar12 (Zhang et al., 2024a) and DPA-RAG (Dong et al., 2025) prompt LLMs to explicitly annotate ranking preferences. While these methods incorporate LLM preferences, they do not explicitly address stylistic bias. In contrast, SARK employs style-augmented multi-task learning to identify evidence rather than linguistic presentation.

## 7 Conclusion

In this paper, we address the challenge where stylistic patterns often misguide RAG systems away from effective knowledge. Our proposed SARK enhances diverse LLM backbones by prioritizing effective knowledge at both passage-level and list-level granularities. Furthermore, through style-augmented multi-task learning, SARK optimizes the model over the joint distribution of style-variant pairs to filter out linguistic noise. Extensive experiments confirm that SARK achieves state-of-the-art robustness in mixed-style retrieval environments.

## Limitations

While our approach shows strong performance, several limitations remain. On the one hand, the reliance on generated data may introduce potential generalization bias. Since our supervision signals are derived from a specific source LLM, the reranker risks overfitting to the specific effective knowledge of that source model, which may fail to generalize consistently across different LLM architectures. On the other hand, the style distribution in our dataset is primarily divided into formal and informal writing, leaving more fine-grained stylistic dimensions underexplored. Extending our framework to account for richer stylistic expression, such as discourse structure, emotional tone, or domain-specific conventions, remains an open direction.

## Ethics Considerations

This work utilizes open-source LLMs for dataset construction and evaluation. We affirm that the source dataset was legally obtained under licenses permitting research use. The data primarily consists of open-domain questions and passages, containing no private or sensitive user information. Besides, we acknowledge the use of AI assistants, including ChatGPT and Gemini, for writing support.

We acknowledge potential risks associated with automated data generation. While our framework is designed to preserve semantic consistency, there remains a possibility that the generated informal variants could inadvertently introduce subtle factual hallucinations. To mitigate these risks, we have employed strict prompts and conducted manual quality checks to ensure the generated content adheres to ethical research standards and contains no offensive content.

Moreover, four human annotators were involved in the experiment to conduct manual quality verification. Each human annotator was compensated at an hourly rate of \$15.00, exceeding the current U.S. federal minimum wage of \$7.25 per hour. We ensure that both the dataset content and construction comply with ethical standards throughout the entire research process.

## Acknowledgments

This work is supported by National Natural Science Foundation of China under Grants No.62472092, No.62172089, and No.62232004,

the Marine Science and Technology Innovation Program of Jiangsu Province, China No. JSZRHYKJ202308, Natural Science Foundation of Jiangsu province under Grants No.BK20191258, Jiangsu Provincial Key Laboratory of Network and Information Security under Grants No.BM2003201, and Key Laboratory of Computer Network and Information Integration of Ministry of Education of China under Grants No.93K-9. This was also supported in part by General Project of MOE (Ministry of Education) Foundation for Humanities and Social Sciences (24YJCZH436).

## References

- Arian Askari, Amin Abolghasemi, Gabriella Pasi, Wesel Kraaij, and Suzan Verberne. 2023. Injecting the bm25 score as text improves bert-based re-rankers. In *European Conference on Information Retrieval*, pages 66–83. Springer.
- Arian Askari, Amin Abolghasemi, Gabriella Pasi, Wesel Kraaij, and Suzan Verberne. 2024. Injecting the score of the first-stage retriever as text improves bert-based re-rankers. *Discover Computing*, 27(1):15.
- Hongliu Cao. 2025. Writing style matters: An examination of bias and fairness in information retrieval systems. In *Proceedings of the Eighteenth ACM International Conference on Web Search and Data Mining*, pages 336–344.
- David Castillo-Bolado, Joseph Davidson, Finlay Gray, and Marek Rosa. 2024. Beyond prompts: Dynamic conversational benchmarking of large language models. *Advances in Neural Information Processing Systems*, 37:42528–42565.
- Dasom Choi, Sunok Lee, Sung-In Kim, Kyungah Lee, Hee Jeong Yoo, Sangsu Lee, and Hwajung Hong. 2024. Unlock life with a chat (gpt): Integrating conversational ai with large language models into everyday lives of autistic individuals. In *Proceedings of the 2024 CHI Conference on Human Factors in Computing Systems*, pages 1–17.
- Guanting Dong, Rumei Li, Sirui Wang, Yupeng Zhang, Yunsen Xian, and Weiran Xu. 2023. Bridging the kb-text gap: Leveraging structured knowledge-aware pre-training for kbqa. In *Proceedings of the 32nd ACM international conference on information and knowledge management*, pages 3854–3859.
- Guanting Dong, Yutao Zhu, Chenghao Zhang, Zechen Wang, Ji-Rong Wen, and Zhicheng Dou. 2025. Understand what llm needs: Dual preference alignment for retrieval-augmented generation. In *Proceedings of the ACM on Web Conference 2025*, pages 4206–4225.
- Yunfan Gao, Yun Xiong, Xinyu Gao, Kangxiang Jia, Jinliu Pan, Yuxi Bi, Yixin Dai, Jiawei Sun, Haofen

- Wang, Haofen Wang, and 1 others. 2023. Retrieval-augmented generation for large language models: A survey. *arXiv preprint arXiv:2312.10997*, 2(1):32.
- Lei Huang, Weijiang Yu, Weitao Ma, Weihong Zhong, Zhangyin Feng, Haotian Wang, Qianglong Chen, Weihua Peng, Xiaocheng Feng, Bing Qin, and 1 others. 2025. A survey on hallucination in large language models: Principles, taxonomy, challenges, and open questions. *ACM Transactions on Information Systems*, 43(2):1–55.
- Yi Jiang, Sendong Zhao, Jianbo Li, Haochun Wang, and Bing Qin. 2025. Gainrag: Preference alignment in retrieval-augmented generation through gain signal synthesis. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 10746–10757.
- Vladimir Karpukhin, Barlas Oguz, Sewon Min, Patrick Lewis, Ledell Wu, Sergey Edunov, Danqi Chen, and Wen-tau Yih. 2020. Dense passage retrieval for open-domain question answering. In *Proceedings of the 2020 conference on empirical methods in natural language processing (EMNLP)*, pages 6769–6781.
- Omar Khattab and Matei Zaharia. 2020. Colbert: Efficient and effective passage search via contextualized late interaction over bert. In *Proceedings of the 43rd International ACM SIGIR conference on research and development in Information Retrieval*, pages 39–48.
- Tom Kwiatkowski, Jennimaria Palomaki, Olivia Redfield, Michael Collins, Ankur Parikh, Chris Alberti, Danielle Epstein, Illia Polosukhin, Jacob Devlin, Kenton Lee, and 1 others. 2019. Natural questions: a benchmark for question answering research. *Transactions of the Association for Computational Linguistics*, 7:453–466.
- Woosuk Kwon, Zhuohan Li, Siyuan Zhuang, Ying Sheng, Lianmin Zheng, Cody Hao Yu, Joseph Gonzalez, Hao Zhang, and Ion Stoica. 2023. Efficient memory management for large language model serving with pagedattention. In *Proceedings of the 29th symposium on operating systems principles*, pages 611–626.
- Kenton Lee, Ming-Wei Chang, and Kristina Toutanova. 2019. Latent retrieval for weakly supervised open domain question answering. In *Proceedings of the 57th annual meeting of the association for computational linguistics*, pages 6086–6096.
- Juncen Li, Robin Jia, He He, and Percy Liang. 2018. Delete, retrieve, generate: a simple approach to sentiment and style transfer. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 1865–1874.
- Chin-Yew Lin. 2004. Rouge: A package for automatic evaluation of summaries. In *Text summarization branches out*, pages 74–81.
- Ao Liu, An Wang, and Naoaki Okazaki. 2022. Semi-supervised formality style transfer with consistency training. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 4689–4701.
- Junlong Liu, Yue Ma, Ruihui Zhao, Junhao Zheng, Qianli Ma, and Yangyang Kang. 2025. Listcon-ranker: A contrastive text reranker with listwise encoding. *arXiv preprint arXiv:2501.07111*.
- Ilya Loshchilov and Frank Hutter. 2017. Decoupled weight decay regularization. *arXiv preprint arXiv:1711.05101*.
- Xueguang Ma, Liang Wang, Nan Yang, Furu Wei, and Jimmy Lin. 2024. Fine-tuning llama for multi-stage text retrieval. In *Proceedings of the 47th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 2421–2425.
- Xueguang Ma, Xinyu Zhang, Ronak Pradeep, and Jimmy Lin. 2023. Zero-shot listwise document reranking with a large language model. *arXiv preprint arXiv:2305.02156*.
- Sourabrata Mukherjee and Ondrej Dušek. 2024. Text style transfer: An introductory overview. *arXiv preprint arXiv:2407.14822*.
- Phil Ostheimer, Mayank Nagda, Marius Kloft, and Sophie Fellenz. 2023. A call for standardization and validation of text style transfer evaluation. In *Findings of the Association for Computational Linguistics: ACL 2023*, pages 10791–10815.
- Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, Ilya Sutskever, and 1 others. 2019. Language models are unsupervised multitask learners. *OpenAI blog*, 1(8):9.
- Pranav Rajpurkar, Jian Zhang, Konstantin Lopyrev, and Percy Liang. 2016. Squad: 100,000+ questions for machine comprehension of text. In *Proceedings of the 2016 conference on empirical methods in natural language processing*, pages 2383–2392.
- Ruiyang Ren, Yuhao Wang, Yingqi Qu, Wayne Xin Zhao, Jing Liu, Hua Wu, Ji-Rong Wen, and Haifeng Wang. 2025. Investigating the factual knowledge boundary of large language models with retrieval augmentation. In *Proceedings of the 31st International Conference on Computational Linguistics*, pages 3697–3715.
- Xubin Ren, Wei Wei, Lianghao Xia, Lixin Su, Suqi Cheng, Junfeng Wang, Dawei Yin, and Chao Huang. 2024. Representation learning with large language models for recommendation. In *Proceedings of the ACM web conference 2024*, pages 3464–3475.
- Stephen Robertson and Hugo Zaragoza. 2009. *The probabilistic relevance framework: BM25 and beyond*, volume 4. Now Publishers Inc.

- Weijia Shi, Sewon Min, Michihiro Yasunaga, Minjoon Seo, Richard James, Mike Lewis, Luke Zettlemoyer, and Wen-tau Yih. 2024. Replug: Retrieval-augmented black-box language models. In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pages 8371–8384.
- Weiwei Sun, Lingyong Yan, Xinyu Ma, Shuaiqiang Wang, Pengjie Ren, Zhumin Chen, Dawei Yin, and Zhaochun Ren. 2023. Is chatgpt good at search? investigating large language models as re-ranking agents. In *Proceedings of the 2023 conference on empirical methods in natural language processing*, pages 14918–14937.
- Wenhui Wang, Furu Wei, Li Dong, Hangbo Bao, Nan Yang, and Ming Zhou. 2020. Minilm: Deep self-attention distillation for task-agnostic compression of pre-trained transformers. *Advances in neural information processing systems*, 33:5776–5788.
- Shitao Xiao, Zheng Liu, Peitian Zhang, Niklas Muenighoff, Defu Lian, and Jian-Yun Nie. 2024. C-pack: Packed resources for general chinese embeddings. In *Proceedings of the 47th international ACM SIGIR conference on research and development in information retrieval*, pages 641–649.
- Soyoung Yoon, Eunbi Choi, Jiyeon Kim, Hyeongu Yun, Yireun Kim, and Seung-won Hwang. 2024. Listt5: Listwise reranking with fusion-in-decoder improves zero-shot retrieval. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2287–2308.
- Lingxi Zhang, Yue Yu, Kuan Wang, and Chao Zhang. 2024a. Arl2: Aligning retrievers with black-box large language models via self-guided adaptive relevance labeling. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 3708–3719.
- Tianjun Zhang, Shishir G Patil, Naman Jain, Sheng Shen, Matei Zaharia, Ion Stoica, and Joseph E Gonzalez. 2024b. Raft: Adapting language model to domain specific rag. *arXiv preprint arXiv:2403.10131*.
- Tianyi Zhang, Varsha Kishore, Felix Wu, Kilian Q Weinberger, and Yoav Artzi. 2019. Bertscore: Evaluating text generation with bert. *arXiv preprint arXiv:1904.09675*.
- Xin Zhang, Yanzhao Zhang, Dingkun Long, Wen Xie, Ziqi Dai, Jialong Tang, Huan Lin, Baosong Yang, Pengjun Xie, Fei Huang, and 1 others. 2024c. mgte: Generalized long-context text representation and reranking models for multilingual text retrieval. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing: Industry Track*, pages 1393–1412.
- Yichi Zhang, Zhuo Chen, Yin Fang, Yanxi Lu, Li Fangming, Wen Zhang, and Huajun Chen. 2024d. Knowledgeable preference alignment for llms in domain-specific question answering. In *Findings of the Association for Computational Linguistics: ACL 2024*, pages 891–904.
- Honglei Zhuang, Zhen Qin, Rolf Jagerman, Kai Hui, Ji Ma, Jing Lu, Jianmo Ni, Xuanhui Wang, and Michael Bendersky. 2023. Rankt5: Fine-tuning t5 for text ranking with ranking losses. In *Proceedings of the 46th international ACM SIGIR conference on research and development in information retrieval*, pages 2308–2313.

## Appendices

### A Data Analysis and Evaluation

#### A.1 Data Generation

Our experiments are conducted on the NQ dataset (Kwiatkowski et al., 2019), which comprises anonymized user queries paired with Wikipedia articles. To adapt NQ for the reranking task, we execute a rigorous data construction pipeline consisting of four stages.

**Corpus Segmentation.** We derive the retrieval corpus from the Dec. 20, 2018, Wikipedia dump (Lee et al., 2019). To ensure data quality, we filter out semi-structured noise, such as tables, info boxes, lists, and disambiguation pages. The remaining articles are segmented into non-overlapping blocks of 100 words. Each passage  $p_i$  is prefixed with its article title in the format “[Title] [Content]”. We utilize a pre-trained Dense Passage Retriever (DPR) (Karpukhin et al., 2020) model to recall candidates, establishing a candidate knowledge corpus for subsequent sample mining.

**Passage-level Sample Mining.** Since NQ provides page-level annotations, we map ground-truth answers to specific passage blocks within the candidate knowledge corpus. For each query  $q$ , passages containing the exact knowledge required to trigger correct LLM reasoning are labeled as positive samples  $p_i^+$ . To mine hard negative samples, we employ a hybrid strategy combining BM25 (Robertson and Zaragoza, 2009) with LLM-based verification. Specifically, we retrieve the Top-100 passages for each query and sample negatives  $p_i^-$  that mislead correct LLM generation.

**List-level Sample Mining.** For each query, we retrieve the Top- $K$  ( $K = 50$ ) most relevant passages to form the candidate list  $\mathcal{C}_q =$

Table 4: Dataset statistics (# is the symbol of ‘the number of’).

Split	# Samples	Percentage
Training	16 094	77.07%
Validation	1788	8.56%
Test	3000	14.37%

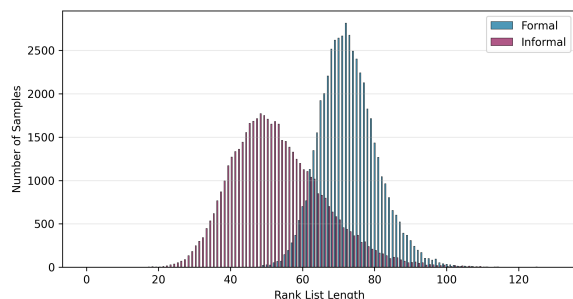


Figure 7: Rank List Length Distribution.

$\{p_1, p_2, \dots, p_{50}\}$ . To learn the global ranking logic, we employ a conversational prompting strategy that guides the target LLM to re-rank these candidates based on their relative evidential effectiveness within the given context.

**Style Augmentation.** We further augment the candidate set with stylistic variations. Specifically, passages sampled in their original formal style are rewritten into semantically similar informal variants using an LLM-based style transfer process. These rewritten passages are then injected back into the candidate list, forming the final augmented dataset. This process preserves semantic content while varying style, ensuring that observed performance reflects stylistic sensitivity rather than differences in knowledge.

**Data Statistics.** The final dataset is structured as query-context pairs, resulting in 16,094 valid training instances. The processed dataset is partitioned into training, validation, and test sets as detailed in Table 4. We further analyze the length distributions of formal and informal passages, as illustrated in Fig. 7. The formal passages exhibit an average length of 634.85 tokens, whereas the informal passages are comparatively shorter, with an average length of 535.39 tokens. The observed distributions confirm that colloquial variations tend to be more compact than encyclopedic text, reflecting the typical length characteristics of realistic QA scenarios in practice.

## A.2 Automatic Evaluation

Following prior work (Ostheimer et al., 2023), we comprehensively assess the quality of generated style-variant pairs across three dimensions: style transfer strength, content preservation, and fluency.

**Style Transfer Strength.** We adopt a Qwen2.5-14B model as a binary style classifier. First, to ensure the reliability of this evaluator, we validate it on the formal test set from the raw corpus, where it achieves an F1-score of 95.23%. Subsequently, we apply it to our generated informal paragraphs. The model classifies 99.97% of the generated samples as belonging to the informal category, confirming that the generated paragraphs are effectively injected with distinct stylistic expressions.

**Content Preservation.** To verify that the effective knowledge remains effectiveness during style transfer, we measure the semantic similarity between the source formal text and the generated informal text using BERTScore. The dataset achieves a similarity score of 0.6686, indicating that the essential factual information is well-preserved during the rewriting process.

**Fluency.** We assess the linguistic quality of the generated text by calculating the Perplexity (PPL) using a pre-trained GPT-2 (Radford et al., 2019). The generated paragraphs yield a low PPL score of 33.38, suggesting that the generated informal text maintains high grammatical fluency.

## A.3 Human Evaluation

To manually verify the correctness and quality of generated data, we conducted a manual validation on a randomly sampled 20% subset of the informal paragraphs. Four human annotators, all holding graduate-level degrees, independently evaluated each paragraph after reviewing a brief annotation guideline. The evaluation was performed along three dimensions, including style type, content preservation, and fluency. The average annotation time was approximately 2 minutes per instance, and disagreements were resolved through discussion. The final labeling consistency was assessed on the sample set, yielding a Kappa score of 0.719.

Furthermore, we report the full text of the instructions given to the annotators, explicitly detailing the instructions used to guide the manual quality check.

### Instruction of Data Annotation

Thank you for participating in our evaluation task. In this task, you will be presented with a pair of passages: an original Formal Passage sourced from Wikipedia and a generated Informal Passage rewritten by an Artificial Intelligence Language Model. Your goal is to assess the quality of the generated informal passage based on three specific dimensions, including Style Transfer Strength, Content Preservation, and Fluency.

Please read the definitions of each dimension carefully and label each sample accordingly.

#### 1. Style Transfer Strength

Determine whether the generated passage successfully reflects an Informal style.

**Formal Style:** Adopts a rigorous, objective, and academic tone. Uses standard grammar and avoids colloquialisms (e.g., Wikipedia articles, news reports).

**Informal Style:** Utilizes a casual, conversational, or emotional tone, often incorporating slang and spoken expressions (e.g., social media posts, forum comments).

#### 2. Content Preservation

The goal of the rewriting process is to change the style while strictly keeping the core semantic meaning unchanged. The informal passage must contain the same key factual information (entities, numbers, events) as the original formal passage.

#### 3. Fluency

Read the Informal Passage and assess the grammatical correctness and readability of the generated text.

Please carefully read the following annotation example, which presents the stylistic expressions and their corresponding labels for your reference.

**Formal Text:** *Big Little Lies is an American drama television series based on the novel of the same name by Liane Moriarty. It received 16 Emmy Award nominations and won eight.*

**Informal Text:** *Omg, have you guys seen Big Little Lies?? It's this total drama fest based*

*on Liane Moriarty's novel. Can you believe it snagged a whopping 16 Emmy noms and took home eight trophies? It is awesome!*

Style Transfer Strength: [] Informal / [] Formal

Content Preservation: [] Preserved / [] Distorted

Fluency: [] Fluent / [] Incoherent

### A.4 Prompt Templates

We adopt formal and informal styles to reflect differences in writing commonly observed across mixed-style retrieval sources. To transfer the formal Wikipedia text into a more relaxed and conversational style while preserving its original meaning, we use the following prompt:

#### Informal Content Generation Prompt

You are a great text style transfer assistant. Transform the following formal paragraph into a super informal tone. Use everyday language, emotional expressions, and even rhetorical flair. No fancy words, no academic tone, JUST super relaxed, fun, and super easy to read in English. Keep the output between 80 and 120 words.

**Paragraph:** "Big Little Lies (TV series)" It received 16 Emmy Award nominations and won eight, including Outstanding Limited Series and acting awards for Kidman and Dern ...

**Output:** Dude, Big Little Lies absolutely crushed it! I mean, seriously, can we talk about how this show snagged a whopping sixteen Emmy nominations ...

To assess the style transfer strength of the generated content, we use a prompt that instructs the LLM to classify whether a given paragraph is formal or informal.

#### Style classification Prompt

You are a great style classification assistant. Your task is to determine whether the given paragraph is written in a formal or informal style. A formal paragraph adopts a rigorous tone and avoids colloquial phrases or expres-

sive phrasing, whereas an informal paragraph uses casual, conversational, or emotional language. Read the paragraph carefully and output only one label: Formal or Informal.

**Paragraph:** “Big Little Lies (TV series)” It received 16 Emmy Award nominations and won eight, including Outstanding Limited Series and acting awards for Kidman and Dern ...

**Output:** Formal

## B Details on Experiment Setup

### B.1 Implementation Details

Following prior work (Dong et al., 2023; Askari et al., 2024; Dong et al., 2025), we utilize DPR as the retriever to retrieve candidate passages for each query. During training, we employ a stratified sampling strategy to construct list-level inputs. Specifically, for each training and validation query, we recall the Top-50 passages and sample  $u = 6$  representative passages at fixed intervals (ranks 1, 10, 20, 30, 40, 50) to discriminate between high-relevance candidates across varying ranking positions. For inference, the full Top-50 candidate list retrieved by DPR is fed into the reranker.

We employ BGE-Reranker-Base (Xiao et al., 2024) as our reranker backbone. The model is fine-tuned for 30 epochs with a batch size of 8, optimized via AdamW (Loshchilov and Hutter, 2017) with a learning rate of  $1 \times 10^{-6}$ . To balance the optimization dynamics between passage-level classification and list-level ranking, we incorporate an adaptive loss weighting mechanism.

Our primary evaluation metrics focus on the downstream generation quality of LLMs. Our main experiments report the performance based on the Top-5 reranked passages. All experiments are primarily conducted using Qwen2.5-14B, deployed via the vLLM (Kwon et al., 2023) framework for high-throughput inference. To assess generalization, we further extend our evaluation to the LLaMA-3.1, Gemma-3, and Qwen3 families across various scales. All reported results are averaged over three runs. Our dataset and code are publicly available<sup>6</sup>.

### B.2 Loss Analysis

We implement an adaptive gradient alignment mechanism to address the potential optimization

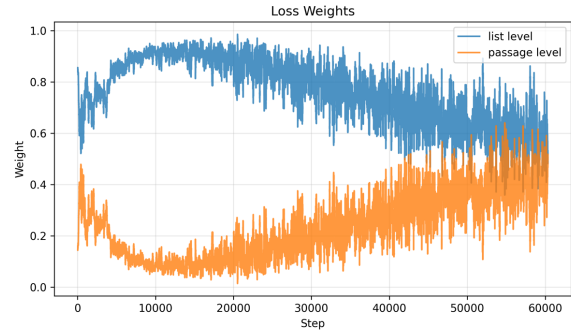


Figure 8: The weights of losses.

imbalance between passage-level classification and list-level ranking. As illustrated in Fig. 8, our dynamic weight calculator initially prioritizes the list-level objective to establish a coarse-grained ranking structure. As training progresses into the later stages, the mechanism automatically increases the weight of the passage-level loss, shifting the focus toward fine-grained knowledge verification. This dynamic shift reflects a coarse structural ranking to a precise knowledge verification optimization, which confirms the effectiveness of our model design.

### B.3 Baselines

We compare SARK against six representative reranking baselines spanning different ranking paradigms, including encoder-based reranker, and large-scale generative rerankers:

- **BGE-Reranker-Base** (Xiao et al., 2024) is a cross-encoder reranking model based on the BERT architecture. It jointly encodes query-passage pairs and captures token-level interactions for relevance estimation.
- **BGE-Reranker-Large** (Xiao et al., 2024) is a larger variant of the BGE architecture with increased model capacity. It provides stronger semantic modeling and improves ranking accuracy across diverse benchmarks.
- **mGTE** (Zhang et al., 2024c) is an encoder-based text embedding model that can be applied to reranking via similarity scoring. It addresses the length constraints of standard encoders, better supporting native processing of long-form text.
- **MiniLM** (Wang et al., 2020) is a lightweight reranker that utilizes a distilled Transformer model for reranking. It achieves a favorable

<sup>6</sup><https://github.com/lambdarw/SARK>

balance between efficiency and effectiveness, making it suitable for latency-sensitive retrieval scenarios.

- **RankGPT** (Sun et al., 2023) is a generative approach that leverages prompts to perform reranking. It processes candidate passages in a sliding-window manner to handle longer input contexts. In our implementation, we use Qwen2.5-7B as its base model.
- **RankLlama** (Ma et al., 2024) is a generative reranker fine-tuned from Llama-7B. It leverages the reasoning capacity of a decoder-only LLM to generate ranking signals, achieving strong performance among large-scale generative rerankers.

#### B.4 Prompt Templates

For passage-level signal generation, we obtain LLM responses under both conditions: with retrieval-augmented passages and without retrieval-augmented passages to isolate the contribution of individual passages. The corresponding prompt template is presented as follows:

##### Passage-level Knowledge Signal Extraction Prompt

You are an answer generation assistant. You need to answer the <Question> based on the <Context> with a brief response. You should only answer a few words.

**Question:** Big little lies season 2 how many episodes?

**Context:** “Big Little Lies (TV series)” It received 16 Emmy Award nominations and won eight, including Outstanding Limited Series and acting awards for Kidman and Dern ...

**Output:** 7 episodes

Furthermore, answers failing the EM constraint undergo a secondary semantic assessment by an LLM to verify their contextual correctness. The prompt template is structured as follows:

##### Effective Knowledge Evaluation Prompt

You are an answer evaluation assistant. Evaluate the <LLM’s Answer> by checking if the <LLM’s Answer> matches the <Question>

and <Groundtruth>. Respond with ‘Match’ if identical, ‘Partial’ if partially correct, or ‘Mismatch’ if incorrect. Use only one word.

**Question:** Big little lies season 2 how many episodes?

**LLM’s Answer:** 7 episodes

**Groundtruth:** seven

**Your Evaluation:**

**Output:** Match

For list-level signal generation, to handle context window limitations of the LLM, we adopt a multi-turn conversational prompting strategy that guides LLM to inspect each recalled passage sequentially. The prompt template is as follows:

##### List-level Knowledge Signal Extraction Prompt

**Instruct:** You are a passages reranker. Your task is to rank the provided passages based on the given question and groundtruth answer. Passages that can directly support the answer should be ranked first.

I will give you {num} passages. Please rank them based on the question: {question} and groundtruth answer: {groundtruth}.

**LLM:** Ok, please provide the passages.

**Instruct:** [1] {passage1}

**LLM:** Received passage [1]

**Instruct:** [2] {passage2}

**LLM:** Received passage [2]

...

**Instruct:** Return the final ranking in descending order using format: [X] > [Y] > ...

**Output:** [2] > [3] > ... > [1]

## C Evaluation on Multiple Styles

### C.1 Style Transfer Setup

To further evaluate the robustness of our approach under diverse stylistic variations, we respectively conduct additional experiments on multiple writing styles, including *Romantic* and *Humorous*, following the established methodology in (Li et al., 2018). These styles represent expressive linguistic variations that differ substantially from the neu-

Table 5: Examples of style transfer with different stylistic expressions.

Style	Example
<b>Raw</b>	"Spooks (series 2)" the seventh episode rose to 6.6 million. Overnight ratings rose to seven million for the second series finale. With consolidated figures factored in, the second series was seen by a per-episode average of 7.19 million viewers. The second series received generally favourable reviews. Gord Lacey of TVShowsOnDVD called the second series "such a fun series", a "plausible spy show" and ...
<b>Humor</b>	<p><i>Prompt: You are a great text style transfer assistant. Rewrite the following formal paragraph into a slightly humor descriptive tone. Avoid idioms, exaggerated metaphors, or informal slang. Do not introduce any figurative expressions that could change the original meaning or tone. Do not add, remove, or change any entities, numbers or dates. Maintain approximately the same length as the original paragraph. Do not include explanations, notes, or other languages. Output ONLY the rewritten paragraph in English.</i></p> <p><i>Output:</i> In <a href="#">the grand adventure of "Spooks" (Series 2)</a>, the seventh episode saw its audience swell to an impressive 6.6 million viewers. By the time the curtain fell on the series finale, overnight ratings had skyrocketed to a stellar seven million. Including <a href="#">all those late-night binge-watchers</a>, each episode managed to attract <a href="#">a whopping average of 7.19 million pairs of eyes</a>. Critics were quite pleased with Series 2 too; Gord Lacey at TVShowsOnDVD couldn't help but call it "such a fun series," praising it as a "plausible spy show" and ...</p>
<b>Romance</b>	<p><i>Prompt: You are a great text style transfer assistant. Rewrite the following formal paragraph into a slightly romanticized descriptive tone. Preserve all original facts and meaning and Avoid excessive metaphors. Do not add, remove, or change any entities, numbers or dates. Maintain approximately the same length as the original paragraph. Do not include explanations, notes, or other languages. Output ONLY the rewritten paragraph in English.</i></p> <p><i>Output:</i> The seventh episode <a href="#">enchanted 6.6 million hearts</a>. The second series finale <a href="#">bewitched an overnight audience of seven million souls</a>. When consolidated figures were accounted for, each episode of the second series captivated an average of 7.19 million viewers. <a href="#">Critics embraced the second series with warm praise</a>. Gord Lacey of TVShowsOnDVD hailed the second series as "such a delightful journey," describing it as a "<a href="#">believable espionage saga</a>" and "an excellent show ...</p>

Table 6: Top-50 performance comparison on diverse styles.

Model	Style: Humor				Style: Romance			
	Accuracy	EM	BERTScore	ROUGE	Accuracy	EM	BERTScore	ROUGE
BGE-Base	0.5012	0.3266	0.7259	0.5557	0.4745	0.3139	0.7144	0.5282
BGE-Large	0.4708	0.3082	0.7083	0.5264	0.4867	0.3129	0.7153	0.5372
MiniLM	0.5397	0.3512	0.7410	0.5864	0.5398	0.3462	0.7395	0.5825
mGTE	0.5453	0.3569	0.7437	0.5946	0.5440	0.3466	0.7378	0.5827
RankGPT	0.5295	0.3402	0.7375	0.5806	0.5267	0.3382	0.7352	0.5731
RankLlama	0.5472	<b>0.3572</b>	<b>0.7465</b>	0.5960	0.5380	0.3492	0.7397	0.5837
<b>SARK</b>	<b>0.5503</b>	0.3545	0.7457	<b>0.5960</b>	<b>0.5529</b>	<b>0.3555</b>	<b>0.7458</b>	<b>0.5940</b>

tral factual tone commonly found in the original dataset. We employ Qwen2.5-32B-Instruct<sup>7</sup> to perform style transfer on the test dataset. Table 5 presents several representative examples illustrating this transformation, which demonstrate that the Humorous and Romantic variants preserve the original content while altering the narrative tone and expression.

## C.2 Style Transfer Evaluation

The quality of the generated passages is evaluated following the method described above. First, we employ BERTScore to measure semantic similarity

between the original passages and their rewritten counterparts for humor and romance styles, yielding scores of 0.7438 and 0.7501, respectively. Second, PPL is used to evaluate the fluency of the generated humor and romance passages. PPL scores of 43.60 and 48.14 indicate that the generated texts maintain high grammatical quality. Moreover, we assess stylistic variation in humor and romance using TextBlob, where changes in mean sentiment scores indicate an overall shift toward the target style, and variations in standard deviation reflect the concentration of affective expression. After style transfer, the average sentiment of the passages increased from 0.1188 to 0.1428 (for humor)

<sup>7</sup><https://huggingface.co/Qwen/Qwen2.5-32B-Instruct>

and 0.1567 (for romance), while the standard deviation of sentiment slightly fluctuated to 0.96% and 0.05%, respectively. This indicates that generated texts tend to exhibit a more positive overall tone, with a more concentrated sentiment distribution.

### **C.3 Results under Style Shift**

We evaluated SARK against several strong baselines, as shown in Table 6. The evaluation is conducted on humor and romance data respectively, and answer performance is measured based on the top-50 reranked passages. Overall, SARK is more robust in handling stylistic variations compared with existing methods.

For the humor style, SARK achieves the highest accuracy of 0.5503, slightly outperforming RankLlama with 0.5472 and mGTE with 0.5453. Although SARK does not obtain the best EM score, it remains highly competitive. Meanwhile, SARK achieves comparable performance to RankLlama, with a BERTScore of 0.7457 and a ROUGE score of 0.5960, maintaining a comparable result to RankLlama at the lexical level. In contrast, under the romance style, SARK shows more substantial advantages. It improves accuracy by 0.89% over mGTE and 1.49% over RankLlama, demonstrating a clear advantage in this setting.