SEAL: Structure and Element Aware Learning to Improve Long Structured Document Retrieval

Xinhao Huang[1*], **Zhibo Ren**[3*], **Yipeng Yu**[3], **Ying Zhou**[4], **Zulong Chen**[3†], **Zeyi Wen**[1,2†]
[1]HKUST (Guangzhou), Guangzhou, China [2]HKUST, Hong Kong, China
[3]Alibaba Group, Hangzhou, China [4]Zhejiang Lab, Hangzhou, China
chenzulong198867@gmail.com, wenzeyi@ust.hk

Abstract

In long structured document retrieval, existing methods typically fine-tune pre-trained language models (PLMs) using contrastive learning on datasets lacking explicit structural information. This practice suffers from two critical issues: 1) current methods fail to leverage structural features and element-level semantics effectively, and 2) the lack of datasets containing structural metadata. To bridge these gaps, we propose SEAL, a novel contrastive learning framework. It leverages structure-aware learning to preserve semantic hierarchies and masked element alignment for fine-grained semantic discrimination. Furthermore, we release StructDocRetrieval, a long structured document retrieval dataset with rich structural annotations. Extensive experiments on both released and industrial datasets across various modern PLMs, along with online A/B testing, demonstrate consistent performance improvements, boosting NDCG@10 from 79.41% to 82.59% on BGE-M3. The resources are available at this URL.

1 Introduction

Document retrieval is a fundamental component of knowledge-intensive systems, such as Retrieval-Augmented Generation (RAG) (Zhao et al., 2024; Gupta et al., 2024). Despite recent advances in PLMs that extend sequence processing capacity (e.g., from 512 to 8192 tokens), the precise identification of query-relevant content in long documents remains an open challenge (Devlin et al., 2019; Chen et al., 2024). Existing methods typically employ contrastive learning trained on query and raw textual content (Xiong et al., 2021b,a; Li et al., 2021b, 2023a; Wang et al., 2021; Li et al., 2022; Rao et al., 2022, 2023, 2025) or to optimize PLM representations. Nevertheless, as illustrated in Figure 1, this paradigm exhibits two key limitations: (1) Structural blindness arising from raw text processing that disrupts document hierarchies and discards semantic markup indicators like H1/H2 headings (Tan et al., 2025); and (2) Insufficient element-level alignment capacity over fragmented text segments, which fails to preserve fine-grained semantic relationships. Furthermore, the lack of structural metadata in current datasets leaves long structured document retrieval scenarios under-studied.

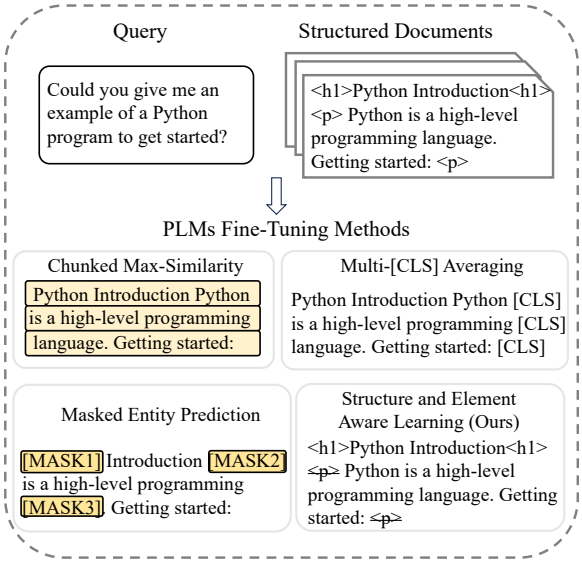


Figure 1: An overview of long structured documents retrieval methods.

To address these limitations, we propose SEAL, a novel contrastive learning framework that integrates structural semantics through two key components: (1) A structure-aware contrastive learning method leveraging HTML transformation and structural tag inclusion/exclusion enables semantic hierarchy induction; and (2) An element-level alignment mechanism employing stochastic element masking forces the model to achieve granular semantic alignment. Our SEAL assigns significantly higher relevance to important elements (e.g.,

[*] Equal Contribution
[†] Corresponding Author

Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing, pages 8538–8548
November 4-9, 2025

chapter titles or intentionally bolded query terms) compared to mentions in body text. This contrasts with structure-agnostic retrieval, which erroneously assigns them equal weights.

Beyond the methodological challenges, the availability of suitable benchmark datasets is crucial. Current retrieval datasets either focus on task-specific retrieval (e.g., passage, product, code, ranking) (Karpukhin et al., 2020a; Reddy et al., 2022; Husain et al., 2019) or lack structured metadata (Nguyen et al., 2016). Their short text lengths (typically <1,000 words) further render their usage. To bridge this gap and provide a reproducible resource for the community, we release StructDocRetrieval, a dataset designed for long structured document retrieval. StructDocRetrieval contains annotated documents with explicit structural semantics and an average length of more than 10,000 words.

We conduct extensive experiments to evaluate SEAL against the state-of-the-art document retrieval methods on StructDocRetrieval and the industrial dataset, using different modern PLMs. We also validate its practical effectiveness through online A/B tests. The experimental results reveal that SEAL achieves remarkable retrieval performance in widely used evaluation metrics. For instance, when implemented with the BGE-M3 model, SEAL elevates NDCG@10 from 73.96% to 77.84%, outperforming existing methods. A series of ablation studies coupled with pattern visualization further confirms that SEAL can effectively capture and utilize document structural semantics.

Our contributions can be summarized as follows.

- We propose SEAL, a novel contrastive learning framework explicitly incorporating document structural semantics through structure-aware learning and fine-grained element-level alignment, thereby enhancing structured data representations in a unified embedding space.
- We release StructDocRetrieval, a dataset specifically designed for long structured document retrieval, which has over 10,000 words in document length on average and contains explicit structural information.
- Extensive experiments across multiple modern PLMs demonstrate SEAL's consistent superiority over the state-of-the-art methods on both StructDocRetrieval and industrial datasets. Online A/B testing further validates the effectiveness of SEAL.

2 Related Work

In this section, we review related work, including Pre-trained Language Models (PLMs), long document retrieval methods, and related benchmarks.

2.1 Pre-trained Language Models

The field of document retrieval has undergone transformative advancements driven by PLMs, particularly through the enhanced capability of extended context windows to effectively encode long documents. Seminal work by Karpukhin et al. (2020b) introduced dense passage retrieval for open-domain question answering, demonstrating the superior capabilities of PLMs in retrieval tasks. Subsequent research validated the effectiveness of BERT-based architectures (Warner et al., 2024), particularly through PLM integration in multi-stage document ranking (Gao and Callan, 2021). Further innovations, such as ColBERT (Khattab and Zaharia, 2020), advanced the field through contextualized late interaction mechanisms over BERT, enabling efficient passage retrieval. More recently, M3-Embedding (Chen et al., 2024) has emerged as a state-of-the-art approach, leveraging self-knowledge distillation to optimize embedding quality and establish itself as a foundational architecture in retrieval systems.

In contrast to the aforementioned encoder-only embedding models, decoder-only embedding models based on large language models (LLMs), such as gte-Qwen2-Instruct (Li et al., 2023b), MiniCPM-Embedding (Hu et al., 2024), and NV-Embed (Lee et al., 2025), introduce significantly higher latency (several times) during vector representation generation, substantially prolonging retrieval time. However, the resulting performance improvement is not commensurate with this considerable overhead. Consequently, this work primarily employs encoder-only pre-trained models.

2.2 Long Document Retrieval

While contemporary document retrieval methods achieve remarkable performance in unstructured textual domains, their architectural limitations become apparent when handling structured data (e.g., technical specifications, legal instruments, and scholarly articles). Works in query optimization include ANCE (Xiong et al., 2021a) and DANCE (Li et al., 2021b), which pioneer adaptive query expansion via contrastive dual learning, and Dai et al. (2024)'s entailment tuning for dense passage

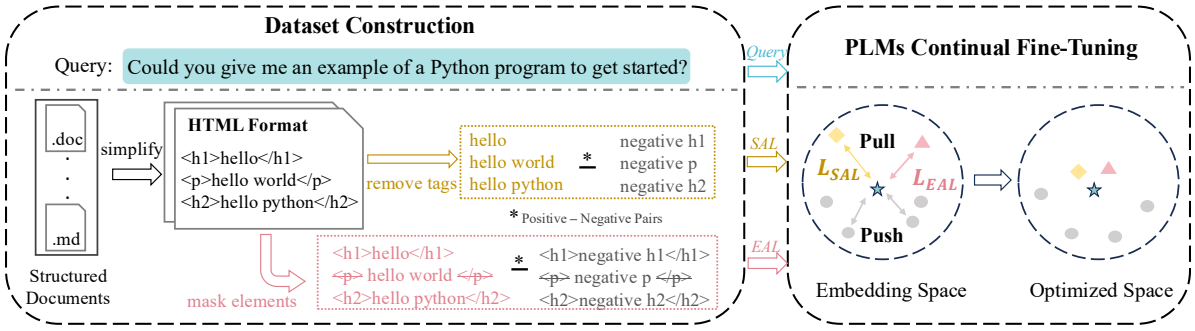


Figure 2: Framework of SEAL. We first construct the used dataset, including HTML transformation, tag processing, and element masking. To guide PLMs to map both queries and structured documents in a unified embedding space, we introduce Structure-Aware Learning (SAL) to incorporate document structural information and Element-Aware Alignment (EAL) to enhance semantic representation.

retrieval such as open-domain question answering. Industrial-grade implementations like Facebook's EBR (Huang et al., 2020) achieve scalability through hybrid embedding topologies, whereas MGDSPR (Li et al., 2021a) retrieves the most relevant products from a large corpus while retaining personalized user features in e-commerce retrieval. Longtriever (Yang et al., 2023) divides long documents into short chunks and then models local semantics within the chunks and global context semantics between the chunks to improve retrieval. Sun et al. (2025) propose a hybrid retriever to obtain keyword and contextual information to further improve the quality of pseudo-documents. SANTA (Li et al., 2023a) and CONAN (Li et al., 2025) employ structure-aware pre-training protocols that combine structured data alignment and masked entity prediction for code retrieval and product retrieval. However, these advances still exhibit a gap in addressing the retrieval requirements of long structured documents: insufficient capture of hierarchical structure and inability to perform fine-grained semantic alignment.

2.3 Related Benchmarks

Foundational benchmarks like MS MARCO (Nguyen et al., 2016), TREC (Craswell et al., 2020), and the multi-domain BEIR benchmark (Thakur et al., 2021) have driven progress in supervised and zero-shot retrieval paradigms. Some datasets focus on specific tasks, such as product and code search (Reddy et al., 2022; Husain et al., 2019). LongBench (Bai et al., 2024a), LongBench-V2 (Bai et al., 2024b), and RULER (Hsieh et al., 2024) are designed to assess LLM long-context reasoning, but are limited by the absence of document-grounded user queries and structural labels. Additionally, LongBench-V2 uses the choice format. Due to the above limitations, these works are insufficient for evaluating modern retrieval models in contemporary information retrieval systems.

3 Methodology

In this section, we first recall the preliminaries of long document retrieval. Subsequently, as shown in Figure 2, we introduce the framework of this work, including dataset construction, such as document pre-processing with tagging removal and element masking of structured documents, and continuous fine-tuning of PLMs with SEAL, which incorporates inherent structural features and fine-grained element alignment.

Preliminary of Document Retrieval Given a natural language query q and a structured document corpus $D = \{d_i\}_{i=1}^{n}$, the retriever identifies the top-k most relevant documents through a ranked list $\{d_1, d_2, ..., d_k\}$ of the k most relevant documents, ranked by relevance scores. Usually, we encode queries and structured documents with PLMs and map them into an embedding space for the calculation of the relevance score.

Let $\phi(\cdot) \in \mathbb{R}^l$ and $\varphi(\cdot) \in \mathbb{R}^l$ denote embedding functions that map queries and documents into an l-dimensional latent space, respectively. The relevance score of a query-document pair (q, d) can be formally expressed as Equation 1 below.

$$f(q, d) = sim(\phi(q), \varphi(d)) \quad (1)$$

where $sim(\cdot)$ is a measurement function such as the inner product.

Query: How to use Python in VS Code?	
Relevant Document	<title> [Nanny-level tutorial] VS Code installation and configuration of Python </title> Configure Jupyter in VS Code Install Jupyter extension Choose the version that suits your computer and start downloading. · · ·
Irrelevant Document	<title> Data, algorithms, computing power, and blockchain + AI </title> Blockchain technology lays the foundation for a decentralized Internet. Many investment firms have turned their attention to the emerging field of machine learning and artificial intelligence. · · ·

Table 1: A data example of StructDocRetrieval.

Split	Query	Doc.	Avg. Words Query	Avg. Words Doc.
Industrial Dataset				
Train	12,047	8,580	10.07	7,310
Evaluation	1,396	1,286	10.03	6,878
StructDocRetrieval				
Train	23816	23816	12.82	10,849
Test	3404	3404	13.04	10,535
Evaluation	6804	6804	12.74	11,047

Table 2: Data statistics of experiment datasets. The "Avg. Words" means the average number of words in queries and documents.

3.1 Dataset Construction

Given HTML's native hierarchical representation capabilities surpassing plain text in modeling retrieved knowledge (Tan et al., 2025), our method first converts structured documents into standardized HTML representations, where each constituent element preserves positional integrity via encapsulation within semantic markup tags. Subsequent pre-processing involves dual operations: (1) Tag Processing creates variants through retaining and removing tags, which facilitates the subsequent learning of basic structural features; (2) Element masking generates markup-depleted variants through stochastic tag elimination, enabling fine-grained alignment.

For industry data derived from real-world applications, we collect user-submitted queries through production system logs and acquire corresponding retrieved document lists via instrumentation in the engineering pipeline, with all documents stored in HTML format. For web-crawled documents, our pipeline initiates with targeted article acquisition, harvesting linked documents through breadth-first search crawling. Following data collection, we implement a preprocessing phase comprising removal of non-structural tags (e.g., line breaks and irrelevant markup tags) through regular expression pattern matching. The sanitized outputs subsequently undergo LLM-powered query synthesis to generate corresponding user queries. We designate this resource as StructDocRetrieval (MIT License). The details of the data example are shown in Table 1.

Table 2 presents statistics of the industrial dataset and StructDocRetrieval. Unlike typical short datasets, documents in our datasets are significantly longer, with an average of more than 7,000 words. In contrast, datasets like MS MARCO (Nguyen et al., 2016) have a maximum of 1,670 words, with most documents under 700 words. Furthermore, documents in MS MARCO and other variants are typically plain text, whereas StructDocRetrieval utilizes HTML format.

3.2 Structure-Aware Learning

PLMs have demonstrated remarkable capabilities in text representation learning via objectives like masked language modeling on large text corpora. However, their inherent lack of mechanisms to capture structural information hinders their ability to effectively comprehend and represent structured documents. This limitation consequently impacts the efficacy of structured document retrieval. To address this, we propose Structure-Aware Learning (SAL) to enhance PLMs with the capacity to encode structural information.

SAL aims to enable the model with structural awareness through a contrastive learning objective that leverages structural variants of relevant documents. We utilize preprocessed HTML-structured relevant documents as positive instances and irrelevant documents as negatives. To guide the model in recognizing the underlying structure, even without explicit tags, we derive plain text versions of these documents by removing all structural tags. The core idea is to train the model to distinguish query-aligned text originating from structured relevant documents from text originating from irrelevant documents.

As defined in Equation 2, the contrastive loss $\mathcal{L}_{SAL}$ maximizes the similarity between the query embedding q and the embedding of the relevant document d^+, while minimizing the similarity with irrelevant documents d^-.

$$\mathcal{L}_{SAL} = -log\frac{e^{f(q,d^+)}}{e^{f(q,d^+)} + \sum_{d^- \in D^-} e^{f(q,d^-)}} \quad (2)$$

The contrastive formulation incorporates dual variants: intact documents preserving markup semantics d^+_{tag} and its destructured counterpart d^+_{untag} with removed tags. The negative samples $D^- = \{d^-\}$ adopt the same process. Both tagged and untagged versions of d^+ and d^- are used in the same loss computation. This design forces the model to recognize that structural semantics and the textual content of relevant documents should align with the query in the same embedding space.

3.3 Element-Aware Alignment

Masking strategies, such as masked language modeling (Li et al., 2020) and masked entity prediction (Sciavolino et al., 2021; Li et al., 2023a), have proven effective in learning robust text representations. Unlike these approaches focusing on token or entity recovery, we introduce Element-Aware Alignment (EAL) based on masking structural elements within documents to foster fine-grained structural understanding.

For structured documents, we define elements as text spans annotated with tags (e.g., headings, list items, paragraphs). To encourage the model to learn fine-grained representations of these elements and their contextual roles, we randomly mask the structural tags of a proportion (e.g., 10%) of elements in a document. Formally, let a structured document be represented as a sequence of elements $d = \{(t_1, tag_1), (t_2, tag_2), \cdots, (t_n, tag_n)\}$, where t_i is the text content and tag_i is the structural tag.

A masked document d_{mask} is constructed by removing tag_i for a random subset of indices, while keeping other elements intact:

$$d_{mask} = \{\epsilon_1^{mask}, \epsilon_2, \epsilon_3^{mask}, \cdots, \epsilon_n\} \quad (3)$$

where ϵ_i^{mask} denotes the i-th element with its structural tag removed, while $\epsilon_i = (t_i, tag_i)$ preserves both the original text and its tag.

We form positive pairs using a query q and its corresponding relevant document subjected to element masking d^+_{mask}. Negative pairs consist of q and masked irrelevant documents d^-_{mask}. The training objective is based on a contrastive loss defined as follows:

$$\mathcal{L}_{EAL} = -log\frac{e^{f(q,d^+_{mask})}}{e^{f(q,d^+_{mask})} + \sum_{d^-\in D^-} e^{f(q,d^-_{mask})}} \quad (4)$$

Minimizing $\mathcal{L}_{EAL}$ trains the model to maintain high similarity between the query and the representation of the masked relevant document, while pushing away masked irrelevant documents. This objective forces the model to leverage the unmasked elements and the textual content within masked elements to infer the document's relevance, thereby enhancing its ability to utilize fine-grained element-level information.

4 Experiments

In this section, we evaluate SEAL across various datasets using different PLMs. We further present in-depth studies of SEAL, including ablation studies and embedding distributions visualization. Additionally, we show the practical improvement of SEAL in the industrial environment.

4.1 Setup

We describe the basic experiment setup used in our work in this section.

Datasets and Models This study utilizes industry data from real-world applications. User-clicked documents serve as positive examples (target documents), while non-clicked documents are treated as negative examples. All documents are stored in HTML format. We select a set of modern embedding models as baselines, including Multilingual-E5-large (Wang et al., 2024), bge-large-zh (Xiao et al., 2023), BGE-M3 (Chen et al., 2024), and GTE-Qwen2-1.5B (Li et al., 2023b).

Evaluation Metrics We adopt three widely-used metrics: HitRate, MRR, and NDCG. HitRate reflects immediate retrieval accuracy, MRR emphasizes the ability of the model to prioritize critical items, and NDCG considers graded relevance and positional sensitivity.

Baselines We compare SEAL with state-of-the-art document retrieval methods: Chunk, MCLS (Chen et al., 2024), and SANTA(Li et al., 2023a).

Chunk-based processing is a conventional solution in document retrieval. Long documents are segmented into fixed-length chunks of 512 tokens, each independently encoded via PLMs. Query-document relevance is determined by computing dot product similarities between the query embedding and each chunk's embedding, with the maximum value retained as the final score.

For MCLS, we insert a "[CLS]" token for every fixed number of tokens (inserting a [CLS] token for each 256 tokens in our experiments). The final document embedding is computed by averaging the last hidden states of all [CLS] tokens.

Method	HitRate@1	HitRate@3	HitRate@5	MRR@5	MRR@10	NDCG@5	NDCG@10
mE5-large	54.11	79.62	85.86	67.39	68.06	72.18	74.11
+ *Chunk*	56.85	82.94	88.79	70.12	71.45	74.78	77.42
+ *MCLS*	57.74	84.12	89.56	71.08	72.41	75.76	78.44
+ *SANTA*	55.79	81.76	88.02	69.01	70.49	73.79	76.50
+ SEAL	**58.63**	**85.29**	**90.34**	**72.02**	**73.37**	**76.74**	**79.35**
bge-large-zh	59.08	83.48	89.47	71.34	72.21	75.84	76.84
+ *Chunk*	61.97	86.11	91.67	74.12	75.25	78.44	78.93
+ *MCLS*	63.15	86.64	92.28	74.83	75.91	78.82	79.38
+ *SANTA*	60.80	85.59	91.07	73.41	74.59	78.08	78.48
+ SEAL	**64.30**	**87.15**	**92.88**	**75.54**	**76.57**	**79.17**	**79.83**
BGE-M3	61.03	85.24	91.69	73.35	73.96	77.97	79.41
+ *Chunk*	64.28	87.69	93.19	76.11	76.91	80.76	81.52
+ *MCLS*	65.27	87.98	93.48	76.74	77.37	81.14	82.05
+ *SANTA*	63.27	87.42	92.91	75.48	76.44	80.38	81.00
+ SEAL	**66.26**	**88.25**	**93.77**	**77.38**	**77.84**	**81.52**	**82.59**

Table 3: Retrieval effectiveness of different models on the industrial dataset.

To adapt Masked Entity Prediction of SANTA for the latest encoder-only models like BGE-M3, we use the same tool to identify co-occurring terms in the Query, Title, and Body as entities, apply random masking, and utilize the model to predict the masked entity tokens.

Implementations We begin with the fine-tuning of contrastive learning of the base PLMs. All the methods demonstrate performance improvements over the fine-tuned PLMs. We use the Adam optimizer with a learning rate of 1e-5 and 2 training epochs. The maximum query length is 32, and the maximum sequence length is 4096. We sample 8 negative samples for each query and use cross device negatives, the total batch size is 8. All the implementations utilize PyTorch and FlagEmbedding[1]. The experiments are performed on 4 NVIDIA A800 GPUs.

4.2 Overall Performance

Retrieval effectiveness is evaluated on two distinct datasets: a real industrial dataset and the StructDocRetrieval web dataset, utilizing three different base embedding models (mE5-large, bge-large-zh, and BGE-M3). We summarize the performance results across various standard metrics, including Hitrate@k, MRR@k, and NDCG@k.

Table 3 presents comparative retrieval effectiveness across real industrial structured documents, where SEAL achieves the state-of-the-art performance with HitRate@3 absolute gains of 5.67% over fine-tuned baselines and 3.53% over existing structural-aware methods. This performance advantage persists in web-crawled retrieval dataset StructDocRetrieval (cf. Table 4), particularly in high-recall metrics (i.e., HitRate@5: +1.70% avg.) and precision-sensitive measures (i.e., NDCG@10: +2.33% avg.). The consistent improvements across both controlled industrial and diverse web environments validate that SEAL enables the advantages of PLMs in representing long structured documents, making PLMs sensitive to document structures and better at representing structured data.

Method	HitRate@5	MRR@10	NDCG@10
mE5-large	92.89	83.02	86.24
+ *Chunk*	93.95	84.90	87.67
+ *MCLS*	94.16	85.39	88.38
+ *SANTA*	93.58	84.31	87.31
+ SEAL	**94.72**	**86.53**	**89.31**
bge-large-zh	94.59	85.95	88.68
+ *Chunk*	95.65	87.83	90.11
+ *MCLS*	95.86	88.32	90.82
+ *SANTA*	95.28	87.24	89.75
+ SEAL	**96.42**	**89.46**	**91.75**
BGE-M3	95.39	87.36	89.92
+ *Chunk*	96.45	89.24	91.35
+ *MCLS*	96.82	89.78	91.95
+ *SANTA*	95.98	88.45	90.83
+ SEAL	**97.09**	**90.10**	**92.25**

Table 4: The retrieval performance on StructDocRetrieval.

4.3 In-depth Analysis

In this subsection, we present an in-depth analysis, including ablation studies to investigate the con-

[1]https://github.com/FlagOpen/FlagEmbedding

Method	HitRate@5	MRR@10	NDCG@10
BGE-M3	91.69	73.96	79.41
w/ SAL	91.98	74.69	80.08
w/ EAL	92.12	75.83	80.85
w/ SEAL	**93.77**	**77.84**	**82.59**

Table 5: The performance of ablation models on industrial structured document retrieval.

ratios (%)	HitRate@5	MRR@10	NDCG@10
1	86.71	75.13	80.67
5	93.24	77.11	81.98
10	**93.77**	**77.84**	**82.59**
30	93.17	76.95	82.31
50	92.99	76.66	81.67

Table 6: The impact of element-aware alignment mask ratios on BGE-M3.

Method	HitRate@5	MRR@10	NDCG@10
BGE-M3	91.69	73.96	79.41
$SAL - EAL$	91.76	74.73	80.06
$SAL^1 - EAL^2$	93.27	76.98	81.96
$EAL^1 - SAL^2$	**93.77**	**77.84**	**82.59**

Table 7: The retrieval performance of different training strategies. The "$SAL - EAL$" means performing structure-aware and element-aware learning simultaneously. "$SAL^1 - EAL^2$" indicates that structure-aware learning is performed first, followed by element-aware alignment, while "$EAL^1 - SAL^2$" is the opposite.

tributions of two fundamental components of our method, an investigation of mask ratios in element-aware alignment, a comparison of training strategies, visualizations of the learned embedding distributions, and validation with an extended-context model.

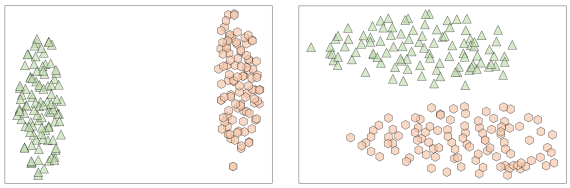
Ablation Study In this work, we employ structure-aware learning (SAL) and element-aware alignment (EAL) to conduct continuous training on the BGE-M3 model, demonstrating their effectiveness in guiding the model to better learn semantic features from structured documents. Table 5 presents the retrieval performance of these two fundamental components.

Compared to the baseline model, SAL and EAL exhibit divergent performance in structured data retrieval tasks. SAL shows no significant improvement over the baseline, restricted by the dependence on tag awareness alone to distinguish structured documents. In contrast, EAL achieves substantial enhancements in all the evaluation metrics. The superiority of EAL is due to its contrastive training paradigm between structural elements (e.g., HTML tags) and unstructured text components. This methodology effectively bridges the modality gap between heterogeneous data types through joint embedding space projection, thereby facilitating cross-modal representation learning with enhanced retrieval efficacy.

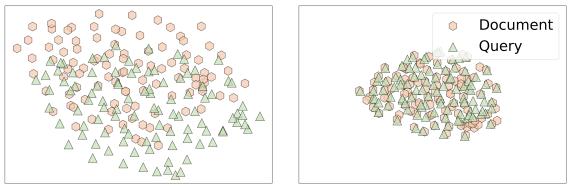
Overall, integrating augmented tasks through SAL and EAL yields progressive performance gains. These empirical results confirm that explicit structural awareness enables models to better encode semantic hierarchies while generating optimized textual representations tailored for complex structured data environments.

Impact of Mask Ratios We further investigate the impact of different element mask ratios in the element-aware alignment on retrieval effectiveness to determine the optimal configuration. Five distinct mask ratios (1%, 5%, 10%, 30%, and 50%) are evaluated using the BGE-M3 model with performance metrics including HitRate@5, NDCG@10, and MRR@10. The Experimental results are shown in Table 6. It can be observed that (1) the 10% mask ratio achieves optimal performance across all the metrics, and (2) performance variations remain marginal across different ratios, such as NDCG@10 differences constrained within a 2% range, demonstrating the robustness of our method to mask ratio selection. Based on these findings, we adopt the 10% masking ratio as the default configuration in the experimental section.

Impact of Training Strategy During continual fine-tuning, our empirical analysis reveals that simply combining L_{EAL} and L_{SAL} as $\mathcal{L} = (\mathcal{L}_{SAL} + \mathcal{L}_{EAL})$ does not achieve the best performance. Therefore, we make explorations for the impacts of different optimization strategies. As shown in Table 7, $EAL^1 - SAL^2$ achieves superior performance. The superiority stems from: (1) Local Semantics Foundation: EAL learns local semantic sensitivity by randomly masking element tags (e.g., <h1>), providing a high-quality textual representation for subsequent structural understanding. (2) Progressive Difficulty: EAL aligns local elements with queries (simpler task), while SAL integrates global structure with query intent (complex task). The $EAL \rightarrow SAL$ sequence follows an easy-to-hard trajectory, thereby preventing premature overfitting to structural noise.



(a) BGE-M3 (b) BGE-M3 w/ SAL



(c) BGE-M3 w/ EAL (d) BGE-M3 w/ SEAL

Figure 3: Embedding visualization of original model and SEAL using T-SNE.

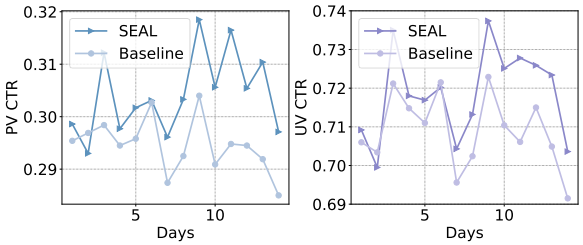


Figure 4: Online PV CTR and UV CTR over a two-Week period.

Method	HitRate@5	MRR@10	NDCG@10
mE5-large	92.17	74.32	79.86
+ *Chunk*	93.25	77.18	81.68
+ *MCLS*	94.00	77.51	82.18
+ SEAL	**94.72**	**78.17**	**83.56**

Table 8: The retrieval performance of GTE-Qwen2-1.5B on StructDocRetrieval.

Visualization To evaluate the quality of the learned representations and validate the establishment of a unified embedding space for queries and structured documents, we visualize the latent representations of queries and their corresponding documentation texts in Figure 3. An ablation study is also presented to isolate the contributions of Structure-Aware Learning and Element-Aware Alignment.

Comparative analysis between Figure 3 (a) and 3 (b) reveals that the effect of incorporating SAL. This integration reduces the divergence between query and documentation embeddings, demonstrating enhanced capture of structural context. Figure 3 (c) illustrates that EAL significantly improves query-document alignment. This improvement is evidenced by increased semantic proximity and tighter cluster fusion between queries and documents in the latent space. This suggests that our fine-grained masked element alignment mechanism enhances the model's capacity to capture more specific semantic distinctions for the refined embedding space.

Finally, comparison of Figures 3 (a) and 3 (d) indicates that SEAL exhibits superior embedding homogeneity compared to the base model through its dual mechanisms of structure-aware learning and fine-grained element alignment. These visualizations demonstrate that SEAL effectively learns a more unified and well-structured representation learning for structured document retrieval.

Extended-context model The experimental validity is strengthened through robustness testing with extended-context models. We introduce GTE-Qwen2-1.5B (32k-token context window) as an extended baseline, demonstrating that: (1) observed performance improvements are consistent with our previous results; (2) model efficacy exhibits progressive scalability across varying context window specifications.

4.4 Online A/B Testing

SEAL has been deployed in the practical platform for long-document retrieval services with hundreds of thousands of daily active users. We compare the performance of SEAL and a baseline method that performs raw-text contrastive learning over a 14-day period, evaluated using both PV CTR (Page View Click-Through Rate, clicks of ads divided by impressions of ads views) and UV CTR (Unique Visitor Click-Through Rate, clicks of unique visitors divided by impressions of unique visitors). The former is used to evaluate how often a page is clicked when it is browsed, and the latter reflects the attractiveness of the page to users.

As shown in Figure 4, compared to the previously deployed model, our approach demonstrates performance improvements in online A/B testing. The experiment, which accounted for approximately 30% of search traffic, was conducted over a two-week period. SEAL achieves an average improvement of 1.6% and 1.2% in PV CTR and UV CTR without introducing additional overhead, and higher PV and UV CTR on 12 out of 14 days. The superior performance of SEAL in

both metrics demonstrates its effective optimization of both page-level attractiveness and user-level engagement, ultimately enhancing the user experience in practical applications.

5 Conclusion

In this work, we propose SEAL, a novel contrastive framework that integrates structural awareness and fine-grained semantic alignment to enhance PLMs for long structured document retrieval. To foster research in this field, we release StructDocRetrieval, a dataset of long documents enriched with structural information, establishing an evaluation scenario close to real-world applications. Extensive experiments demonstrate SEAL achieves state-of-the-art performance in long structured document retrieval across various PLMs and datasets. Our in-depth analysis further reveals that SEAL induces a unified embedding space that effectively aligns queries and relevant documents.

Limitations

Although SEAL exhibits strong performance in long structured document retrieval, its reliance on alignment signals between structured and unstructured data raises open questions about its general superiority over baseline models in all downstream tasks, such as code retrieval. Additionally, our current empirical validation focuses primarily on the Chinese-language community, though we are actively constructing an English-language corpus to assess generalization capabilities. Finally, our approach preserves the potential of exploiting the document structure during pre-training.

Acknowledgments

This work is supported by the following funding sources: the Guangzhou Industrial Information and Intelligent Key Laboratory Project (No.2024A03J0628); the National Natural Science Foundation of China (NSFC) (No.62306256); the Natural Science Foundation of Guangdong Province (No.2025A1515010261); and the Key R&D Program of Zhejiang Province (No. 2024C01036). This work is also supported by Alibaba Group.

References

Yushi Bai, Xin Lv, Jiajie Zhang, Hongchang Lyu, Jiankai Tang, Zhidian Huang, Zhengxiao Du, Xiao Liu, Aohan Zeng, Lei Hou, Yuxiao Dong, Jie Tang, and Juanzi Li. 2024a. Longbench: A bilingual, multitask benchmark for long context understanding. In *ACL (1)*, pages 3119–3137. Association for Computational Linguistics.

Yushi Bai, Shangqing Tu, Jiajie Zhang, Hao Peng, Xiaozhi Wang, Xin Lv, Shulin Cao, Jiazheng Xu, Lei Hou, Yuxiao Dong, Jie Tang, and Juanzi Li. 2024b. Longbench v2: Towards deeper understanding and reasoning on realistic long-context multitasks. *CoRR*, abs/2412.15204.

Jianlyu Chen, Shitao Xiao, Peitian Zhang, Kun Luo, Defu Lian, and Zheng Liu. 2024. M3-embedding: Multi-linguality, multi-functionality, multi-granularity text embeddings through self-knowledge distillation. In *ACL (Findings)*, pages 2318–2335. Association for Computational Linguistics.

Nick Craswell, Bhaskar Mitra, Emine Yilmaz, Daniel Campos, and Ellen M. Voorhees. 2020. Overview of the TREC 2019 deep learning track. *CoRR*, abs/2003.07820.

Lu Dai, Hao Liu, and Hui Xiong. 2024. Improve dense passage retrieval with entailment tuning. In *EMNLP (1)*. Association for Computational Linguistics.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: pre-training of deep bidirectional transformers for language understanding. In *NAACL-HLT (1)*, pages 4171–4186. Association for Computational Linguistics.

Luyu Gao and Jamie Callan. 2021. Condenser: a pre-training architecture for dense retrieval. In *EMNLP (1)*, pages 981–993. Association for Computational Linguistics.

Shailja Gupta, Rajesh Ranjan, and Surya Narayan Singh. 2024. A comprehensive survey of retrieval-augmented generation (RAG): evolution, current landscape and future directions. *CoRR*, abs/2410.12837.

Cheng-Ping Hsieh, Simeng Sun, Samuel Kriman, Shantanu Acharya, Dima Rekesh, Fei Jia, Yang Zhang, and Boris Ginsburg. 2024. RULER: what's the real context size of your long-context language models? *CoRR*, abs/2404.06654.

Shengding Hu, Yuge Tu, Xu Han, Chaoqun He, Ganqu Cui, Xiang Long, Zhi Zheng, Yewei Fang, Yuxiang Huang, Weilin Zhao, Xinrong Zhang, Zhen Leng Thai, Kai Zhang, Chongyi Wang, Yuan Yao, Chenyang Zhao, Jie Zhou, Jie Cai, Zhongwu Zhai, Ning Ding, Chao Jia, Guoyang Zeng, Dahai Li, Zhiyuan Liu, and Maosong Sun. 2024. Minicpm: Unveiling the potential of small language models with scalable training strategies. *CoRR*, abs/2404.06395.

Jui-Ting Huang, Ashish Sharma, Shuying Sun, Li Xia, David Zhang, Philip Pronin, Janani Padmanabhan, Giuseppe Ottaviano, and Linjun Yang. 2020.

Embedding-based retrieval in facebook search. In *KDD*, pages 2553–2561. ACM.

Hamel Husain, Ho-Hsiang Wu, Tiferet Gazit, Miltiadis Allamanis, and Marc Brockschmidt. 2019. Codesearchnet challenge: Evaluating the state of semantic code search. *CoRR*, abs/1909.09436.

Vladimir Karpukhin, Barlas Oguz, Sewon Min, Patrick Lewis, Ledell Wu, Sergey Edunov, Danqi Chen, and Wen-tau Yih. 2020a. Dense passage retrieval for open-domain question answering. In *EMNLP (1)*, pages 6769–6781. Association for Computational Linguistics.

Vladimir Karpukhin, Barlas Oguz, Sewon Min, Patrick S. H. Lewis, Ledell Wu, Sergey Edunov, Danqi Chen, and Wen-tau Yih. 2020b. Dense passage retrieval for open-domain question answering. In *EMNLP (1)*, pages 6769–6781. Association for Computational Linguistics.

Omar Khattab and Matei Zaharia. 2020. Colbert: Efficient and effective passage search via contextualized late interaction over BERT. In *SIGIR*, pages 39–48. ACM.

Chankyu Lee, Rajarshi Roy, Mengyao Xu, Jonathan Raiman, Mohammad Shoeybi, Bryan Catanzaro, and Wei Ping. 2025. Nv-embed: Improved techniques for training llms as generalist embedding models. In *ICLR*. OpenReview.net.

Bohan Li, Hao Zhou, Junxian He, Mingxuan Wang, Yiming Yang, and Lei Li. 2020. On the sentence embeddings from pre-trained language models. In *EMNLP (1)*, pages 9119–9130. Association for Computational Linguistics.

Sen Li, Fuyu Lv, Taiwei Jin, Guli Lin, Keping Yang, Xiaoyi Zeng, Xiao-Ming Wu, and Qianli Ma. 2021a. Embedding-based product retrieval in taobao search. In *KDD*, pages 3181–3189. ACM.

Xiaonan Li, Yeyun Gong, Yelong Shen, Xipeng Qiu, Hang Zhang, Bolun Yao, Weizhen Qi, Daxin Jiang, Weizhu Chen, and Nan Duan. 2022. CodeRetriever: A large scale contrastive pre-training method for code search. In *ACL*, pages 2898–2910. Association for Computational Linguistics.

Xinze Li, Zhenghao Liu, Chenyan Xiong, Shi Yu, Yu Gu, Zhiyuan Liu, and Ge Yu. 2023a. Structure-aware language model pretraining improves dense retrieval on structured data. In *ACL (Findings)*, pages 11560–11574. Association for Computational Linguistics.

Xinze Li, Hanbin Wang, Zhenghao Liu, Shi Yu, Shuo Wang, Yukun Yan, Yukai Fu, Yu Gu, and Ge Yu. 2025. Building a coding assistant via the retrieval-augmented language model. *ACM Trans. Inf. Syst.*, 43(2):39:1–39:25.

Yizhi Li, Zhenghao Liu, Chenyan Xiong, and Zhiyuan Liu. 2021b. More robust dense retrieval with contrastive dual learning. In *ICTIR*, pages 287–296. ACM.

Zehan Li, Xin Zhang, Yanzhao Zhang, Dingkun Long, Pengjun Xie, and Meishan Zhang. 2023b. Towards general text embeddings with multi-stage contrastive learning. *CoRR*, abs/2308.03281.

Tri Nguyen, Mir Rosenberg, Xia Song, Jianfeng Gao, Saurabh Tiwary, Rangan Majumder, and Li Deng. 2016. MS MARCO: A human generated machine reading comprehension dataset. In *CoCo@NIPS*, volume 1773 of *CEUR Workshop Proceedings*. CEUR-WS.org.

Jun Rao, Liang Ding, Shuhan Qi, Meng Fang, Yang Liu, Li Shen, and Dacheng Tao. 2023. Dynamic contrastive distillation for image-text retrieval. *IEEE Transactions on Multimedia*, 25:8383–8395.

Jun Rao, Zepeng Lin, Xuebo Liu, Xiaopeng Ke, Lian Lian, Dong Jin, Shengjun Cheng, Jun Yu, and Min Zhang. 2025. APT: improving specialist LLM performance with weakness case acquisition and iterative preference training. In *ACL (Findings)*, pages 20958–20980. Association for Computational Linguistics.

Jun Rao, Fei Wang, Liang Ding, Shuhan Qi, Yibing Zhan, Weifeng Liu, and Dacheng Tao. 2022. Where does the performance improvement come from - a reproducibility concern about image-text retrieval. In *SIGIR*.

Chandan K. Reddy, Lluís Màrquez, Fran Valero, Nikhil Rao, Hugo Zaragoza, Sambaran Bandyopadhyay, Arnab Biswas, Anlu Xing, and Karthik Subbian. 2022. Shopping queries dataset: A large-scale ESCI benchmark for improving product search. *CoRR*, abs/2206.06588.

Christopher Sciavolino, Zexuan Zhong, Jinhyuk Lee, and Danqi Chen. 2021. Simple entity-centric questions challenge dense retrievers. In *EMNLP (1)*, pages 6138–6148. Association for Computational Linguistics.

Dong Sun, Wenya Guo, Xumeng Liu, Ying Zhang, Zhaoxiang Hou, and Zengxiang Li. 2025. Zero-shot document retrieval with hybrid pseudo-document retriever. In *ICASSP 2025-2025 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 1–5. IEEE.

Jiejun Tan, Zhicheng Dou, Wen Wang, Mang Wang, Weipeng Chen, and Ji-Rong Wen. 2025. Htmlrag: HTML is better than plain text for modeling retrieved knowledge in RAG systems. In *WWW*, pages 1733–1746. ACM.

Nandan Thakur, Nils Reimers, Andreas Rücklé, Abhishek Srivastava, and Iryna Gurevych. 2021. BEIR: A heterogeneous benchmark for zero-shot evaluation of information retrieval models. In *NeurIPS Datasets and Benchmarks*.

Liang Wang, Nan Yang, Xiaolong Huang, Linjun Yang, Rangan Majumder, and Furu Wei. 2024. Multilingual E5 text embeddings: A technical report. *CoRR*, abs/2402.05672.

Yue Wang, Weishi Wang, Shafiq R. Joty, and Steven C. H. Hoi. 2021. Codet5: Identifier-aware unified pre-trained encoder-decoder models for code understanding and generation. In *EMNLP (1)*, pages 8696–8708. Association for Computational Linguistics.

Benjamin Warner, Antoine Chaffin, Benjamin Clavié, Orion Weller, Oskar Hallström, Said Taghadouini, Alexis Gallagher, Raja Biswas, Faisal Ladhak, Tom Aarsen, Nathan Cooper, Griffin Adams, Jeremy Howard, and Iacopo Poli. 2024. Smarter, better, faster, longer: A modern bidirectional encoder for fast, memory efficient, and long context finetuning and inference. *CoRR*, abs/2412.13663.

Shitao Xiao, Zheng Liu, Peitian Zhang, and Niklas Muennighoff. 2023. C-pack: Packaged resources to advance general chinese embedding. *CoRR*, abs/2309.07597.

Lee Xiong, Chenyan Xiong, Ye Li, Kwok-Fung Tang, Jialin Liu, Paul N. Bennett, Junaid Ahmed, and Arnold Overwijk. 2021a. Approximate nearest neighbor negative contrastive learning for dense text retrieval. In *ICLR*. OpenReview.net.

Wenhan Xiong, Xiang Lorraine Li, Srini Iyer, Jingfei Du, Patrick S. H. Lewis, William Yang Wang, Yashar Mehdad, Scott Yih, Sebastian Riedel, Douwe Kiela, and Barlas Oguz. 2021b. Answering complex open-domain questions with multi-hop dense retrieval. In *ICLR*. OpenReview.net.

Junhan Yang, Zheng Liu, Chaozhuo Li, Guangzhong Sun, and Xing Xie. 2023. Longtriever: a pre-trained long text encoder for dense document retrieval. In *EMNLP*, pages 3655–3665. Association for Computational Linguistics.

Wayne Xin Zhao, Jing Liu, Ruiyang Ren, and Ji-Rong Wen. 2024. Dense text retrieval based on pretrained language models: A survey. *ACM Trans. Inf. Syst.*, 42(4):89:1–89:60.