SuRe-EM: Subspace-Routing and Residual-Corrected Expert Model for Domain-Adaptive Retrieval

Xifan Liu[1], Youyou Huang[1], Zebiao Chen[1], Renchao Zen[1], Siqiang Li[1] and Shouqiang Liu[1(✉)]

[1] School of Artificial Intelligence, South China Normal University, Foshan 528225, China
2024025457@m.scnu.edu.cn

Abstract. As the de facto standard for knowledge-intensive tasks, Retrieval-Augmented Generation (RAG) has significantly enhanced the reliability of Large Language Models by incorporating external non-parametric knowledge. However, adapting general-purpose retrievers to specific vertical domains often triggers catastrophic forgetting, severely degrading performance on open-domain queries. Additionally, existing mitigation strategies, such as linear model fusion, are mathematically constrained within a linear geometric manifold, limiting their ability to effectively rectify complex non-linear semantic drifts caused by domain shifts. To address these limitations, we present an effective approach, SuRe-EM (Subspace-routing & Residual-corrected Expert Model), designed to resolve the issues of domain specialization and cross-domain generalization in domain-adaptive retrieval. SuRe-EM enhances cross-domain representation by integrating fine-grained subspace routing with non-linear residual correction. Specifically, SuRe-EM first employs subspace routing to dynamically decouple high-dimensional features for maximizing domain specialization, followed by a residual module that generates non-linear semantic compensations. We validate our model on a vertical domain dataset (AHD) and general domain datasets (CMRC, SQuAD). Quantitative results demonstrate the effectiveness of SuRe-EM, which maintains strong in-domain precision while improving Recall@10 by up to 4.46 points over state-of-the-art linear fusion baselines in cross-domain scenarios. Furthermore, comprehensive ablation studies validate the non-redundant synergy of the key design elements within SuRe-EM.

Keywords: Retrieval Augmented Generation, Catastrophic Forgetting, Natural Language Processing, Large Language Models.

1 Introduction

Large Language Models (LLMs), such as DeepSeek-R1 and GPT-4 [1, 2], have demonstrated remarkable capabilities in text generation and logical reasoning tasks, attributed to their sophisticated architectures and massive parameters. However, false or invented facts are still occasionally produced by these models. This specific issue is known as hallucination [3]. High accuracy is strictly required in fields such as healthcare and finance. Consequently, these errors can cause severe harm within such knowledge-

based sectors [4]. To solve this problem, researchers now widely use Retrieval-Augmented Generation (RAG) [5], a method that enables large language models to effectively handle knowledge-intensive tasks [6]. By obtaining real external information, RAG technology fills the shortcomings of the model itself. The accuracy of the final text is guaranteed by these introduced data. However, the whole set of operating mechanisms is often subject to the retriever. Once the retrieval module fails to capture the correct background content, the output result of the model will become worse.

The practice of improving the accuracy rate of specific fields by fine-tuning the embedded model has been confirmed by predecessor research institutes. However, when the model adapts to the new target scenario, its cross-field performance tends to deteriorate. The original knowledge will be discarded in large quantities in the process, which is called catastrophic forgetting [7, 8]. Linear fusion, which is commonly adopted at present, can maintain specific skills and general abilities at the same time. The specific practice is to find the weighted average of the fine-tuning model (FT) and the base model (Base) [9]. Complex semantic data is difficult to process by static linear combination. The method assumes that the optimal solution is located on the linear manife of the two models. However, once a complex semantic shift occurs, this premise will be invalid. A fixed configuration may be useful for some data, but it cannot be adapted to the rest of the samples.

In response to the above defects, this study proposes SuRe-EM (Subspace-routing & Residual-corrected Expert Model). As a multi-expert search model, it mainly relies on two core components to operate. Not only subspace routing gate control is used inside the model, but the nonlinear residual correction module is also introduced into it. Domain adaptive decoupling is achieved through the mutual cooperation of these two parts. Through subspace routing, the specialized characteristics of the domain are effectively amplified. The model uses residual correction to deal with semantic offset. These two mechanisms cooperate with each other, so that SuRe-EM can adapt well to vertical scenes without losing its universal robustness. The relevant test then in a vertical domain dataset (AHD) and two general benchmarks (CMRC, SQuAD) unfold on. The results show that SuRe-EM reliably beats traditional linear fusion methods. It keeps high precision on in-domain data. Furthermore, it boosts Recall@10 by 4.46 points on cross-domain tasks.

The main contributions of this article are summarized as follows:

1. We designed a domain adaptive retrieval model called SuRe-EM. The model includes a fine-grained subspace routing module and a nonlinear residual correction module. The two modules jointly optimize the balance between domain specialization and general robustness, providing a solution to the problem of catastrophic forgetting.
2. We have conducted a systematic evaluation of SuRe-EM. A large number of experiments have been carried out on multiple datasets. Experimental results show that the performance of SuRe-EM is better than that of the baseline model.
3. We also did a comprehensive ablation experiment. These experiments confirm the independent contribution of each module in SuRe-EM, and also confirm the synergy between modules.

2 Related Work

The issue of hallucinations in Large Language Models (LLMs) is currently managed best by Retrieval-Augmented Generation (RAG). Factual consistency is significantly boosted because external non-parametric knowledge is added to the system [6, 10]. This approach is applied across many domains, such as medical [11] and tourism QA [12], as well as general text generation [13]. Even so, the final performance is strictly tied to the quality of the retrieval step. According to recent studies, LLM reasoning can be severely damaged by noisy or irrelevant documents. When this occurs, the accuracy of the generated output drops sharply [14].

The dense retrieval model is usually based on the Transformer dual encoder structure (e.g., BERT [15]) and contrastive learning methods (e.g., InfoNCE [16]). This kind of model has become the basis of the retrieval system. However, general models often do not grasp the semantic characteristics of specific areas in vertical fields. Fine-tuning is a commonly used adaptive strategy [17]. Through fine-tuning, the model can align the target distribution. However, this fine-tuning process will destroy the original general knowledge structure of the model and cause catastrophic forgetting. [6, 8].

Model fusion provides a feasible way to solve this forgetting problem. In addition, this method will not increase additional calculation costs in the reasoning process. Methods such as WiSE-FT [18] and LM-Cocktail [9] prove its effectiveness. These methods combine the weights of the pre-trained model (Base) and the fine-tuning model (FT) through linear mixing. In this way, the network not only retains a wide range of general skills, but also achieves good performance in the target domain.

In addition, Task Arithmetic [19] proposes to use vector operations in parameter space to combine model capabilities, such as adding and subtracting specific task vectors. Model Soups [20] average the weights of multiple fine-tuning models to improve the generalization ability. TIES-Merging [21] more stable model integration by selecting sparse parameters and resolving conflicts.

But most of these methods only do global linear operations in parameter space. The combined representation is essentially still a linear or convex combination of the original model. This geometric limitation makes it difficult for them to capture complex cross-domain semantic transformations. If the target semantics is distributed outside the maniform of the original model, these methods usually cannot achieve precise alignment.

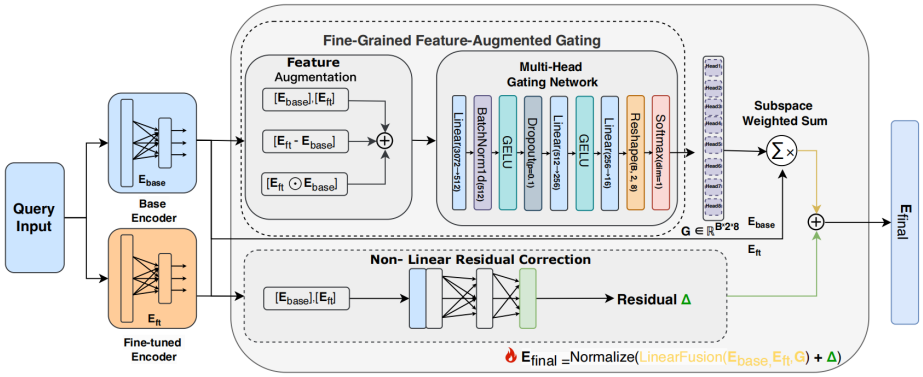

Fig. 1. Schematic diagram of SuRe-EM framework. The model uses a dual-flow architecture. In the upper stream, fine-grained subspace routing is used to dynamically integrate expert knowledge. In the lower flow, nonlinear residual correction is used to correct semantic drift. Through this design, the model achieves robust domain adaptation.

3 Methodology

This section systematically explains the SuRe-EM framework. Let's first introduce a data construction process based on the Large Language Model (LLM), which is used to alleviate the problem of sparse data in the domain. Next, we will discuss the two core components of SuRe-EM: the fine-grained subspace routing mechanism and the nonlinear residual correction module. Finally, explain the optimization and training strategy of the model.

3.1 Data Construction Pipeline

We have adopted an LLM-based process to synthesize the real training corpus [22] to alleviate the problem of data sparseness. (1) Preprocessing: The original corpus is divided into semantic paragraph D. (2) Generation: GPT-4 synthesizes diversified queries according to specific instructions: *Based on the document below, generate K insightful and diverse questions that require understanding the entire text to answer. The questions should be phrased naturally.* (3) Quality Control: Manual audit is carried out according to the standardized process to filter out factual errors and ambiguities and ensure data consistency. (4) Triplet Construction: The effective query is paired with the regular paragraph D, and we apply BGE-M3 [23] to retrieve the difficult sample N for fine-grained discrimination.

3.2 SuRe-EM Framework

We propose the SuRe-EM method. This method is used to overcome the geometric constraints of linear manives in existing linear fusion methods. SuRe-EM obtains accurate semantic representation through feature decoupling and nonlinear correction. As

shown in **Fig. 1**, the architecture contains two core parts: a fine-grained subspace routing module and a nonlinear residual correction module.

Fine-grained Subspace Routing

The traditional Mixed Expert (MoE) model usually assigns a scalar weight to the entire embedded vector. This practice ignores an inherent feature of high-dimensional representation: different dimensions encode different semantic features. To solve this problem, we divide the D-dimensional embedded space into H non-overlapping orthogonal subspaces, as shown in Equation (1). This decoupling mechanism allows the model to reorganize the knowledge from the basic model (Base) and fine-tuning model (FT) at a finer granularity. In the implementation, the number of subspaces $H = 8$ and the dimension of each subspace is $d = D/H$. This value is used as a fixed superparameter in all experiments.

$$\mathrm{V} = \left[\mathrm{v}^{(1)}, \mathrm{v}^{(2)}, \ldots, \mathrm{v}^{(\mathrm{H})}\right], \mathrm{v}^{(\mathrm{h})} \in \mathrm{R}^{\mathrm{d}} \tag{1}$$

For each subspace $\mathbf{v}^{(\mathbf{h})}$, the gate control network independently calculates the allocation weight of experts. In order to enhance the robustness of decision-making, we have added an explicit feature enhancement. This mechanism integrates the differential characteristics and product characteristics of experts, thus strengthening the judgment of semantic differences and correlations by the gated network. The specific calculation process is as follows:

- *Gating Input Feature Construction.* For the h subspace, the augmented input vector is defined as Equation (2).

$$I^{(h)} = \left[E_{\text{base}}^{(h)}; E_{\text{ft}}^{(h)}; \left(E_{\text{base}}^{(h)} - E_{\text{ft}}^{(h)}\right); \left(E_{\text{base}}^{(h)} \odot E_{\text{ft}}^{(h)}\right)\right]. \tag{2}$$

Here, the semicolon ; denotes vector concatenation. The term $\mathrm{E}_{\text{base}}^{(\mathrm{h})} - \mathrm{E}_{\text{ft}}^{(\mathrm{h})}$ captures semantic conflicts, while $\mathrm{E}_{\text{base}}^{(\mathrm{h})} \odot \mathrm{E}_{\text{ft}}^{(\mathrm{h})}$ measures semantic commonality.

- *Routing Weight* Calculation. The expert weights $W^{(h)}$ are computed via a lightweight MLP ($\mathrm{MLP_g}$) followed by softmax function as Equation (3).

$$W^{(h)} = \left[w_{\text{base}}^{(h)}, w_{\text{ft}}^{(h)}\right] = \mathrm{Softmax}\left(\mathrm{MLP_g}\left(I^{(h)}\right)\right). \tag{3}$$

The constraint $w_{\text{base}}^{(h)} + w_{\text{ft}}^{(h)} = 1$ ensures stability within each subspace.

- *Subspace Feature Synthesis.* Finally, the output vector $O^{(h)}$ is synthesized via weighted aggregation as Equation (4).

$$O^{(h)} = w_{\text{base}}^{(h)} E_{\text{base}}^{(h)} + w_{\text{ft}}^{(h)} E_{\text{ft}}^{(h)}. \tag{4}$$

The complete linear fusion representation E_{linear} is obtained by concatenating outputs across all subspaces.

Non-linear Residual Correction Module

Although subspace routing optimizes the allocation of expert weights, the linear merge model essentially limits the final embedding to a convex combination of expert vectors. Geometrically, this restriction makes the representation can only fall on the line segment connecting the two points, which is a compromise solution. In order to break through this geometric limitation, this module introduces a nonlinear residual correction term, written as Δ. This correction term provides additional semantic compensation capability through nonlinear mapping in high-dimensional space. This module adopts a multi-layer sensor (MLP) structure. Its input is the cascade characteristics of the basic model (BASE) and fine-tuning expert (FT). Equation (5) for the calculation method.

$$\Delta = \mathrm{MLP}_{(\mathrm{res})}\left(\left[\mathrm{E}_{(\mathrm{base})}; \mathrm{E}_{(\mathrm{ft})}\right]\right) \tag{5}$$

The MLP_{res} here uses the GeLU activation function [24]. This allows the model to capture nonlinear semantic deviations. In addition, the module adopts a zero initialization strategy. Specifically, the standard deviation of the last layer is set to 1e-5. This can ensure that the model is mainly dominated by robust linear fusion at the beginning of training.

We first add the residual correction term Δ to the multi-head linear synthesis result E_{linear}, and then normalize the layer according to Equation (6), so as to get the final fusion representation E_{final}.

$$E_{\text{final}} = LayerNorm(E_{\text{linear}} + \Delta) \tag{6}$$

In the vector geometric space, Δ is a displacement component deviating from the linear maniform. When the system encounters cross-domain scenarios or highly specialized contexts, Δ will generate the missing "semantic blocks" in the original expert model. In this way, the representation vector is moved from the linear mariform to a more precise coordinate point.

Optimization And Training Strategy

We adopt a two-stage decoupling training strategy. In this way, SuRe-EM can achieve both the generalization ability in the general field and the professional accuracy in the vertical field. In the first stage, the model uses in-domain data and optimizes the loss function through InfoNCE contrastive learning. In this stage, the distance between the positive sample pairs is closer and the distance between the negative sample pairs is pulled apart. Through this process, a discriminative semantic space is constructed, as shown in Equation (7).

$$Loss = -log \frac{e^{\left(\mathrm{sim}\left(q_i, d_i^+\right)/\tau\right)}}{\sum_{j=1}^{B} e^{\left(\mathrm{sim}\left(q_i, d_j^+\right)/\tau\right)} + \sum_{k=1}^{M} e^{sim(q_i, n_{ik})/\tau}} \tag{7}$$

Here, sim( ) denotes the cosine similarity function. d_i^+ represents the positive sample, n_{ik} denotes the k hard negative sample, and τ serves as the temperature hyperparameter.

In the second stage, the parameters of the expert model were frozen by us. Then we use the hybrid dataset to jointly train the routing module and the residual correction module. Let $E_{\text{linear}} = \text{Concat}\left(O^{(1)}, O^{(2)}, \ldots, O^{(H)}\right)$ denote the fused representation obtained by concatenating the routed subspace outputs, and let $E_{\text{final}} = E_{\text{linear}} + \Delta$ denote the final embedding after residual correction. The overall training objective is defined as Equation (8).

$$L_{\text{total}} = InfoNCE(E_{\text{final}}) + \lambda\|\Delta\|_2 \tag{8}$$

E_{linear} is here the dynamic fusion subspace representation generated by the routing module. The residual term Δ is used to model nonlinear cross-domain deviations, which cannot be represented by linear fusion alone. The scalar λ is a learnable parameter whose initial value 0. The l_2 regularization term is used to constrain the size of the residual. This constraint prompts the model to start from linear fusion and gradually learn the necessary corrections. The above joint goals enable the routing mechanism and the residual mechanism to learn together. In this way, the model can achieve cross-domain alignment while maintaining domain-specific capabilities.

4 Experiments

4.1 Datasets

The experimental datasets utilized in this study comprise two publicly available general-domain datasets: CMRC2018 (denoted as CMRC) [25] and ChineseSQuAD (denoted as SQuAD), along with a self-constructed vertical-domain dataset named AHD (AutoHandbook Dataset).

CMRC2018 constitutes a Chinese machine reading comprehension dataset containing triplets of questions, answers, and contextual passages. Through rigorous data curation and augmentation techniques, we constructed a refined corpus of 6,685 QA instances.

ChineseSquad is a dataset specifically constructed for Chinese machine reading comprehension research, based on the original SQuAD [26] dataset. It retains the complete information of questions, answers, and corresponding contexts. Through carefully designed filtering mechanisms and data augmentation methods, we ultimately selected 10,995 high-quality instances from the original dataset.

AHD is a specialized vertical-domain benchmark focused on the intelligent cockpit. The corpus is derived from 500+ pages of original vehicle manuals. After data augmentation, it ultimately comprises 2,145 high-fidelity samples. To ensure annotation quality, we implemented a rigorous protocol where three domain experts conducted a double-blind review of the GPT-4 generated queries based on factual consistency, linguistic naturalness, and semantic complexity. Furthermore, each query is paired with five hard negative samples retrieved via BGE-M3. The high density of long-tail technical terminology and unstructured tabular data in AHD poses a significant challenge for cross-domain retrieval robustness.

Training Strategy. Stage 1 fine-tunes three domain experts on AHD, CMRC, and SQuAD individually. Stage 2 trains SuRe-EM on mixed in-domain and cross-domain data using AHD:CMRC (4:6, 2K), CMRC:AHD (4:6, 2K), and SQuAD:AHD (6:4, 3K). We train SuRe-EM separately for each domain expert, resulting in three pairwise domain-adaptive SuRe-EM models.

4.2 Baseline

We compare our method with the following baseline methods.

- **Base**: The pre-trained BGE-base-zh-v1.5 [27] deployed directly without any further fine-tuning.
- **FT**: We use the Sentence-Transformer framework, where the model was fine-tuned on our augmented training corpus.
- **LM**: Linearly fuses the Base and FT models [9]:
  (1) **LM (fixed)**, with $\lambda = 0.5$, a commonly adopted default in prior work;
  (2) **LM (oracle)**, where $\lambda \in \{0.1, 0.2, \ldots, 0.9\}$ is selected via grid search, and we report the best-performing result for each evaluation setting. Detailed results across different λ values are provided in 5.3, and **Fig. 2** further illustrates the sensitivity of LM performance to the choice of λ. The oracle variant represents an upper bound for linear fusion, as it requires access to target-domain validation signals and scenario-specific tuning, which is typically unavailable in real-world deployment.

4.3 Evaluation Metrics

In the experiments conducted for this study, we employed multiple evaluation metrics to assess retrieval quality, specifically: Recall@10, NDCG@10, and MAP@100. **Recall@10**: This metric measures retrieval depth. It reflects the proportion of correct answers contained within the top-k candidate documents, demonstrating the model's capability for preliminary screening. **NDCG@10**: This metric serves as an indicator of ranking quality. By incorporating position-based logarithmic discounting, it underscores the importance of positioning highly relevant documents at the top of the list. **MAP@100**: This metric evaluates the overall ranking performance within a larger retrieval scope from the perspective of long-tail distribution, serving as a critical indicator of retrieval robustness.

4.4 Implementation Details

We employ BGE-base-zh-v1.5 as the backbone encoder. For the expert fine-tuning stage, we use a learning rate of 2e-5. In the subsequent training of the SuRe-EM components, the learning rate is set to 5e-4. We use a batch size of $B = 512$ and sample $M = 5$ hard negatives for each query. The temperature is set to $\tau = 0.05$, following common practice in contrastive learning. To facilitate fine-grained feature disentanglement, the number of routing subspaces is set to $H = 8$. All experiments are implemented in PyTorch and conducted on NVIDIA A100 80GB GPUs.

5 Results And Analysis

In this section, we systematically evaluate and analyze the performance of our proposed SuRe-EM model, and conduct additional ablation studies to substantiate our experimental findings.

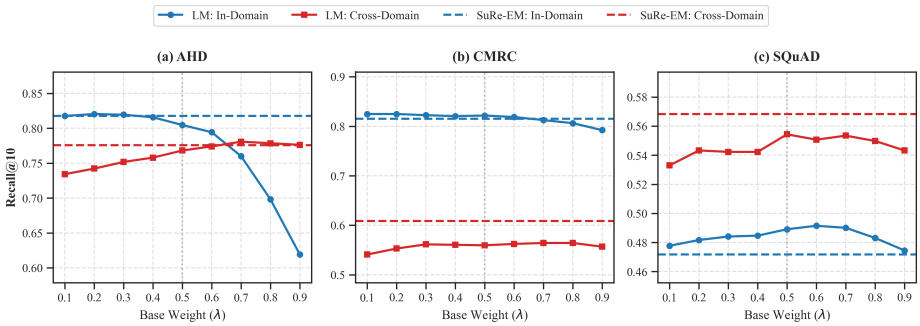

Fig. 2. Recall@10 of LM across different fusion coefficients λ under in-domain and cross-domain evaluations. The optimal λ varies significantly across settings, revealing the limitation of static linear interpolation.

5.1 In-Domain Performance

Table 1 reports the in-domain retrieval performance of all models. On the highly specialized AHD dataset, the Base model performs poorly due to the lack of domain-specific knowledge, achieving a Recall@10 of only 0.5479.

On the AHD dataset, the Recall@10 value obtained by SuRe-EM is 0.8177. This result is slightly lower than the pure fine-tuning (FT) model, but it is still competitive compared with the linear fusion model. This shows that the fine-grained subspace routing mechanism can effectively guide in-domain queries to the corresponding domain experts. Therefore, while maintaining the overall robustness, the model can also obtain high in-domain accuracy.

On the CMRC dataset, the Recall@10 value of SuRe-EM is 0.8152, which is slightly lower than LM(oracle). This small difference can be regarded as a reasonable trade-off. While avoiding coefficient tuning for specific scenarios, SuRe-EM minimizes the loss of in-domain accuracy and significantly improves cross-domain performance.

Overall, these results demonstrate that SuRe-EM maintains competitive in-domain performance while providing a better balance between domain specialization and general robustness.

Table 1. In-Domain retrieval performance comparison.

Test Scenario	Model	R@10	N@10	M@100
CMRC (in-domain)	Base	0.7724	0.8029	0.7459
	FT	0.8215	0.8330	0.7875

	LM(fixed)	0.8216	0.8390	**0.7922**
	LM(oracle)	**0.8247**	0.8360	0.7877
	SuRe-EM	0.8152	0.8300	0.7847
	Base	0.4654	0.4963	0.3934
	FT	0.4740	0.4847	0.3956
SQuAD (in-domain)	LM(fixed)	0.4890	0.5130	0.4178
	LM(oracle)	**0.4914**	**0.5161**	**0.4185**
	SuRe-EM	0.4716	0.4868	0.3935
	Base	0.5479	0.5439	0.4550
	FT	**0.8233**	**0.7598**	**0.7318**
AHD (in-domain)	LM(fixed)	0.8047	0.7500	0.7112
	LM(oracle)	0.8205	0.7564	0.7260
	SuRe-EM	0.8177	0.7530	0.7165

5.2 Cross-Domain Performance

Table 2 summarizes the retrieval performance under cross-domain settings, evaluating robustness against catastrophic forgetting and zero-shot transfer ability.

In the AHD→CMRC scenario, SuRe-EM achieves competitive performance compared to LM with tuned λ, while still outperforming FT by a clear margin. Although LM (oracle) attains slightly higher Recall@10 in this case, the optimal λ is strongly biased toward the base model, effectively reducing the fusion model to a near-base configuration. This behavior benefits this specific transfer direction but does not generalize.

In contrast, in the CMRC→AHD and SQuAD→AHD settings, SuRe-EM consistently outperforms both LM (fixed) and LM (oracle), achieving gains of up to +4.46 and +1.40 in Recall@10, respectively. Similar improvements are observed in NDCG@10 and MAP@100, demonstrating superior transferability to the vertical domain. These results indicate that the non-linear residual correction module effectively compensates for semantic drift introduced during domain-specific fine-tuning.

The results of **Fig. 2** show that the performance of LM is highly sensitive to λ values. Under in-domain settings and cross-domain settings, the optimal value of λ is different. This highlights the limitations of static linear fusion under domain offset, because this fusion needs to be adjusted for specific scenarios. In contrast, SuRe-EM can maintain consistent good performance without manual selection coefficients and show better robustness under various domain offsets.

Table 2. cross-domain and zero-shot transfer performance. $A \rightarrow B$ denotes training on source A and evaluating on target B.

Test Scenario	Model	R@10	N@10	M@100
CMRC → AHD	FT	0.5428	0.5258	0.4411
	LM(fixed)	0.5595	0.5488	0.4632
	LM(oracle)	0.5642	0.5540	0.4665
	SuRe-EM	**0.6088**	**0.5845**	**0.5075**

SQuAD → AHD	FT	0.5330	0.5214	0.4367
	LM(fixed)	0.5544	0.5425	0.4530
	LM(oracle)	0.5544	0.5425	0.4530
	SuRe-EM	**0.5684**	**0.5567**	**0.4733**
AHD → CMRC	FT	0.7294	0.7597	0.7001
	LM(fixed)	0.7682	0.7971	0.7419
	LM(oracle)	**0.7808**	**0.8080**	**0.7516**
	SuRe-EM	0.7758	0.8004	0.7481

5.3 Sensitivity Analysis Of Linear Fusion

To demonstrate the limitations of static linear fusion, we report the Recall@10 of LM under varying coefficients λ in **Table 3** and **Table 4**. The results reveal a critical coefficient drift: the optimal λ varies significantly across tasks, shifting from 0.2 in AHD in-domain evaluation to 0.7 in CMRC → AHD cross-domain transfer. This instability confirms that a single linear manifold cannot effectively rectify complex semantic drifts.

Table 3. cross-domain performance of LM under different fusion coefficients (Base : FT).

Base:FT	AHD → CMRC			CMRC → AHD			SQuAD → AHD		
	R@10	N@100	M@100	R@10	N@100	M@100	R@10	N@100	M@100
0.1:0.9	0.7343	0.7648	0.7066	0.5409	0.5316	0.4471	0.5330	0.5213	0.4364
0.2:0.8	0.7423	0.7728	0.7152	0.5530	0.5395	0.4514	0.5433	0.5289	0.4413
0.3:0.7	0.7518	0.7814	0.7242	0.5614	0.5492	0.4604	0.5423	0.5311	0.4447
0.4:0.6	0.7580	0.7886	0.7339	0.5605	0.5503	0.4635	0.5423	0.5348	0.4497
0.5:0.5	0.7682	0.7971	0.7419	0.5595	0.5488	0.4632	0.5544	0.5425	0.4530
0.6:0.4	0.7741	0.8032	0.7485	0.5623	0.5501	0.4635	0.5507	0.5418	0.4525
0.7:0.3	0.7808	0.8080	0.7516	0.5642	0.5540	0.4665	0.5535	0.5426	0.4520
0.8:0.2	0.7786	0.8080	0.7537	0.5642	0.5540	0.4645	0.5498	0.5443	0.4548
0.9:0.1	0.7763	0.8060	0.7500	0.5567	0.5502	0.4591	0.5433	0.5396	0.4513

Table 4. In-Domain performance of LM under different fusion coefficients (Base : FT).

Base:FT	AHD			CMRC			SQuAD		
	R@10	N@100	M@100	R@10	N@100	M@100	R@10	N@100	M@100
0.1:0.9	0.8177	0.7560	0.7270	0.8247	0.8357	0.7873	0.4776	0.4917	0.4010
0.2:0.8	0.8205	0.7564	0.7260	0.8247	0.8360	0.7877	0.4816	0.4988	0.4071
0.3:0.7	0.8195	0.7567	0.7250	0.8223	0.8369	0.7902	0.4840	0.5042	0.4121
0.4:0.6	0.8158	0.7563	0.7218	0.8205	0.8366	0.7908	0.4846	0.5083	0.4157
0.5:0.5	0.8047	0.7500	0.7112	0.8216	0.8390	0.7922	0.4890	0.5130	0.4178
0.6:0.4	0.7944	0.7421	0.6935	0.8186	0.8381	0.7912	0.4914	0.5161	0.4185
0.7:0.3	0.7600	0.7192	0.6575	0.8124	0.8344	0.7866	0.4900	0.5154	0.4164
0.8:0.2	0.6981	0.6707	0.5940	0.8062	0.8292	0.7781	0.4830	0.5114	0.4113
0.9:0.1	0.6191	0.6028	0.5168	0.7921	0.8189	0.7651	0.4744	0.5042	0.4030

5.4 Ablation

We did further experiments to explore the contribution of each component in the model. These experiments include systematic ablation studies on nonlinear residual correction modules and fine-grained subspace routing modules. Experimental results show that each module plays a different role in different scenarios. At the same time, there is also an obvious synergy between these modules.

Ablation Analysis Of The Non-linear Residual Correction Module

The Non-linear Residual Correction module is designed to correct cross-domain semantic misalignment. As shown in **Table 5**, removing this module results in only marginal changes in in-domain performance, while causing a substantial degradation in cross-domain scenarios. For example, Recall@10 drops by 4.18 points in the CMRC→AHD setting. This observation indicates that linear fusion alone is insufficient to handle domain shift, and highlights the importance of residual correction for cross-domain alignment.

Table 5. Impact of removing the Non-linear Residual Correction module. $A \rightarrow B$ denotes training on source A and evaluating on target B. Δ values are reported in absolute points (×100).

Test Scenario	R@10	ΔR(points)	N@10	ΔN(points)
AHD (in-domain)	0.8093	-0.84	0.7475	-0.55
CMRC (in-domain)	0.8207	+0.55	0.8340	+0.40
SQuAD (in-domain)	0.4728	+0.12	0.4855	-0.13
AHD → CMRC	0.7674	-0.84	0.7941	-0.63
CMRC → AHD	0.5670	-4.18	0.5546	-2.99
SQuAD → AHD	0.5367	-3.17	0.5223	-3.44

Ablation Analysis Of The Fine-grained Subspace Routing Module

To evaluate the necessity of fine-grained routing, we reduce the number of subspaces from $H = 8$ to $H = 1$, effectively degenerating the model into a global scalar gating mechanism. As reported in **Table 6**, this degeneration leads to consistent performance drops across cross-domain tasks, with Recall@10 decreasing by up to 7.53 points in CMRC→AHD. These results demonstrate that fine-grained feature partitioning is crucial for capturing heterogeneous semantic shifts across domains. Empirically, we suggest $H = 8$ as a heuristic balance between representation capacity and computational efficiency, as higher values offer diminishing returns.

Table 6. Analysis of Subspace Routing granularity ($H = 8$ vs $H = 1$). $A \rightarrow B$ denotes training on source A and evaluating on target B. Δ values are reported in absolute points (×100).

Test Scenario	R@10	ΔR(points)	N@10	ΔN(points)
AHD (in-domain)	0.8177	+0.00	0.7543	+0.13
CMRC (in-domain)	0.8182	+0.30	0.8301	-0.01

SQuAD (in-domain)	0.4694	-0.22	0.4819	-0.49
CMRC → AHD	0.5335	-7.53	0.5345	-5.00
SQuAD → AHD	0.5205	-4.79	0.5148	-4.19
AHD → CMRC	0.7266	-4.92	0.7572	-4.32

Ablation Analysis Of Dual-Module Synergy

We further studied the combined effect of subspace routing and residual correction. The practice is to remove these two modules at the same time. As shown in **Table 7**, t this practice leads to a significant decline in performance in most cases. In the scene in the AHD domain, the recall rate (Recall@10) decreased by 14.61 percentage points.

A slight improvement (+1.17) is observed in the AHD→CMRC setting, which may be attributed to reduced model capacity and weaker source-domain bias. However, this comes at the cost of degraded performance in other settings, indicating that the full model provides more robust overall performance.

Table 7. Assessment of component synergy. $A \rightarrow B$ denotes training on source A and evaluating on target B. Δ values are reported in absolute points (×100).

Test Scenario	R@10	ΔR(points)	N@10	ΔN(points)
AHD (in-domain)	0.6716	-14.61	0.6483	-10.47
CMRC (in-domain)	0.8124	-0.28	0.8266	-0.34
SQuAD (in-domain)	0.4664	-0.52	0.4804	-0.64
CMRC → AHD	0.5870	-2.18	0.5661	-1.84
SQuAD → AHD	0.5316	-3.68	0.5077	-4.90
AHD → CMRC	0.7875	+1.17	0.8119	+1.15

5.5 Efficiency Analysis

As shown in **Table 8**, SuRe-EM introduces only 2.60M trainable parameters (+2.54%) while the Base and FT encoders remain frozen. The fusion modules are highly efficient, with the additional computation over subspace representations resulting in a negligible latency overhead of only +0.03ms beyond the dual-encoder baseline. This ensures that SuRe-EM achieves superior domain adaptation with minimal deployment cost compared to existing merging strategies.

Table 8. Computational Complexity Comparison (Batch Size = 1).

Model	Parameters (M)	Latency (ms)
Base	102.27	8.44
FT	102.27	8.45
SuRe-EM (Ours)	104.87 (+2.54%)	16.92 (+0.03)

6 Conclusion

This paper presents SuRe-EM, an efficient retrieval model that addresses the trade-off between domain specialization and general robustness via a non-linear residual correction module and fine-grained subspace routing. Experimental results demonstrate that SuRe-EM consistently improves cross-domain performance while maintaining in-domain precision, thereby effectively alleviating catastrophic forgetting. Future work will extend this framework to heterogeneous multi-expert and multi-modal retrieval scenarios.

7 Acknowledgments

This work was supported in part by Guangdong Provincial Undergraduate University Teaching Quality and Teaching Reform Engineering Project (Yue Jiao Gao Han [2026] No. 4, Project No. 97),South China Normal University Teaching Quality and Teaching Reform Engineering Project (Jiao Xue [2025] No. 44, Project No. 11; Jiao Xue [2023] No. 39, Project No. 191),Guangdong Province Joint Postgraduate Education and Training Demonstration Base (866[2024] No. 1,Project No.032,),the ”Challenge Based Leadership” Action Plan Selected Projects (School of Computer[2025] No. 6,Project No. 19, 20, 21).

References

1. Guo, D., Yang, D., Zhang, H., Song, J., Zhang, R., Xu, R., Zhu, Q., Ma, S., Wang, P., Bi, X., others: Deepseek-r1: Incentivizing reasoning capability in llms via reinforcement learning. arXiv preprint arXiv:2501.12948. (2025).
2. Achiam, J., Adler, S., Agarwal, S., Ahmad, L., Akkaya, I., Ale-man, F.L., Almeida, D., Altenschmidt, J., Altman, S., Anadkat, S., others: Gpt-4 technical report. arXiv preprint arXiv:2303.08774. (2023).
3. Huang, L., Yu, W., Ma, W., Zhong, W., Feng, Z., Wang, H., Chen, Q., Peng, W., Feng, X., Qin, B., others: A survey on hallucination in large language models: Principles, taxonomy, challenges, and open questions. ACM Transactions on Information Systems. 43, 1–55 (2025).
4. Chen, Z.Z., Ma, J., Zhang, X., Hao, N., Yan, A., Nourbakhsh, A., Yang, X., McAuley, J., Petzold, L., Wang, W.Y.: A survey on large language models for critical societal domains: Finance, healthcare, and law. arXiv preprint arXiv:2405.01769. (2024).
5. Lewis, P., Perez, E., Piktus, A., Petroni, F., Karpukhin, V., Goyal, N., Küttler, H., Lewis, M., Yih, W., Rocktäschel, T., others: Retrieval-augmented generation for knowledge-intensive nlp tasks. Advances in neural information processing systems. 33, 9459–9474 (2020).
6. Guu, K., Lee, K., Tung, Z., Pasupat, P., Chang, M.: Retrieval augmented language model pre-training. In: International conference on machine learning. pp. 3929–3938. PMLR (2020).
7. Goodfellow, I.J., Mirza, M., Xiao, D., Courville, A., Bengio, Y.: An empirical investigation of catastrophic forgetting in gradient-based neural networks. arXiv preprint arXiv:1312.6211. (2013).

8. Chen, S., Hou, Y., Cui, Y., Che, W., Liu, T., Yu, X.: Recall and learn: Fine-tuning deep pretrained language models with less forgetting. In: Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP). pp. 7870–7881 (2020).
9. Xiao, S., Liu, Z., Zhang, P., Xing, X.: Lm-cocktail: Resilient tuning of language models via model merging. In: Findings of the Association for Computational Linguistics: ACL 2024. pp. 2474–2488 (2024).
10. Zhu, K., Luo, Y., Xu, D., Yan, Y., Liu, Z., Yu, S., Wang, R., Wang, S., Li, Y., Zhang, N., others: Rageval: Scenario specific rag evaluation dataset generation framework. In: Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers). pp. 8520–8544 (2025).
11. Pang, T., Tan, K., Yao, Y., Liu, X., Meng, F., Fan, C., Zhang, X.: Remed: Retrieval-augmented medical document query responding with embedding fine-tuning. In: 2024 International Joint Conference on Neural Networks (IJCNN). pp. 1–8. IEEE (2024).
12. Gupta, A., Rawat, M., Stolcke, A., Pieraccini, R.: REFINE on scarce data: Retrieval enhancement through fine-tuning via model fusion of embedding models. In: Australasian Joint Conference on Artificial Intelligence. pp. 73–85. Springer (2024).
13. Yu, W., Zhu, C., Li, Z., Hu, Z., Wang, Q., Ji, H., Jiang, M.: A survey of knowledge-enhanced text generation. ACM Computing Surveys. 54, 1–38 (2022).
14. Yoran, O., Wolfson, T., Ram, O., Berant, J.: Making Retrieval-Augmented Language Models Robust to Irrelevant Context. In: ICLR 2024 Workshop on Large Language Model (LLM) Agents.
15. Devlin, J., Chang, M.-W., Lee, K., Toutanova, K.: Bert: Pre-training of deep bidirectional transformers for language understanding. In: Proceedings of the 2019 conference of the North American chapter of the association for computational linguistics: human language technologies, volume 1 (long and short papers). pp. 4171–4186 (2019).
16. Oord, A. van den, Li, Y., Vinyals, O.: Representation learning with contrastive predictive coding. arXiv preprint arXiv:1807.03748. (2018).
17. Karpukhin, V., Oguz, B., Min, S., Lewis, P.S., Wu, L., Edunov, S., Chen, D., Yih, W.: Dense Passage Retrieval for Open-Domain Question Answering. In: EMNLP (1). pp. 6769–6781 (2020).
18. Wortsman, M., Ilharco, G., Kim, J.W., Li, M., Kornblith, S., Roelofs, R., Lopes, R.G., Hajishirzi, H., Farhadi, A., Namkoong, H., others: Robust fine-tuning of zero-shot models. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 7959–7971 (2022).
19. Ortiz-Jimenez, G., Favero, A., Frossard, P.: Task arithmetic in the tangent space: Improved editing of pre-trained models. Advances in Neural Information Processing Systems. 36, 66727–66754 (2023).
20. Wortsman, M., Ilharco, G., Gadre, S.Y., Roelofs, R., Gontijo-Lopes, R., Morcos, A.S., Namkoong, H., Farhadi, A., Carmon, Y., Kornblith, S., others: Model soups: averaging weights of multiple fine-tuned models improves accuracy without increasing inference time. In: International conference on machine learning. pp. 23965–23998. PMLR (2022).
21. Yadav, P., Tam, D., Choshen, L., Raffel, C.A., Bansal, M.: Ties-merging: Resolving interference when merging models. Advances in neural information processing systems. 36, 7093–7115 (2023).
22. Gilardi, F., Alizadeh, M., Kubli, M.: ChatGPT outperforms crowd workers for text-annotation tasks. Proceedings of the National Academy of Sciences. 120, e2305016120 (2023).

23. Chen, J., Xiao, S., Zhang, P., Luo, K., Lian, D., Liu, Z.: M3-Embedding: Multi-Linguality, Multi-Functionality, Multi-Granularity Text Embeddings Through Self-Knowledge Distillation. In: Ku, L.-W., Martins, A., and Srikumar, V. (eds.) Findings of the Association for Computational Linguistics: ACL 2024. pp. 2318–2335.
24. Hendrycks, D., Gimpel, K.: Gaussian error linear units (gelus). arXiv preprint arXiv:1606.08415. (2016).
25. Cui, Y., Liu, T., Che, W., Xiao, L., Chen, Z., Ma, W., Wang, S., Hu, G.: A span-extraction dataset for Chinese machine reading comprehension. In: Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP). pp. 5883–5889 (2019).
26. Rajpurkar, P., Zhang, J., Lopyrev, K., Liang, P.: Squad: 100,000+ questions for machine comprehension of text. arXiv preprint arXiv:1606.05250. (2016).
27. Xiao, S., Liu, Z., Zhang, P., Muennighoff, N., Lian, D., Nie, J.-Y.: C-pack: Packed resources for general chinese embeddings. In: Proceedings of the 47th international ACM SIGIR conference on research and development in information retrieval. pp. 641–649 (2024).