Self-Improving Sparse Retrieval Through Heuristic Representation Refinement and Representation-Focused Learning

Xiaojing Li[1], **Bin Wang**[1*], **Xiaochun Yang**[1], **Meng Luo**[2]

[1]Northeastern University, China

[2]The State Key Laboratory of Blockchain and Data Security (Zhejiang University), China

xiaojingli1220@gmail.com, {binwang, yangxc}@mail.neu.edu.cn, meng.luo@zju.edu.cn

Abstract

Learnable sparse retrieval (LSR) models encode texts into high-dimensional sparse representations, supporting token-level expansion beyond the original text and addressing the vocabulary mismatch problem in traditional bag-of-words retrieval. However, in the absence of representation-level supervision, these representations usually overemphasize irrelevant tokens while neglecting truly relevant ones. We term this phenomenon the **Representation Hallucination** problem in LSR models, a critical bottleneck impeding accurate retrieval. To address this challenge, we introduce SIRE, a self-improving training framework for sparse retrieval that integrates two core strategies: **Heuristic Representation Refinement** and **Representation-Focused Learning**. Specifically, SIRE first identifies and corrects representation hallucinations in the outputs of the current LSR model using heuristic methods. The resulting representations serve as the primary supervision signals, guiding a pretrained language model (*e.g.,* BERT) to mitigate the problem directly at the representation level. This process can be iterated, enabling progressive model improvement. Extensive experiments on both in-domain and out-domain benchmarks show that SIRE produces higher-quality sparse representations, significantly enhancing retrieval performance over strong baselines.

1 Introduction

Sparse retrieval remains a widely used and competitive approach in information retrieval systems, supporting applications like search engines (Lin et al. 2021) and retrieval-augmented generation for large language models (LLMs) (Gao et al. 2023; Zhang et al. 2024, 2025). Compared with dense retrieval, sparse retrieval offers stronger interpretability and better domain generalization (Formal et al. 2021b). However, traditional bag-of-words models, such as BM25 (Robertson and Zaragoza 2009), suffer from the vocabulary mismatch problem, where relevant documents might not contain terms in the query (Kong et al. 2023). Learnable sparse retrieval (LSR) methods (MacAvaney et al. 2020; Mallia et al. 2021; Bai et al. 2020) address this limitation by employing pre-trained language models (PLMs), such as BERT (Devlin et al. 2019), to convert text into high-dimensional sparse representations, where each dimension represents the relevance between a particular vocabulary token and the input text. Essentially, these representations perform weighted vocabulary expansion to mitigate the vocabulary mismatch problem.

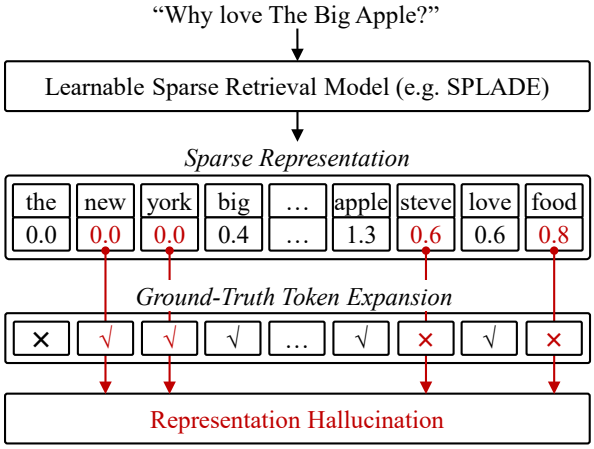

Figure 1: The representation hallucination problem in LSR models. The term "Big Apple" is widely recognized in English as a nickname for "New York".

Despite recent progress, current LSR models still struggle with a key problem: **Representation Hallucination**, where the model assigns high weights to irrelevant or low-impact tokens, or fails to emphasize semantically important ones. As a result, the overall quality of text representations declines, ultimately impairing retrieval effectiveness. A primary cause of representation hallucination is that *LSR models are typically trained solely with text-level supervision, which neglects guidance at the representation level*. For instance, the standard training paradigm for LSR typically relies on contrastive learning between relevant and irrelevant documents for a given query (Formal et al. 2021a). While text-level supervision yield some improvements, it does not provide direct feedback for optimizing representations. Consequently, models often fail to learn truly effective and discriminative representations. Furthermore, given the high dimensionality of sparse representations and the difficulty of quantifying the value of each dimension, it is impractical to manually annotate ideal representations to resolve this issue.

*Corresponding author.

In this work, we propose SIRE, a novel training framework for sparse retrieval that directly addresses representation hallucination. SIRE adopts a self-improving training strategy that progressively enhances LSR models through iterative cycles of **Heuristic Representation Refinement** and **Representation-Focused Learning**. For a given text input and its sparse representation produced by the current LSR model, such as SPLADE (Formal et al. 2021b), SIRE applies heuristic strategies to identify and adjust three types of tokens—removing irrelevant ones, reducing low-impact ones, and adding truly relevant ones. These refined representations serve as direct optimization targets for further training of LSR models at the representation level, complementing or substituting traditional retrieval signals (Formal et al. 2024). The trained LSR models are then reused in the next iteration, forming a loop of representation optimization and model learning. This iterative process drives continual improvements in sparse retrieval performance.

We evaluate SIRE on a broad range of retrieval benchmarks, covering both in-domain datasets such as MS MARCO (Bajaj et al. 2016), DL19 (Craswell et al. 2020) and DL20 (Craswell et al. 2021), and out-domain datasets from the BEIR benchmark (Thakur et al. 2021). Experimental results show that SIRE not only achieves state-of-the-art retrieval performance, but also consistently mitigates representation hallucination across various sparse retrievers, including SPLADE (Formal et al. 2021b) and SPLADE++ (Formal et al. 2022). These results underscore the effectiveness of our method in improving representation quality, as well as its strong generalization ability. Additionally, we conduct comprehensive ablation studies and detailed analysis to examine each component's contribution, shedding light on how the iterative refinement-learning loop drives performance gains and produces high-quality representations.

Our main contributions are as follows:

- We identify representation hallucination as a critical problem in existing LSR models and introduce a heuristic strategy to refine their sparse representations.
- We propose SIRE, a novel self-improving framework that alternates between heuristic representation refinement and representation-focused learning to optimize LSR models.
- Extensive experiments on MS MARCO, DL19, and DL20 benchmarks demonstrate the effectiveness of SIRE and its strong generalization ability to out-domain settings.

2 Methodology

As illustrated in Figure 2, SIRE adopts an iterative cycle that alternates between *heuristic representation refinement* and *representation-focused learning*. Starting with sparse text representations generated by a current LSR model $\mathcal{M}_0$, such as SPLADE (Formal et al. 2021b), SIRE first refines these representations using heuristic strategies. A pretrained language model $\mathcal{M}$ is subsequently trained to learn these refined representations, yielding an improved intermediate model $\mathcal{M}_1$. This updated model $\mathcal{M}_1$ replaces $\mathcal{M}_0$, and the cycle of refinement and learning is repeated. After t iterations, the process converges to the final model $\mathcal{M}_t$.

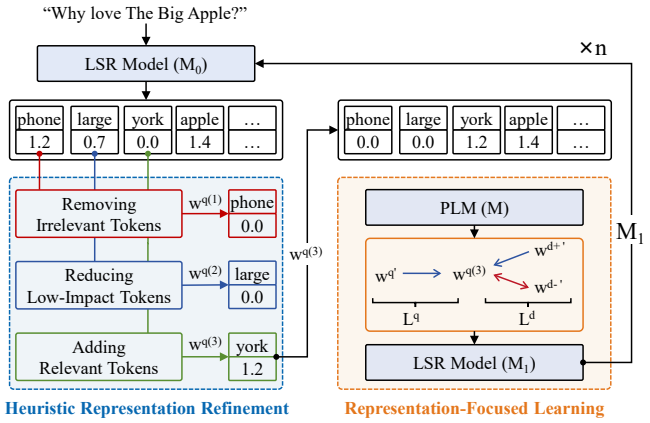

Figure 2: The overall architecture of SIRE, which entails n iterative cycles of heuristic representation refinement and representation-focused learning.

2.1 Background

LSR models encode input text t (*i.e.,* a query q or a document d) into a high-dimensional sparse vector $\boldsymbol{w}^t \in \mathbb{R}^V$, where V denotes the vocabulary. Each dimension w_i^t indicates the semantic relevance between the i-th vocabulary token and the input text, with $w_i^t = 0$ indicating irrelevance. Recent approaches like SPLADE (Formal et al. 2021b) leverage BERT's masked language modeling head to produce sparse representations. By selectively expanding semantically relevant tokens, LSRs achieve both high retrieval accuracy and enhanced interpretability.

2.2 Heuristic Representation Refinement

For a given text (*e.g.,* "*Why love The Big Apple?*"), tokens expanded from the current LSRs typically fall into three categories:

(1) **Irrelevant Tokens**: Tokens with little or no semantic connection to the input text. For instance, the token "phone" is unrelated to the context.

(2) **Low-Impact Tokens**: Tokens that are semantically related to the text but ubiquitous, offering limited retrieval utility (*e.g.,* "large").

(3) **Relevant Tokens**: Tokens that are both semantically aligned with the input and highly discriminative. For example, "New" and "York" accurately capture the meaning of "The Big Apple".

SIRE refines query representations $\boldsymbol{w}^q$ by suppressing irrelevant and low-impact tokens while emphasizing the most discriminative ones. The refinement proceeds through several transformation stages: $\boldsymbol{w}^q \rightarrow \boldsymbol{w}^{q(1)} \rightarrow \boldsymbol{w}^{q(2)} \rightarrow \boldsymbol{w}^{q(3)}$, where modifications are applied to each stage: irrelevant token removal, reduction of low-impact tokens, and augmentation of relevant tokens.

Removing Irrelevant Tokens. Typically, a token irrelevant to a document is also irrelevant to the corresponding query. To identify and remove such tokens, we use the model $\mathcal{M}_0$ to construct the sparse representations of the query and its

positive documents—denoted as $\boldsymbol{w}^q$ and $\mathcal{D} = \{\boldsymbol{w}^{d_k^+}\}_{k=1}^m$, respectively. We then apply max pooling to aggregate all document representations in $\mathcal{D}$, resulting in the overall representation $\boldsymbol{w}^{d^+}$ for $\mathcal{D}$. Finally, we eliminate query tokens that do not appear in $\boldsymbol{w}^{d^+}$:

$$\boldsymbol{w}_i^{q(1)} = \boldsymbol{w}_i^q \cdot \mathbb{1}(\boldsymbol{w}_i^{d^+} \neq 0), \tag{1}$$

where $\mathbb{1}$ denotes indicator function. When $\boldsymbol{w}_i^q \neq 0$ and $\boldsymbol{w}_i^{d^+} = 0$, the i-th token is removed from the query representation $\boldsymbol{w}_i^q$.

Reducing Low-Impact Tokens. We introduce a metric $\mathcal{F}$ to measure the contribution of each token to the retrieval results and use $\mathcal{F}$ to filter out low-impact query tokens. Given that LSR models often use the dot product between query and document representations as their similarity (Formal et al. 2021a), the definition of F and the filtering process for low-impact tokens are formulated as follows:

$$\mathcal{F}(\boldsymbol{w}_i^{q(1)}) = \frac{\boldsymbol{w}_i^{q(1)} \cdot \boldsymbol{w}_i^{d^+}}{\sum_{j=1}^{|V|} \boldsymbol{w}_j^{q(1)} \cdot \boldsymbol{w}_j^{d^+}} \tag{2}$$

$$\boldsymbol{w}_i^{q(2)} = \boldsymbol{w}_i^{q(1)} \cdot \mathbb{1}(\mathcal{F}(\boldsymbol{w}_i^{q(1)}) > \theta), \tag{3}$$

where the denominator in Eq. 2 represents the dot product between query and document representations, and θ is a predefined hyperparameter.

Adding Relevant Tokens. A major challenge for current LSR models is the limited diversity of positive documents in datasets like MS MARCO, which restricts the model's ability to fully identify relevant tokens. To address this issue, we utilize LLMs to generate more positive documents and extract additional query-relevant tokens from them.

We design the prompt template: "*Given a query [q], respond to this query from different perspectives*", to guide LLMs in generating diverse and semantically relevant documents through repeat sampling. By adding these generated documents to the original positive document set, we use the LSR model $\mathcal{M}_0$ and max-pooling to compute the representation of the updated positive document set, denoted as $\boldsymbol{w}^{\hat{d^+}}$. Tokens that rank in the top 10% of weights in $\boldsymbol{w}^{\hat{d^+}}$ but have zero weight in the query representation $\boldsymbol{w}^q$, form the candidate token set $\mathcal{T}$ for expansion. Their weights are set to the mean of the non-zero dimensions in $\boldsymbol{w}^{q(2)}$, as follows:

$$\boldsymbol{w}_i^{q(3)} = \begin{cases} \text{avg}(\boldsymbol{w}^{q(2)}), & \text{if } i \in \mathcal{T}, \\ \boldsymbol{w}_i^{q(2)}, & \text{otherwise.} \end{cases} \tag{4}$$

2.3 Representation-Focused Learning

Based on the optimized representation, we introduce two training objectives, $\mathcal{L}^q$ and $\mathcal{L}^d$, to guide the PLM $\mathcal{M}$ (*e.g.,* BERT) to produce high-quality sparse representations for both queries and documents.

The objective $\mathcal{L}^q$ employs the optimized representation $\boldsymbol{w}^{q(3)}$ as a supervision signal by minimizing the Kullback–Leibler (KL) divergence between the token-level distributions of $\boldsymbol{w}^{q'}$ (generated by $\mathcal{M}$) and $\boldsymbol{w}^{q(3)}$. The generation of $\boldsymbol{w}^{q'}$ by $\mathcal{M}$ is consistent with that in $\mathcal{M}_0$. The loss function is defined as follows:

$$\mathcal{L}^q = \sum_{i=1}^{|V|} \boldsymbol{w}_i^{q(3)} \log \frac{\boldsymbol{w}_i^{q(3)}}{\boldsymbol{w}_i^{q'}}. \tag{5}$$

$\mathcal{L}^d$ applies classic contrastive learning to refine document representations:

$$\mathcal{L}^d = -\log \frac{e^{\boldsymbol{w}^{q(3)} \cdot \boldsymbol{w}^{d^{+\prime}}}}{e^{\boldsymbol{w}^{q(3)} \cdot \boldsymbol{w}^{d^{+\prime}}} + e^{\boldsymbol{w}^{q(3)} \cdot \boldsymbol{w}^{d^{-\prime}}}}, \tag{6}$$

where d^+ and d^- are the positive and negative documents for the query, respectively, and $\boldsymbol{w}^{d^{+\prime}}$ and $\boldsymbol{w}^{d^{-\prime}}$ are their corresponding representations generated by $\mathcal{M}$.

In addition to the representation-focused objectives $\mathcal{L}^q$ and $\mathcal{L}^d$, we incorporate the retrieval-focused objective $\mathcal{L}^{q,d}$ from $\mathcal{M}_0$ for joint model optimization. The overall loss is defined as follows:

$$\mathcal{L} = \lambda(\mathcal{L}^q + \mathcal{L}^d) + (1 - \lambda)\mathcal{L}^{q,d}, \tag{7}$$

where λ is the trade-off hyper-parameter to balance the contributions of different loss terms.

2.4 Iterative Optimization Framework

For a weak initial model $\mathcal{M}_0$, a single round of heuristic representation refinement and representation-focused learning in SIRE is often insufficient to achieve optimal performance. Consequently, our approach iteratively alternates between these two stages to gradually improve representation quality and strengthen model performance. Specifically, in the t-th iteration, we apply heuristic refinement to the representations generated by the previous model $\mathcal{M}_{t-1}$. These refined representations are then used as supervision signals to fine-tune the PLMs, yielding an updated model $\mathcal{M}_t$. The newly trained model $\mathcal{M}_t$ subsequently generates representations for the next iteration, and the process repeats. After n iterations, we take $\mathcal{M}_n$ as the final model. In essence, SIRE allows for flexible iteration counts to compensate for the initial model's limitations. This design ensures that, even when starting from an under-performing initial model $\mathcal{M}_0$, sufficient iterations enables SIRE to yield performance comparable to models initialized from a stronger baseline.

3 Experiments

3.1 Experiment Settings

Datasets. Following SPLADE++ (Formal et al. 2022), we train our model on the public msmarco-hard-negatives dataset. The training set consists of 8.8 million documents and 500k queries, with each query associated with 50 hard negative documents. Our evaluation includes both in-domain and out-domain benchmarks. For in-domain evaluation, we use the MS MARCO development set, DL19 (Craswell et al. 2020) and DL20 (Craswell et al. 2021), all of which share the same document pool as the training set.

Out-domain performance is assessed using the BEIR benchmark (Thakur et al. 2021), including ArguAna, Climate-FEVER, DBPedia, FEVER, FiQA-2018, HotpotQA, NFCorpus, NQ, Quora, SCIDOCS, SciFact, TREC-COVID, and

Method	MS MARCO (In.)		DL19 (In.)		DL20 (In.)		BEIR (Out.)	
	MRR@10	R@1k	NDCG@10	R@1k	NDCG@10	R@1k	NDCG@10	R@100
• **Dense Retrieval Baselines**								
TAS-B	34.7	97.8	71.7	84.3	67.9	87.3	41.9	57.4
RocketQAv2	38.8	98.1	70.2	84.3	69.1	85.1	44.7	62.4
ColBERTv2	39.7	98.4	72.1	86.9	72.3	88.4	46.9	63.3
• **Sparse Retrieval Baselines**								
BM25	18.4	85.3	50.6	74.5	48.0	78.6	42.3	57.6
DeepCT	24.3	91.3	55.1	75.6	55.0	83.8	42.7	58.5
DeepImpact	32.6	94.8	69.5	79.4	65.1	83.4	43.0	60.4
SparseEmbed	39.2	98.1	71.4	87.6	74.7	88.5	46.5	63.2
SPLADE	32.2	95.5	66.5	81.3	68.8	85.1	43.2	61.7
SPLADE++	38.0	98.2	71.2	87.5	74.5	87.3	47.7	65.1
SPLADE-v3	40.2	98.7	72.6	88.1	75.4	90.4	48.2	66.4
• **SIRE (Ours)**								
$\mathcal{M}_0$: SPLADE	41.7	99.1	74.4	89.4	76.8	91.1	50.1	67.8
$\mathcal{M}_0$: SPLADE++	**42.1**	**99.5**	**75.1**	**90.2**	**78.1**	**92.8**	**51.3**	**68.4**

Table 1: Retrieval performance of various dense and sparse models on both in-domain benchmarks (In.: MS MARCO, DL19, DL20) and out-domain benchmarks (Out.: BEIR). Two versions of SIRE are evaluated, initialized with SPLADE (Formal et al. 2021b) and SPLADE++ (Formal et al. 2022) as $\mathcal{M}_0$ respectively. Following standard practice (Formal et al. 2021b), benchmark-specific evaluation metrics are used.

Touche-2020, which spans a diverse set of retrieval scenarios and domains. This benchmark is widely adopted for evaluating a model's generalization ability beyond the training distribution.

Implementation Details. We use SPLADE (Formal et al. 2021b) and SPLADE++ (Formal et al. 2022) respectively as the initial model $\mathcal{M}_0$ for the heuristic representation refinement. In this process, the threshold parameter θ for removing low-impact tokens is set to 0.2. Relevant token augmentation is performed using five positive documents generated by Qwen-14B (Team 2024). For representation-focused learning, we initialize the model $\mathcal{M}$ with BERT and set the hyperparameter λ to 0.6. During the iterative refinement phase, the maximum number of iterations n is set to 3. Training is conducted on two NVIDIA A100 GPUs with 80GB of memory. All other experimental settings follow those of SPLADE++ (Formal et al. 2022).

Baselines and Metrics. Our baselines are grouped into dense and sparse retrieval methods. Dense retrieval baselines include TAS-B (Hofstätter et al. 2021), RocketQAv2 (Ren et al. 2021), and ColBERTv2 (Santhanam et al. 2022). Sparse retrieval baselines include BM25 (Robertson and Zaragoza 2009), DeepCT (Dai and Callan 2020), DeepImpact (Mallia et al. 2021), SparseEmbed (Kong et al. 2023), SPLADE (Formal et al. 2021b), SPLADE++ (Formal et al. 2022), and SPLADE-v3 (Lassance et al. 2024).

We adopt standard metrics following established conventions (Formal et al. 2021b) for each benchmark: MRR@10 and R@1k for the MS MARCO benchmark; NDCG@10 and R@1k for the DL19 and DL20 benchmarks; and NDCG@10 and R@100 for the BEIR benchmark. We report the average results across the 13 datasets in the BEIR benchmark. This comprehensive metric selection ensures a thorough and accurate evaluation across benchmarks.

3.2 Main Results

In-Domain Evaluation. As shown in Table 1, our models outperform all surveyed baselines for the MS MARCO, DL19 and DL20 benchmarks. For example, SIRE with SPLADE as $\mathcal{M}_0$ achieves a 1.5% improvement in MRR@10 over SPLADE-v3 on MS MARCO. Using SPLADE++ as $\mathcal{M}_0$ yields a 1.9% improvement, further surpassing SPLADE-v3. This improvement is particularly significant considering that SPLADE-v3 uses more sophisticated training resources (Lassance et al. 2024), such as more powerful PLMs and knowledge distillation from cross-encoder re-rankers. In contrast, we deliberately employ only basic settings on SIRE to ensure a fair comparison with other baselines. Moreover, SIRE consistently improves both SPLADE and SPLADE++, demonstrating its broad applicability to different $\mathcal{M}_0$ configurations. The SPLADE++–based variant of our method also consistently outperforms the one based on SPLADE, likely due to the higher-quality initial representations provided by SPLADE++, which facilitates more effective learning.

Out-Domain Evaluation. While a performance decline in NDCG@10 is observed in out-domain settings—highlighting generalization challenges—our approach sustains significant advantages. On the BEIR benchmark, SIRE based on SPLADE achieves at least a 1.9% gain in NDCG@10 over all baselines. In addition, we observe that although the dense model TAS-B performs significantly better than the bag-of-words model BM25 in in-domain settings, the trend reverses in out-domain scenarios. Moreover, LSR models such as SparseEmbed generally outperform dense models in out-domain evaluations, likely because their high-dimensional representations are less prone to overfitting. These observa-

	MS MAR.	DL19	DL20	BEIR
SIRE (Ours)	**42.1**	**75.1**	**78.1**	**51.3**
• *Heuristic Representation Refinement*				
w/o RIT	40.7	73.2	76.5	49.0
w/o RLIT	41.5	74.1	77.3	50.2
w/o ART	41.1	73.6	76.9	49.4
• *Representation-Focused Learning*				
w/o $\mathcal{L}^q$	40.2	72.9	76.2	48.7
w/o $\mathcal{L}^d$	40.6	73.4	76.8	49.2
w/o $\mathcal{L}^{q,d}$	38.8	71.3	75.5	47.3
• *Iterative Pipeline Execution*				
w/o Iteration	40.9	73.6	76.9	49.4

Table 2: Ablation results of SIRE using SPLADE++ as $\mathcal{M}_0$, evaluating the contribution of each component from three perspectives: heuristic representation refinement, representation-focused learning, and iterative pipeline execution. MS MARCO (MS MAR.) results are reported using the MRR@10 metric, while NDCG@10 is used for other benchmarks. RIT, RLIT, and ART denote the three heuristic strategies introduced in Section 2.2: Removing Irrelevant Tokens, Reducing Low-Impact Tokens, and Adding Relevant Tokens, respectively.

tions highlight the strength of sparse retrieval models. Our SIRE, as an LSR model, not only inherits the generalization advantages of sparse models over dense ones but also enables consistent gains by using stronger $\mathcal{M}_0$ models—a key strength of our approach.

3.3 Ablation Analysis

Table 2 presents the ablation results of SIRE from three perspectives: heuristic representation refinement, representation-focused learning, and iterative pipeline execution. All variants of SIRE in our experiments adopt SPLADE++ as the initial sparse retriever $\mathcal{M}_0$.

Analysis of Heuristic Representation Refinement. Our heuristic refinement strategy includes three core techniques: removing irrelevant tokens (RIT), reducing low-impact tokens (RLIT), and adding relevant tokens (ART). The results show that ablating any individual component leads to a significant performance drop, demonstrating the unique contribution of each technique to the overall pipeline. Among the three, RIT yields the greatest improvement, indicating that suppressing erroneous expansion over irrelevant tokens is critical for improving sparse representation quality. Moreover, we observe that the benefits of this strategy are particularly pronounced in the out-domain setting (BEIR). Specifically, on MS MARCO, RIT, RLIT, and ART yield improvements of 1.4%, 0.6%, and 1.0% respectively, while on BEIR, the gains rise to 2.3%, 1.1%, and 1.9%. This suggests that our heuristic refinement helps mitigate overfitting on the training set, which is crucial for enhancing generalization.

Analysis of Representation-Focused Learning. The representation-focused learning component includes two representation-oriented loss functions ($\mathcal{L}^q$ and $\mathcal{L}^d$) and one retrieval-based loss ($\mathcal{L}^{q,d}$). Ablation results indicate that $\mathcal{L}^q$ offers more substantial improvements than $\mathcal{L}^d$. This is expected, as $\mathcal{L}^q$ directly aligns the refined query representations via a KL-divergence objective, while $\mathcal{L}^d$ updates document representations indirectly through contrastive learning. Crucially, despite the strength of representation-focused learning, traditional retrieval-based loss remains indispensable. These findings underscore the complementary nature of representation-based and retrieval-based learning—neither alone is sufficient for optimal performance.

	MS MAR.	DL19	DL20	BEIR
SPLADE	32.2	66.5	68.8	43.2
w/ HRR	33.2	68.0	69.9	44.9
(Δ_{SPLADE})	+1.0	+1.5	+1.1	+1.7
SPLADE++	38.0	71.2	74.5	47.7
w/ HRR	39.3	73.1	76.5	50.0
($\Delta_{\text{SPLADE++}}$)	+1.3	+1.9	+2.0	+2.3
SPLADE-v3	40.2	72.6	75.4	48.2
w/ HRR	41.7	74.8	77.6	50.9
($\Delta_{\text{SPLADE-v3}}$)	+1.5	+2.2	+2.2	+2.7

Table 3: Retrieval performance of baseline models using query representations refined by our heuristic representation refinement (HRR) strategy.

Analysis of Iterative Pipeline Execution. Our approach progressively optimizes sparse retrieval models through iterative application of heuristic refinement and representation-focused learning. The ablation study on the iteration reveals that this iterative process improves SIRE's performance from 40.9% to 42.1% on MS MARCO, and from 49.4% to 51.3% on BEIR, validating the effectiveness of this design. These results suggest that optimal representations cannot be reliably obtained through single-step optimization. Instead, our iterative strategy enables gradual convergence toward higher-quality representations

3.4 Discussion

Direct Validation of the Effectiveness of the Heuristic Representation Refinement. We refine the query representations generated by various baseline models using our heuristic strategy, and directly assess retrieval performance based on the refined representations. As shown in Table 3, all baseline models exhibit notable performance improvements when enhanced with our heuristic strategy. Two observations are particularly noteworthy. First, the effectiveness of our heuristic refinement strategy tends to increase with baseline quality. For example, on the MS MARCO dataset, applying our strategy yields gains of 1.0%, 1.3%, and 1.5% for SPLADE, SPLADE++, and SPLADE-v3, respectively. This ascending trend suggests that our strategy is particularly effective at further enhancing robust models by identifying and correcting hallucinated components within their representations. Second, the performance gains are even more pronounced on the out-domain BEIR benchmark than on the in-domain setting. This clearly highlights the strong generalization capability of our method.

Proportion of Tokens Targeted by Heuristic Representation Refinement. Figure 3 presents the proportions of

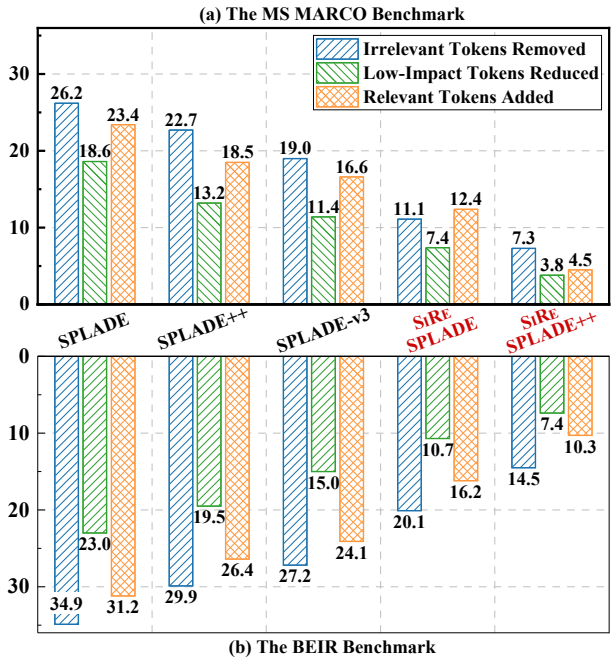

Figure 3: Proportions of three token categories—irrelevant tokens removed, low-impact tokens reduced, and newly added relevant tokens—relative to the total expanded tokens, across different models enhanced with our heuristic representation refinement.

irrelevant, low-impact, and added relevant tokens in the expanded representations generated by different models. A key observation is that the application of SiRe significantly reduces the presence of all three types of problematic tokens. This suggests that our approach contributes to higher-quality representations, which in turn directly benefits retrieval performance, as reflected in the retrieval gains reported in Table 1. Moreover, across all evaluated models, irrelevant tokens consistently account for the largest proportion among the three categories. This reveals that indiscriminate token expansion remains a major bottleneck for existing sparse retrieval systems. Encouragingly, our method effectively suppresses this category of tokens. Finally, across all models, the BEIR benchmark shows a noticeably higher proportion of hallucinated tokens compared to MS MARCO. This finding implies that representation hallucination and overfitting are more severe in the out-domain setting.

Trade-Off Hyper-Parameter λ Between Representation Learning and Retrieval Objectives. In Eq. 7, the hyper-parameter λ controls the trade-off between two learning objectives. To investigate the impact of this parameter, we systematically vary λ and evaluate the resulting performance of SiRe. As shown in Figure 4, increasing λ first improves retrieval effectiveness, which peaks between 0.5–0.6 before declining at higher values. Interestingly, the most significant performance gain occurs when λ increases from 0 to 0.1, highlighting the importance of our representation-focused objective. Moreover, for the SPLADE++-based variant of SiRe, we observe that while the performance on MS MARCO becomes stable as λ increases from 0.3 to 0.4, there are still notable improvements on the BEIR benchmark. This observation underlines that in-domain evaluation alone is insufficient to fully reflect the retrieval capability of a model.

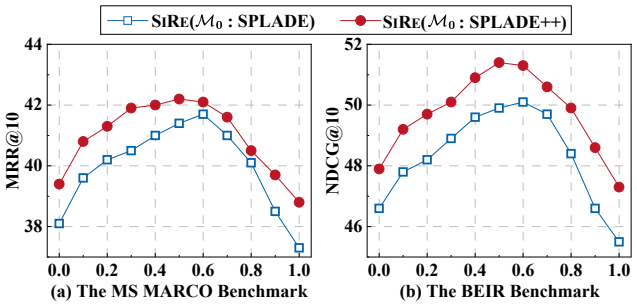

Figure 4: Performance of SiRe with different λ values in Eq. 7, evaluated on MS MARCO and BEIR.

Method	MS MARCO		BEIR	
	Query	Doc	Query	Doc
SPLADE	18	96	62	216
SPLADE++	43	123	82	214
SPLADE-v3	24	178	133	295
• **SiRe (Ours)**				
$\mathcal{M}_0$: SPLADE	**10**	**76**	**47**	**148**
$\mathcal{M}_0$: SPLADE++	**16**	**98**	**53**	**145**

Table 4: Average number of non-zero dimensions in the representations produced by different models.

Sparsity of Text Representations. Sparser representations generally lead to more efficient retrieval when using inverted indexes, as fewer non-zero terms reduce computation (Formal et al. 2021b). To assess representation sparsity, we calculate the average number of non-zero dimensions in the representations generated by different models on both the MS MARCO and BEIR benchmarks. The results in Table 4 show that SPLADE achieves reasonably good sparsity among the baselines, whereas SPLADE-v3 produces denser representations, despite its superior retrieval performance shown in Table 1. This indicates that previous baselines struggle to balance representation sparsity and retrieval quality. However, our SiRe effectively addresses this challenge, achieving the best performance on both aspects. In addition, across benchmarks, we observe that the representations tend to be denser on BEIR than on MS MARCO. This phenomenon stems from two primary factors: first, certain BEIR datasets (*e.g.,* ArguAna) contain longer input texts that naturally require denser representations; second, the out-domain nature of BEIR poses greater challenges to producing compact representations.

Effect of the Number of SiRe Iterations. We conduct a detailed analysis to examine the impact of the number of iterations on SiRe. As shown in Figure 5, increasing the number of iterations consistently brings notable performance gains, with results stabilizing after about four iterations. Two observations are worth emphasizing. First, although different initial $\mathcal{M}_0$ models introduce substantial performance gaps,

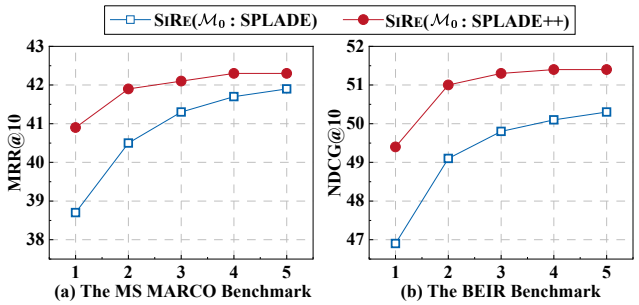

Figure 5: Effect of the number of iterative cycles, each comprising heuristic refinement followed by representation-focused learning, on retrieval performance.

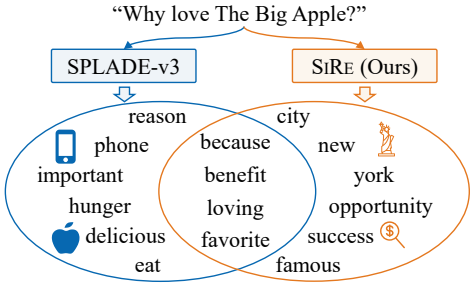

Figure 6: Venn diagram of the top-10 expanded tokens produced by SiRe (with SPLADE as $\mathcal{M}_0$) and SPLADE-v3 for the query "Why love The Big Apple?".

increasing the number of iterations gradually reduces these disparities. For instance, the performance difference between two model variants on MS MARCO drops from 2.2% in the first iteration to just 0.4% by the fifth. Second, the steeper curve observed in the out-domain setting indicates that the iterative process provides greater benefits for generalization than for in-domain performance. This further supports our claim that the proposed method effectively mitigates the risk of overfitting to training datasets.

Case Study. To better illustrate the advantages of the representations generated by SiRe, Figure 6 shows a Venn diagram comparing the token expansions produced by SiRe and SPLADE-v3 (Lassance et al. 2024) for the input query "Why love The Big Apple?". Both models expand the input with relevant but non-present tokens such as "because" and "benefit", demonstrating their ability to capture implicit meaning. However, a key difference emerges in how they interpret "Big Apple". SPLADE-v3 appears to rely on surface-level associations, primarily producing expansions related to literal apples or Apple products. In contrast, SiRe successfully expands to semantically deeper and more contextually accurate tokens like "New York", "success", and "opportunity"—concepts that reflect the underlying intent and cultural context of the query. This kind of meaningful expansion is particularly beneficial for information retrieval tasks. While a single example cannot fully establish superiority, the improvements across multiple benchmarks and metrics in Table 1 clearly demonstrate SiRe's consistent advantage over competing models.

4 Related Work

Current information retrieval methods are broadly classified into dense and sparse retrieval.

Dense Retrieval. Dense retrieval models encode text (*e.g.,* queries and documents) into low-dimensional vector representations. Existing studies generally adopt similar model architectures, with greater emphasis on optimizing the training objective (Karpukhin et al. 2020; Khattab and Zaharia 2020; Santhanam et al. 2022; Gao, Yao, and Chen 2021; Li et al. 2022; Xie et al. 2023; Gao and Callan 2022; Kang et al. 2025) and negative sampling strategies (Hofstätter et al. 2021; Ren et al. 2021). Additionally, Faggioli et al. (2024) explores methods to improve retrieval efficiency by selectively preserving the most informative dimensions in text representations.

Sparse Retrieval. Sparse Retrieval models map texts into high-dimensional vocabulary spaces. Our work focuses on sparse retrieval models, as they offer three main advantages over dense models (Formal et al. 2021a): (1) stronger interpretability; (2) better generalization on out-domain data; and (3) more efficient retrieval.

Recent advances in PLM-based LSR methods (Dai and Callan 2020; Gao, Dai, and Callan 2021; MacAvaney et al. 2020; Mallia et al. 2021; Formal et al. 2021b,a, 2022; Lassance et al. 2024) have focused on two aspects: term expansion and term importance estimation. DeepCT (Dai and Callan 2020) utilizes PLMs to reweight term importance. DeepImpact (Mallia et al. 2021) performs term expansion using large document collections and optimizes term weights. SPLADE (Formal et al. 2021b) combines masked language modeling head to achieve both term expansion and importance reweighting, generating high-dimensional sparse representations of texts. Subsequent studies have proposed improvements to the SPLADE series of methods (Formal et al. 2021a, 2022; Lassance et al. 2024) through enhancements in data preprocessing, PLM selection, and training strategies. Additionally, SparseEmbed (Kong et al. 2023) further enhances model expressiveness by constructing context-aware sparse lexical representations.

5 Conclusion

In this work, we identify a common issue in existing LSR models—Representation Hallucination, where models often assign weights to irrelevant tokens while overlooking relevant ones. To address this challenge, we propose SiRe, a novel training framework tailored for sparse retrieval. SiRe introduces a heuristic representation refinement strategy that identifies and reweights hallucinated dimensions in the sparse outputs of existing models such as SPLADE. The refined representations are then used as representation-level supervision signals, combined with standard retrieval objectives to guide model training. This refinement and training process is performed iteratively to progressively improve the quality of sparse representations. Comprehensive evaluations on both in-domain and out-domain benchmarks demonstrate that SiRe achieves state-of-the-art performance in information retrieval tasks.

Acknowledgments

The work is partially supported by the National Key Research and Development Program of China (2024YFF0617702), the National Natural Science Foundation of China (Nos. U22A2025, 62232007, U23A20309), 111 Project (No. B16009), and CCF Alibaba Cloud Yaochi Research Fund(No. CCF-Aliyun2024006).

References

Bai, Y.; Li, X.; Wang, G.; Zhang, C.; Shang, L.; Xu, J.; Wang, Z.; Wang, F.; and Liu, Q. 2020. SparTerm: Learning Term-based Sparse Representation for Fast Text Retrieval. *CoRR*, abs/2010.00768.

Bajaj, P.; Campos, D.; Craswell, N.; Deng, L.; Gao, J.; Liu, X.; Majumder, R.; McNamara, A.; Mitra, B.; Nguyen, T.; et al. 2016. Ms marco: A human generated machine reading comprehension dataset. *arXiv preprint arXiv:1611.09268*.

Craswell, N.; Mitra, B.; Yilmaz, E.; and Campos, D. 2021. Overview of the TREC 2020 deep learning track. *CoRR*, abs/2102.07662.

Craswell, N.; Mitra, B.; Yilmaz, E.; Campos, D.; and Voorhees, E. M. 2020. Overview of the TREC 2019 deep learning track. *CoRR*, abs/2003.07820.

Dai, Z.; and Callan, J. 2020. Context-Aware Term Weighting For First Stage Passage Retrieval. In *Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '20, 1533–1536. New York, NY, USA: Association for Computing Machinery. ISBN 9781450380164.

Devlin, J.; Chang, M.; Lee, K.; and Toutanova, K. 2019. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. In Burstein, J.; Doran, C.; and Solorio, T., eds., *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2019, Minneapolis, MN, USA, June 2-7, 2019, Volume 1 (Long and Short Papers)*, 4171–4186. Association for Computational Linguistics.

Faggioli, G.; Ferro, N.; Perego, R.; and Tonellotto, N. 2024. Dimension Importance Estimation for Dense Information Retrieval. In Yang, G. H.; Wang, H.; Han, S.; Hauff, C.; Zuccon, G.; and Zhang, Y., eds., *Proceedings of the 47th International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR 2024, Washington DC, USA, July 14-18, 2024*, 1318–1328. ACM.

Formal, T.; Lassance, C.; Piwowarski, B.; and Clinchant, S. 2021a. SPLADE v2: Sparse Lexical and Expansion Model for Information Retrieval. *CoRR*, abs/2109.10086.

Formal, T.; Lassance, C.; Piwowarski, B.; and Clinchant, S. 2022. From Distillation to Hard Negative Sampling: Making Sparse Neural IR Models More Effective. In Amigó, E.; Castells, P.; Gonzalo, J.; Carterette, B.; Culpepper, J. S.; and Kazai, G., eds., *SIGIR '22: The 45th International ACM SIGIR Conference on Research and Development in Information Retrieval, Madrid, Spain, July 11 - 15, 2022*, 2353–2359. ACM.

Formal, T.; Lassance, C.; Piwowarski, B.; and Clinchant, S. 2024. Towards effective and efficient sparse neural information retrieval. *ACM Transactions on Information Systems*, 42(5): 1–46.

Formal, T.; et al. 2021b. SPLADE: Sparse Lexical and Expansion Model for First Stage Ranking. In Diaz, F.; Shah, C.; Suel, T.; Castells, P.; Jones, R.; and Sakai, T., eds., *SIGIR '21: The 44th International ACM SIGIR Conference on Research and Development in Information Retrieval, Virtual Event, Canada, July 11-15, 2021*, 2288–2292. ACM.

Gao, L.; and Callan, J. 2022. Unsupervised Corpus Aware Language Model Pre-training for Dense Passage Retrieval. In Muresan, S.; Nakov, P.; and Villavicencio, A., eds., *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, 2843–2853. Dublin, Ireland: Association for Computational Linguistics.

Gao, L.; Dai, Z.; and Callan, J. 2021. COIL: Revisit Exact Lexical Match in Information Retrieval with Contextualized Inverted List. In Toutanova, K.; Rumshisky, A.; Zettlemoyer, L.; Hakkani-Tür, D.; Beltagy, I.; Bethard, S.; Cotterell, R.; Chakraborty, T.; and Zhou, Y., eds., *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2021, Online, June 6-11, 2021*, 3030–3042. Association for Computational Linguistics.

Gao, T.; Yao, X.; and Chen, D. 2021. SimCSE: Simple Contrastive Learning of Sentence Embeddings. In Moens, M.; Huang, X.; Specia, L.; and Yih, S. W., eds., *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing, EMNLP 2021, Virtual Event / Punta Cana, Dominican Republic, 7-11 November, 2021*, 6894–6910. Association for Computational Linguistics.

Gao, Y.; Xiong, Y.; Gao, X.; Jia, K.; Pan, J.; Bi, Y.; Dai, Y.; Sun, J.; Wang, H.; and Wang, H. 2023. Retrieval-augmented generation for large language models: A survey. *arXiv preprint arXiv:2312.10997*, 2(1).

Hofstätter, S.; Lin, S.; Yang, J.; Lin, J.; and Hanbury, A. 2021. Efficiently Teaching an Effective Dense Retriever with Balanced Topic Aware Sampling. In Diaz, F.; Shah, C.; Suel, T.; Castells, P.; Jones, R.; and Sakai, T., eds., *SIGIR '21: The 44th International ACM SIGIR Conference on Research and Development in Information Retrieval, Virtual Event, Canada, July 11-15, 2021*, 113–122. ACM.

Kang, J.; Li, R.; Liu, Q.; Huang, Z.; Zhang, Z.; Chen, Y.; Zhu, L.; and Su, Y. 2025. Distribution-Driven Dense Retrieval: Modeling Many-to-One Query-Document Relationship. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, 11933–11941.

Karpukhin, V.; Oguz, B.; Min, S.; Lewis, P.; Wu, L.; Edunov, S.; Chen, D.; and Yih, W. 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 2020, Online, November 16-20, 2020*, 6769–6781. Association for Computational Linguistics.

Khattab, O.; and Zaharia, M. 2020. ColBERT: Efficient and Effective Passage Search via Contextualized Late Interaction

over BERT. In Huang, J. X.; Chang, Y.; Cheng, X.; Kamps, J.; Murdock, V.; Wen, J.; and Liu, Y., eds., *Proceedings of the 43rd International ACM SIGIR conference on research and development in Information Retrieval, SIGIR 2020, Virtual Event, China, July 25-30, 2020*, 39–48. ACM.

Kong, W.; Dudek, J. M.; Li, C.; Zhang, M.; and Bendersky, M. 2023. SparseEmbed: Learning Sparse Lexical Representations with Contextual Embeddings for Retrieval. SIGIR '23, 2399–2403. New York, NY, USA: Association for Computing Machinery. ISBN 9781450394086.

Lassance, C.; Déjean, H.; Formal, T.; and Clinchant, S. 2024. SPLADE-v3: New baselines for SPLADE. *CoRR*, abs/2403.06789.

Li, R.; Duan, L.; Xie, G.; Xiao, S.; and Jiang, W. 2022. AdCSE: An Adversarial Method for Contrastive Learning of Sentence Embeddings. In Bhattacharya, A.; Lee, J.; Li, M.; Agrawal, D.; Reddy, P. K.; Mohania, M. K.; Mondal, A.; Goyal, V.; and Kiran, R. U., eds., *Database Systems for Advanced Applications - 27th International Conference, DASFAA 2022, Virtual Event, April 11-14, 2022, Proceedings, Part III*, volume 13247 of *Lecture Notes in Computer Science*, 165–180. Springer.

Lin, J.; Ma, X.; Lin, S.-C.; Yang, J.-H.; Pradeep, R.; and Nogueira, R. 2021. Pyserini: A Python toolkit for reproducible information retrieval research with sparse and dense representations. In *Proceedings of the 44th international ACM SIGIR conference on research and development in information retrieval*, 2356–2362.

MacAvaney, S.; Nardini, F. M.; Perego, R.; Tonellotto, N.; Goharian, N.; and Frieder, O. 2020. Expansion via Prediction of Importance with Contextualization. In Huang, J. X.; Chang, Y.; Cheng, X.; Kamps, J.; Murdock, V.; Wen, J.; and Liu, Y., eds., *Proceedings of the 43rd International ACM SIGIR conference on research and development in Information Retrieval, SIGIR 2020, Virtual Event, China, July 25-30, 2020*, 1573–1576. ACM.

Mallia, A.; Khattab, O.; Suel, T.; and Tonellotto, N. 2021. Learning Passage Impacts for Inverted Indexes. In *Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '21, 1723–1727. New York, NY, USA: Association for Computing Machinery. ISBN 9781450380379.

Ren, R.; Qu, Y.; Liu, J.; Zhao, W. X.; She, Q.; Wu, H.; Wang, H.; and Wen, J.-R. 2021. RocketQAv2: A Joint Training Method for Dense Passage Retrieval and Passage Re-ranking. In Moens, M.-F.; Huang, X.; Specia, L.; and Yih, S. W.-t., eds., *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, 2825–2835. Online and Punta Cana, Dominican Republic: Association for Computational Linguistics.

Robertson, S. E.; and Zaragoza, H. 2009. The Probabilistic Relevance Framework: BM25 and Beyond. *Found. Trends Inf. Retr.*, 3(4): 333–389.

Santhanam, K.; Khattab, O.; Saad-Falcon, J.; Potts, C.; and Zaharia, M. 2022. ColBERTv2: Effective and Efficient Retrieval via Lightweight Late Interaction. In Carpuat, M.; de Marneffe, M.; and Ruíz, I. V. M., eds., *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL 2022, Seattle, WA, United States, July 10-15, 2022*, 3715–3734. Association for Computational Linguistics.

Team, Q. 2024. Qwen2.5: A Party of Foundation Models.

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)*.

Xie, J.; He, X.; Wang, J.; Qiu, Z.; Kebarighotbi, A.; and Ghassemi, F. 2023. SimTDE: Simple Transformer Distillation for Sentence Embeddings. In Chen, H.; Duh, W. E.; Huang, H.; Kato, M. P.; Mothe, J.; and Poblete, B., eds., *Proceedings of the 46th International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR 2023, Taipei, Taiwan, July 23-27, 2023*, 2389–2393. ACM.

Zhang, L.; Wang, B.; Wang, J.; Zhao, X.; Zhang, M.; Yang, H.; Zhang, M.; LI, Y.; Li, J.; Yu, J.; and Zhang, M. 2025. Function-to-Style Guidance of LLMs for Code Translation. In *Forty-second International Conference on Machine Learning*.

Zhang, L.; Zhang, Y.; Long, D.; Xie, P.; Zhang, M.; and Zhang, M. 2024. A Two-Stage Adaptation of Large Language Models for Text Ranking. In Ku, L.-W.; Martins, A.; and Srikumar, V., eds., *Findings of the Association for Computational Linguistics: ACL 2024*, 11880–11891. Bangkok, Thailand: Association for Computational Linguistics.