Exploring Knowledge Filtering for Retrieval-Augmented Discriminative Tasks

Minjie Qiang[1], **Zhongqing Wang**[1*], **Xiaoyi Bao**[2], **Haoyuan Ma**[1]
Shoushan Li[1], **Guodong Zhou**[1]
[1]Natural Language Processing Lab, Soochow University, Suzhou, China
[2]The Hong Kong Polytechnic University, Hong Kong, China
qiangminjie27@gmail.com
{wangzq,lishoushan,gdzhou}@suda.edu.cn
p2213545413@outlook.com, 20244027009@stu.suda.edu.cn

Abstract

Retrieval-augmented methods have achieved remarkable advancements in alleviating the hallucination of large language models. Nevertheless, the introduction of external knowledge does not always lead to the expected improvement in model performance, as irrelevant or harmful information present in the retrieved knowledge can compromise the prediction process. To address these challenges, we propose a novel framework aimed at improving model performance by incorporating knowledge filtering and prediction fusion mechanisms. In particular, our approach first employs a perplexity-based annotation method to collect training data. Then, we design four distinct strategies to filter out harmful retrieved knowledge. Finally, we integrate the filtered knowledge to generate the final result via batch-wise predictions. We conduct extensive experiments across multiple discriminative task datasets to evaluate the proposed framework. The results demonstrate that our framework can significantly enhance the performance of models on discriminative tasks.

1 Introduction

Recently, large language models have demonstrated remarkable potential across various natural language processing (NLP) tasks (Wang et al., 2023a; Boshar et al., 2024; Hasan et al., 2024). However, they also exhibit several critical limitations, including the propensity to generate hallucinated content (Zhou et al., 2021) and the inability to update their internal knowledge dynamically (Kandpal et al., 2023). To mitigate these problems, Retrieval-Augmented Generation (RAG) has been introduced as a promising approach. RAG enhances LLMs by retrieving relevant information from external sources, such as Wikipedia, and integrating it into the input context, thereby improving response accuracy and reliability (Fan et al., 2024).

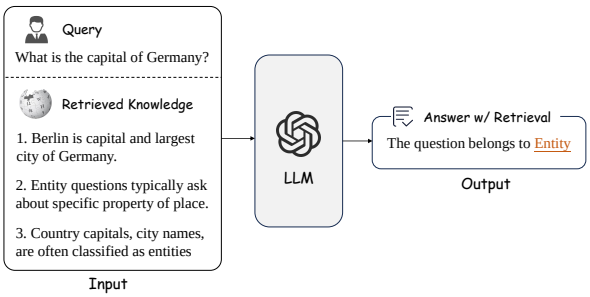

Figure 1: The illustration of RAG in discriminative tasks.

Most research on RAG typically follows the Retrieve-then-Read paradigm and has achieved significant success across multiple tasks, such as dialogue generation (Wu et al., 2019) and question answering (Izacard and Grave, 2020). Nevertheless, most recent studies have primarily focused on generative tasks, while its potential in discriminative tasks remains largely unexplored. Notably, RAG holds considerable promise for discriminative tasks such as sentiment analysis, as retrieving relevant knowledge can provide implicit meanings, cultural context, and domain-specific concepts, thereby facilitating more accurate classification. The investigation of discriminative tasks not only broadens the application scope of RAG but also offers valuable insights into how retrieved knowledge contributes to enhanced discrimination performance. As illustrated in Figure 1, our study explores the application of RAG across a range of discriminative tasks, including linguistic acceptablity (Warstadt, 2019), question classification (Li and Roth, 2002), and sentiment analysis (Socher et al., 2013).

However, applying RAG to discriminative tasks presents two significant challenges: On the one hand, retrieval-augmented methods are constrained by the quality of retrieved knowledge (Cheng et al., 2023). The presence of noise in external knowledge bases, coupled with the performance limitations of retrievers, may result in retrieved knowledge con-

* Zhongqing Wang is the corresponding author

Findings of the Association for Computational Linguistics: ACL 2025, pages 1716–1729

taining irrelevant or even contradictory information. For instance, in question classification tasks, irrelevant knowledge may introduce spurious correlations, causing the model to overlook essential classification features. On the other hand, retrieval-augmented methods are constrained by their prediction strategy (Shi et al., 2023b). Incorporating all retrieved knowledge indiscriminately may confuse the model. For example, in sentiment classification tasks, the presence of knowledge with opposing sentiments is inevitable. This can lead the model to make incorrect judgments as it becomes uncertain which knowledge to trust.

To address these challenges, we propose a novel framework as illustrated in Figure 2. Our framework enhances performance through knowledge filtering and prediction fusion, comprising three core steps: 1) **Label Collection**, annotating harmfulness labels of knowledge for training. 2) **Knowledge Filtering**, implementing four distinct strategies for knowledge filtering. 3) **Prediction Fusion**, mitigating the potential bias introduced by knowledge. Specifically, we first annotate query-knowledge pairs by analyzing variations in the model perplexity with and without knowledge to collect training data. Then, leveraging the annotated training data, we design four filtering strategies to eliminate harmful knowledge. Finally, we conduct batch-wise predictions using the filtered knowledge and employ a weighted fusion mechanism to generate the final output. Extensive experiments across multiple discriminative tasks demonstrate the generality and effectiveness of our framework.

2 Task Formulation & Framework

In this study, we explore the application of RAG across multiple discriminative tasks: 1) **Linguistic Acceptability** (Warstadt, 2019), which evaluates whether the given sentence follows grammar standards. 2) **Question Classification** (Li and Roth, 2002), which aims to categorize questions into predefined topics (e.g., location). 3) **Sentiment Analysis** (Socher et al., 2013), which determines the specific sentence as positive, negative, or neutral. Formally, these tasks can be defined as discriminative problems. A unified definition is as follows: Given a text input x, the objective is to predict an output $\hat{y}$ from a pre-defined label set Y.

Figure 2 provides an overview of our proposed framework. Specifically, given a text input x as a query, we first employ a retriever $R(\cdot)$ to retrieve a relevant subset of knowledge from an external database K, where Wikipedia serves as our knowledge source. Then, we employ four filtering strategies, leveraging annotated training data, to eliminate harmful knowledge. Finally, the filtered knowledge is concatenated with the original query and fed into a large language model $\mathcal{M}(\cdot)$ to generate the output.

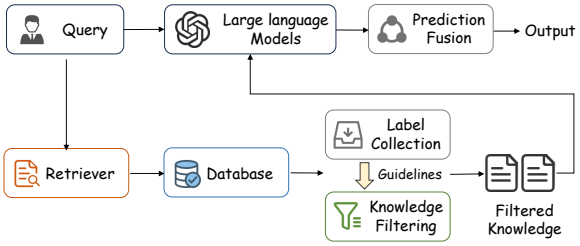

Figure 2: Overview of our framework.

In the following section, we introduce the two fundamental components of the RAG pipeline:

Retriever We utilize the dense encoder BGE (Xiao et al., 2023) as our retriever. To optimize retrieval costs, we establish a pre-built static index for the external database by encoding knowledge. Then, the queries are embedded into the same latent vector space. The similarity score $\text{sim}(q, k_i)$ between the query vector $E(q)$ and each knowledge vector $E(k_i)$ is computed using dot product, followed by efficient similarity retrieval within the static index through FAISS (Johnson et al., 2019):

$$\text{sim}(q, k_i) = E(q)^T E(k_i), k_i \in K \quad (1)$$

Finally, based on these similarity scores, we select the top-k most relevant knowledge for the next process stage.

Generator We adopt LLaMA3-8B (Dubey et al., 2024) as the backbone of our framework. Following the retrieval phase, we concatenate the query with the retrieved knowledge K_{ret} to construct a prompt, which is then fed into a large language model to generate the final output $\hat{y}(q, K_{ret})$:

$$\hat{y}(q, K_{ret}) = \mathcal{M}(\text{Prompt}(q, K_{ret})) \quad (2)$$

Note that the generator's performance remains suboptimal without task-specific fine-tuning, as evidenced by our experimental results in Appendix C.

3 Methodology

As illustrated in Figure 3, our framework comprises three primary steps: *Label Collection*, *Knowledge Filtering*, and *Prediction Fusion*. Specifically, we

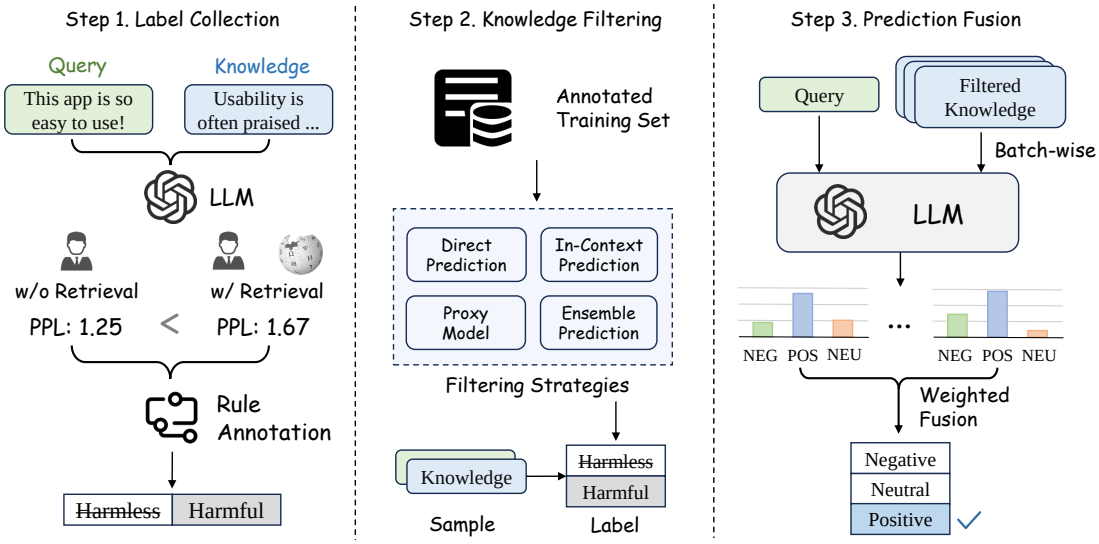

Figure 3: The illustration of each step in our framework.

first propose a perplexity-based approach to annotate query-knowledge pairs, collecting training data for subsequent filtering procedures. We then design four distinct strategies to filter out harmful knowledge. Finally, we integrate the filtered knowledge through the batch-wise prediction to achieve more accurate classification. The following sections provide a detailed description of each step.

3.1 Label Collection

Since subsequent filtering strategies require labeled training samples, we propose an annotation approach that evaluates the harmfulness of knowledge by analyzing the perplexity variation of the model with and without external knowledge.

Specifically, given a subset $\mathcal{D} = \{(q_i, y_i)\}_{i=1}^{|\mathcal{D}|}$ from training samples, we first employ the fine-tuned LLM to generate the output for each query:

$$\hat{y}(q_i) = \mathcal{M}(\text{Prompt}(q_i)) \tag{3}$$

The generated output $\hat{y}(q_i)$ reflects the model's intrinsic knowledge level for the given query (Varshney et al., 2022).

Subsequently, we utilize the retriever to obtain the top-k most relevant knowledge from the external database K and incorporate each retrieved knowledge into the prompt. Then we employ LLM to generate output $\hat{y}(q_i, k_i)$ with retrieved knowledge. To annotate harmfulness for query-knowledge pairs, we set a threshold θ and compare the model perplexity PPL$(\cdot)$ variation between the

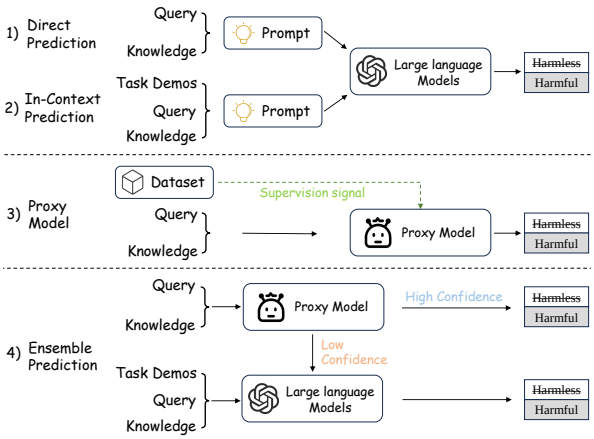

Figure 4: The illustration of four filtering strategies.

two generation settings:

$$(q_i, k_i) = \begin{cases} \text{Harmful}, & \text{if } \frac{\text{PPL}(\hat{y}(q_i,k_i))}{\text{PPL}(\hat{y}(q_i))} > \theta; \\ \text{Harmless}, & \text{otherwise.} \end{cases} \tag{4}$$

Through this process, we ultimately construct a training dataset $\mathcal{D}_{\text{kf}}$ comprising query-knowledge pairs annotated for harmfulness.

3.2 Knowledge Filtering

As illustrated in Figure 4, we propose four distinct strategies to filter retrieved knowledge: *Direct Prediction*, *In-Context Prediction*, *Proxy Model*, and *Ensemble Prediction*. A detailed description of each strategy is provided below.

Direct Prediction Given a query and a piece of knowledge, the objective is to evaluate the harmfulness of retrieved knowledge. A straightforward approach is to construct a prompt template, en-

abling the LLM itself to directly assess the harmfulness of the provided knowledge. Here, we adopt the Multiple-Choice Question (MCQ) prompt template (Wang et al., 2023b), as LLMs generally exhibit greater familiarity with the MCQ format compared to the conventional prompts. The set of prompt templates can be found in Appendix E.

In the MCQ format, each predefined label is mapped to a corresponding option, which is then presented to the LLM for selection. This strategy is both simple and intuitive and adapts well across various tasks. Nevertheless, due to the absence of explicit task-specific demonstrations, the model's performance may be inherently constrained.

In-Context Prediction Recent studies have demonstrated that incorporating task-specific demonstrations into prompts can significantly enhance the performance of LLMs without fine-tuning (Zhang et al., 2022). Therefore, we randomly sample several query-knowledge pairs from $\mathcal{D}_{kf}$ and integrate them into the prompt as task-specific demonstrations. Subsequently, we leverage the LLM itself to assess the harmfulness of the provided knowledge.

Both of the aforementioned strategies that utilize the LLM itself for filtering exhibit several limitations: 1) Context-based methods are inherently susceptible to the quality of provided demonstrations (Fan et al., 2024), frequently resulting in suboptimal performance. 2) These methods are constrained by the LLM's context length limitations, making it struggle to utilize more training samples.

Proxy Model To overcome these limitations, we propose a proxy model strategy, which is implemented by fine-tuning a lightweight LLaMA3-1B (Dubey et al., 2024) on the dataset $\mathcal{D}_{\text{kf}}$. Specifically, we formulate the harmfulness evaluation as a text generation task, where given a query q and a piece of knowledge k_i, the proxy model $\text{PM}(\cdot)$ generates discriminative predictions. The prediction process is formally defined as:

$$\hat{a}(q, k_i) = \text{PM}(\text{Prompt}(q, k_i)) \tag{5}$$

where $\hat{a}(q, k_i) \in \{\text{Harmful}, \text{Harmless}\}$ indicates the harmfulness of knowledge. By leveraging a lightweight proxy model, our strategy enhances the efficiency and accuracy of harmfulness evaluation.

Ensemble Prediction Existing research (Ma et al., 2023) has demonstrated that LLMs exhibit superior performance in handling difficult samples, while small language models (i.e., our proxy model) tend to perform more effectively on simpler samples. Motivated by this, we suppose that our proposed filtering strategies are not mutually exclusive but complementary. To leverage the strengths of distinct strategies, we propose an ensemble prediction strategy. Specifically, we first employ the proxy model for initial sample predictions:

$$\text{conf}(k_i) = \max_{a \in A} \text{Prob}_{PM}(a|q, k_i) \tag{6}$$

where $\text{conf}(k_i)$ denotes the confidence level of the given sample, A is the set of candidate labels and $\text{Prob}_{PM}(a|q, k_i)$ represents the conditional probability distribution generated by the proxy model. We set a threshold θ to distinguish the difficulty of samples. For high-confidence samples ($\text{conf}(k_i) \geq \theta$), we directly adopt the proxy model's predictions. In contrast, for low-confidence samples, we further apply *In-Context Prediction* strategy, thereby integrating both strategies in a complementary manner.

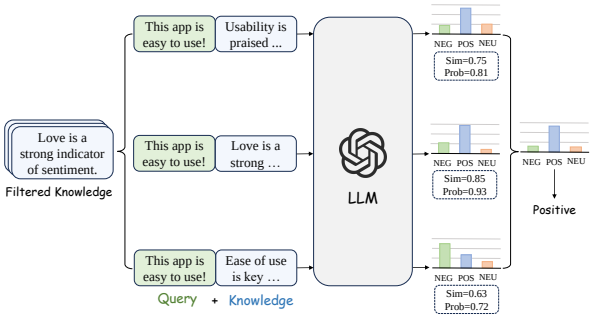

Figure 5: The illustration of prediction fusion.

3.3 Prediction Fusion

To mitigate the problem that the LLMs are susceptible to being misled by certain pieces of knowledge, we propose the *Prediction Fusion* mechanism, as illustrated in Figure 5. Specifically, we first integrate the query with each filtered knowledge separately to construct prompts and then perform batch-wise predictions. Subsequently, we apply a weighted fusion strategy to aggregate the probability distributions $p(y|q, k_i)$ generated by the LLM, yielding the final prediction:

$$p(y|q, K'_{\text{ret}}) = \sum_{k_i \in K'_{\text{ret}}} \lambda(q, k_i) \cdot p(y|q, k_i) \tag{7}$$

where K'_{ret} denotes the filtered knowledge set. Note that when $K'_{\text{ret}} = \varnothing$, our method degenerates into the standard non-retrieval generation. Furthermore, $\lambda(q, k_i)$ represents the weight assigned to each

Methods	SST-2	SST-5	CR	MR	MPQA	CoLA	TREC	Average
w/o Retrieval								
T5-Large	92.43	48.39	90.75	78.35	84.10	64.60	76.20	76.40
Roberta-Large	74.74	40.60	89.40	81.15	57.85	67.21	73.00	69.14
ICL*	91.17	46.00	73.20	61.70	71.60	72.29	74.80	70.11
LM-BFF*	90.80	49.00	89.05	83.50	86.40	53.69	70.60	74.72
w/ Retrieval								
GPT-4o	91.63	52.01	90.50	88.65	87.20	78.62	73.20	80.26
Direct RAG	91.74	52.56	89.35	87.05	86.30	77.47	90.80	82.18
FiD	91.05	52.46	90.20	78.35	84.10	64.62	76.20	76.71
PGRA	92.43	53.82	90.30	78.00	82.95	70.80	76.00	77.76
RECOMP	92.31	53.64	90.65	87.75	87.85	76.99	89.20	82.63
SKR	93.34	54.09	91.50	88.45	86.95	78.04	88.60	83.00
Ours	**94.50**	**55.68**	**92.85**	**90.20**	**89.45**	**81.78**	**93.40**	**85.41**

Table 1: Comparison with baselines. * denoted we partially refer to the results from other papers (Guo et al., 2023) .

piece of knowledge, which is computed as follows:

$$\lambda(q, k_i) = \alpha \text{sim}(q, k_i) + (1 - \alpha) P_x(a|q, k_i) \quad (8)$$

where $P_x(a|q, k_i)$ corresponds to the probability of harmlessness estimated by the LLM or the proxy model, and α controls the balance of two factors.

4 Experiment

4.1 Datasets & Metric

This study evaluates our proposed framework using datasets from three distinct text classification tasks: (1) Linguistic Acceptability: **CoLA** (Warstadt, 2019), assessing the grammatical acceptability of English sentences. (2) Question Classification: **TREC** (Li and Roth, 2002), classifying questions into six predefined categories. (3) Sentiment Analysis: **SST-2** and **SST-5** (Socher et al., 2013) for determining the sentiment polarity of sentences into two or five categories; **MR** (Pang and Lee, 2004) from the movie review domain; **MPQA** (Wiebe et al., 2005) from the news opinion domain; and **CR** (Ding et al., 2008) from the product review domain. We adopt accuracy as the evaluation metric for all datasets and follow the data splits from LM-BFF (Gao et al., 2020). The statistics of the datasets are presented in Table 2, while a detailed description of each dataset is provided in Table 6 in the appendix. The implementation details of our approach are discussed in Appendix A.

Datasets	Train	Dev	Test	Label	Len
CoLA	32	32	1,043	2	8
TREC	96	96	500	6	10
SST-2	32	32	872	2	19
SST-5	79	79	2,209	5	18
MR	32	32	2,000	2	20
MPQA	32	32	2,000	2	3
CR	32	32	2,000	2	19

Table 2: Statistics of datasets. **Label** denotes the number of candidate labels. **Len** represents the length of the input content.

4.2 Main Result

This section conducts a comparison between our proposed framework and methods both with and without retrieval. For methods without retrieval, **T5-Large** (Raffel et al., 2020) and **Roberta-Large** (Liu, 2019) serve as representative pre-trained models; **LM-BFF** (Gao et al., 2021) employs prompt-based fine-tuning to enhance few-shot performance; **ICL** utilizes OPT-13B (Zhang et al., 2023) with eight task-specific demonstrations without fine-tuning; For methods with retrieval, **Direct RAG** incorporates retrieved knowledge directly into the LLaMA3-8B model's prompt for generation; **GPT-4o** (OpenAI, 2024) is the state-of-the-art large language model released by OpenAI; **FiD** (Izacard and Grave, 2021) adopts a generative approach by aggregating information from multiple documents; **PGRA** (Guo et al., 2023) employs a two-stage framework for re-ranking retrieved knowledge; **SKR** (Wang et al., 2023c) retrieves external knowledge based on problem's complexity adaptively; **RECOMP** (Xu et al., 2023) enhances performance by compressing retrieved knowledge.

As demonstrated in Table 1, large language models relying exclusively on ICL show significant limitations in achieving competitive performance, even underperforming small models (e.g., T5-Large). This can be attributed to ICL's sensitivity to demonstration quality, which often results in suboptimal performance. Our analysis reveals that retrieval-augmented methods consistently outperform those without retrieval, confirming the necessity of external knowledge for discriminative tasks. Notably, while advanced retrieval-augmented methods such

Methods	SST-2	SST-5	CR	MR	MPQA	CoLA	TREC	Average
Direct RAG	91.74	52.56	89.35	87.05	86.30	77.47	90.80	82.18
w/ Knowledge Filtering								
Direct Prediction	92.07	51.65	91.10	87.15	87.95	77.95	90.20	82.58
In-Context Prediction	93.12	52.92	91.50	88.25	87.50	76.51	90.80	82.94
Proxy Model	92.20	53.28	91.45	87.80	88.50	79.58	89.60	83.20
Ensemble Prediction	93.00	54.14	91.90	87.60	88.40	80.15	92.00	83.88
w/ Prediction Fusion								
Ours	**94.50**	**55.68**	**92.85**	**90.20**	**89.45**	**81.78**	**93.40**	**85.41**

Table 3: The effects of knowledge filtering and prediction fusion.

Methods	CR			MPQA			MR			SST-5		
	P	R	F	P	R	F	P	R	F	P	R	F
Direct	41.7	11.6	18.2	39.3	23.9	29.7	49.4	30.3	37.6	48.5	40.4	44.1
In-Context	59.4	95.3	73.2	51.7	100.0	68.1	58.3	94.4	72.0	54.0	66.9	59.8
Proxy	86.0	100.0	92.5	83.7	89.1	86.3	86.5	94.4	90.2	84.9	88.8	86.8
Ensemble	89.5	100.0	94.5	82.1	100.0	90.2	87.7	95.8	90.9	88.1	92.9	90.4

Table 4: The performance of strategies on knowledge filtering.

as *SKR* and *RECOMP* incorporate sophisticated mechanisms like problem filtering and knowledge compression, their performance improvements over *Direct RAG* remain marginal. This suggests that these advanced strategies still struggle to eliminate the influence of harmful knowledge.

In comparison, our framework achieves significant performance improvements over baseline models across all tasks, demonstrating its superior effectiveness and robust generalization capabilities.

4.3 Impact of Filtering Strategies and Prediction Fusion

This section conducts an ablation study to evaluate two critical components of our framework. We take *Direct RAG* as the baseline and investigate the effectiveness of four distinct filtering strategies alongside prediction fusion.

As illustrated in Table 3, the performance of *Direct Prediction* barely surpasses the baseline, suggesting that LLMs struggle to assess the harmfulness of knowledge by themselves. When supplemented with task demonstrations for harmful knowledge identification, the performance of *In-Context Prediction* achieves improvement across most datasets. However, the decline in performance observed in certain datasets can be attributed to the suboptimal quality of task demonstrations. *Proxy Model* demonstrates superior performance compared to both LLM-based approaches, indicating that fine-tuning a smaller model can achieve higher accuracy without manually providing demonstrations. *Ensemble Prediction* outperforms all other strategies, demonstrating the complementary nature of our proposed methods. Dynamic adjustment of strategies based on sample difficulty enables full exploitation of their respective advantages.

While knowledge filtering has achieved considerable performance improvement, the integration of prediction fusion yields optimal results. This indicates that weighted consideration of each piece of knowledge can indeed mitigate the risk of model misguidance by a single piece of knowledge.

5 Analysis and Discussion

5.1 Performance of Strategies on Knowledge Filtering

Although we have demonstrated the effectiveness of the filtering strategies, it remains unclear whether the improvement stems from the removal of harmful knowledge. To address this gap, we conduct a comprehensive analysis across four representative datasets and evaluate the performance of four strategies in filtering harmful knowledge using F1 scores. Given that the objective of filtering is to prevent the introduction of harmful knowledge, we provide the performance of strategies on "Harmful" label rather than the overall.

As shown in Table 4, *Direct* exhibits consistently inferior performance across evaluation metrics, indicating that LLMs struggle to accurately assess the harmfulness of knowledge without demonstrations. *In-Context* achieves a substantial improvement in Recall (even reaching 100%) by incorporating task demonstrations. However, this comes at the expense of Precision, indicating excessive false positives in classification. *Proxy* further improves Precision while maintaining Recall. *Ensemble* achieves the optimal filtering performance by integrating

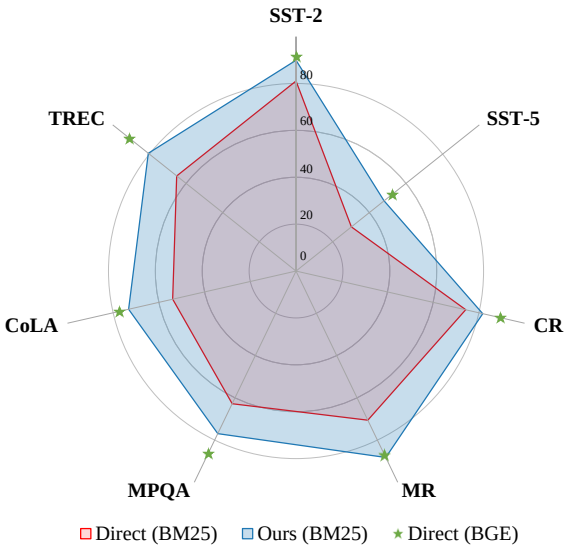

Figure 6: The performance of different retrievers.

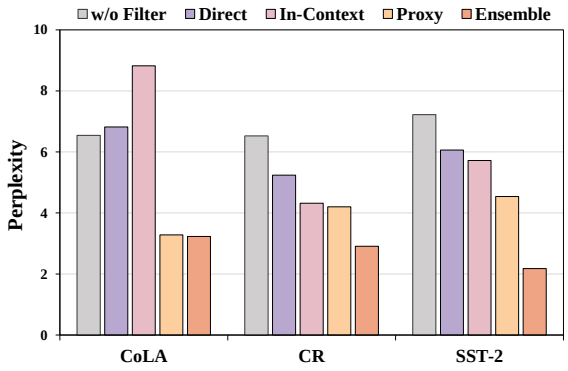

Figure 7: The influence of strategies on perplexity.

the strengths of these strategies, demonstrating that the proposed strategies are complementary and can jointly enhance filtering performance. Furthermore, we observe a positive correlation between filtering effectiveness and overall model performance, suggesting that the model improvements are indeed driven by the elimination of harmful knowledge.

5.2 Effects of Different Retrievers

While our framework demonstrates effectiveness within dense retrievers (e.g., BGE), such retrievers are constrained by computational cost and storage requirements. This leads to a critical question: *Can our framework maintain its effectiveness when implemented with lightweight sparse retrievers such as BM25?* To address this gap, we conduct evaluations across multiple datasets using BM25 as the retriever, comparing two scenarios: (1) *Direct* incorporates the retrieved knowledge directly, and (2) *Ours* employs the proposed framework to filter the retrieved knowledge. For comparison, we adopt the BGE-based method as the baseline.

As illustrated in Figure 6, the overall performance of BM25 is significantly lower compared to the baseline, primarily due to its limited semantic matching capabilities. This limitation results in the retrieval of irrelevant or even harmful knowledge that may compromise model performance. Nevertheless, when our proposed framework is employed, we observe significant performance improvements, indicating its ability to eliminate harmful knowledge from different retrievers. These results confirm our framework's capability of knowledge filtering and cross-retriever adaptability.

5.3 Impact of Filtering Strategies on PPL

Our annotation strategy leverages changes in perplexity (PPL) to assess whether a given piece of knowledge is harmful. However, the effectiveness of this annotation strategy in optimizing the final model PPL remains uncertain. We therefore conduct experiments to investigate the changes in the final model PPL across three datasets using proposed filtering strategies. We employ the model without filtering as a baseline for comparison.

As illustrated in Figure 7, *Direct* exhibits limited ability to assess knowledge harmfulness when lacking task-specific guidances. While incorporating task demonstrations further reduces model PPL, the effectiveness of *In-Context* is highly dependent on the quality of the provided demonstrations. This limitation is particularly evident in the CoLA dataset, where the suboptimal quality of demonstrations resulted in a substantial PPL increase. *Proxy* achieves significant model PPL reduction but its performance isn't stable. *Ensemble* demonstrates the lowest PPL while maintaining consistent performance across all datasets, validating the robustness and effectiveness of our ensemble mechanism.

5.4 Effects of Different Knowledge for Prediction Fusion

To investigate whether the effectiveness of prediction fusion stems exclusively from its mechanism design, we conduct experiments to analyze the influence of knowledge quality on model performance. Specifically, we compare the performance variations before and after applying prediction fusion using random knowledge, with filtered knowledge serving as the baseline for comparison.

As shown in Figure 8, the introduction of random knowledge results in a substantial performance degradation of the model compared to the

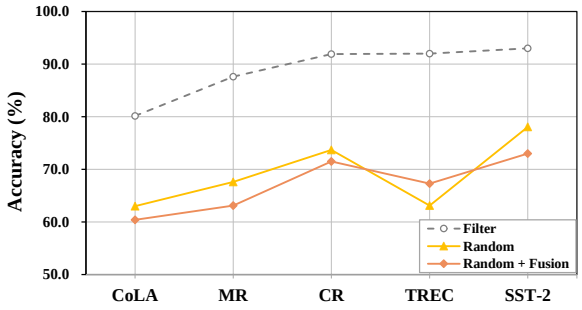

Figure 8: The performance of prediction fusion under different quality of knowledge.

baseline. We suppose that this decline is primarily caused by the introduction of irrelevant and harmful information present in random knowledge. When applying batch-wise prediction fusion to random knowledge, we observe a general performance decline rather than improvement across most datasets. This phenomenon suggests that considerable irrelevant information in random knowledge will compromise the fusion process. Furthermore, this also validates that the performance improvements achieved by prediction fusion are not only derived from its mechanism design but also depend on high-quality filtered knowledge. In general, the components within our framework are complementary, collectively contributing to significant improvements in model performance.

6 Related Works

6.1 Retrieval-Augmented Generation

Retrieval-Augmented methods have been widely applied since the era of pre-trained models (Liu, 2019), achieving significant progress in many NLP tasks such as dialogue generation (Wu et al., 2019) and question answering (Izacard and Grave, 2020; Wang et al., 2024). With the emergence of large language models, RAG has been further utilized to alleviate problems such as hallucinations in LLMs (Shao et al., 2023). The mainstream approach to solving generative tasks follows *Retrieve-then-Read* paradigm (Yoran et al., 2023). Recent research has primarily focused on improving important components in RAG systems: query augmenter (Tan et al., 2024), paragraph retriever (Shi et al., 2023b), and generator (Jin et al., 2024).

Since retrieved knowledge is not always helpful, especially when the internal knowledge of LLMs is sufficient to answer the question (Shi et al., 2023a). To mitigate this problem, mainstream approaches can be divided into two categories: 1) **Pre-retrieval** methods, which aim to determine whether retrieval is necessary for a given query: Wang et al. (2023c) proposes using a small classification model or the LLM itself to analyze queries in order to determine whether the LLM can rely on its own knowledge to provide an answer. Tan et al. (2024) proposes using a proxy model to first generate a pseudo-answer, which is then used to determine whether the LLM needs to retrieve external knowledge. 2) **Post-retrieval** methods, which focus on optimizing the retrieved knowledge in various ways: Since the generator is a black-box model, Yang et al. (2023) proposes training an adapter through reinforcement learning to compress the retrieved knowledge. Zhu et al. (2024) introduces the information bottleneck theory to optimize the noise filter from a comprehensive perspective, aiming to optimize the retrieved passages by simultaneously maximizing useful information and minimizing noise.

6.2 Discriminative Tasks

The current mainstream methods for discriminative tasks can be divided into the following categories: Tree-based (Khandagale et al., 2020; Wang et al., 2022; Bao et al., 2023), Graph-based (Ye et al., 2021; Xu et al., 2021), Embedding-based (Chen et al., 2020; Gweon and Schonlau, 2024) and Ensemble-based approaches (Zhao et al., 2022; Liu et al., 2022). Bao et al. (2023) proposes a novel opinion tree parsing model to extract all sentiment elements from opinion trees. Zhao et al. (2025) proposes an instruction augmentation approach to enhance the performance of emotion classification, which does not rely on labeled instances. Although significant progress has been made in these tasks, few studies have attempted to leverage external knowledge for completion.

Existing RAG methods struggle to accurately identify retrieval requirements, while compression techniques cannot completely discard harmful knowledge. Furthermore, the integration of external knowledge is often overlooked in discriminative task research. To address this gap, our framework aims to apply RAG to discriminative tasks and enhance performance from the perspective of knowledge filtering. Notably, our approach prevents the missing of potentially helpful information due to the misidentification of retrieval requirements. Furthermore, in scenarios where all retrieved knowledge is harmful, our filtering strategies can discard all knowledge and degenerate to standard non-retrieval generation.

7 Conclusion

In this study, we propose a novel framework to enhance the performance of RAG in discriminative tasks by introducing knowledge filtering and prediction fusion mechanisms. Specifically, our method first employs a perplexity-based annotation method to collect training data for subsequent procedures. Then, we propose four strategies to effectively filter out harmful knowledge. Finally, we integrate the filtered knowledge via batch-wise predictions to generate the final results. We conduct extensive experiments across multiple datasets, and the experimental results demonstrate that our framework significantly enhances model performance.

Limitations

The limitations of our work lie in below: The integration of additional filtering processes for the retrieved knowledge within our framework inevitably heightens the overall time complexity. While these processes are designed to enhance the quality and relevance of the knowledge used, the trade-off is a less efficient retrieval pipeline compared to simpler methodologies.

Acknowledgments

We would like to thank Prof. Zhongqing Wang for his helpful advice and discussion during this work. Also, we would like to express our gratitude to the five anonymous reviewers for their insightful comments on this paper. This research was supported by the National Natural Science Foundation of China (No. 62376178), and Project Funded by the Priority Academic Program Development of Jiangsu Higher Education Institutions.

References

Xiaoyi Bao, Xiaotong Jiang, Zhongqing Wang, Yue Zhang, and Guodong Zhou. 2023. Opinion tree parsing for aspect-based sentiment analysis. In *Findings of the Association for Computational Linguistics: ACL 2023*, pages 7971–7984.

Sam Boshar, Evan Trop, Bernardo P de Almeida, Liviu Copoiu, and Thomas Pierrot. 2024. Are genomic language models all you need? exploring genomic language models on protein downstream tasks. *Bioinformatics*, 40(9):btae529.

Boli Chen, Xin Huang, Lin Xiao, Zixin Cai, and Liping Jing. 2020. Hyperbolic interaction model for hierarchical multi-label classification. In *Proceedings of the AAAI conference on artificial intelligence*, volume 34, pages 7496–7503.

Daixuan Cheng, Shaohan Huang, Junyu Bi, Yuefeng Zhan, Jianfeng Liu, Yujing Wang, Hao Sun, Furu Wei, Denvy Deng, and Qi Zhang. 2023. Uprise: Universal prompt retrieval for improving zero-shot evaluation. *arXiv preprint arXiv:2303.08518*.

Xiaowen Ding, Bing Liu, and Philip S Yu. 2008. A holistic lexicon-based approach to opinion mining. In *Proceedings of the 2008 international conference on web search and data mining*, pages 231–240.

Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Amy Yang, Angela Fan, et al. 2024. The llama 3 herd of models. *arXiv preprint arXiv:2407.21783*.

Wenqi Fan, Yujuan Ding, Liangbo Ning, Shijie Wang, Hengyun Li, Dawei Yin, Tat-Seng Chua, and Qing Li. 2024. A survey on rag meeting llms: Towards retrieval-augmented large language models. In *Proceedings of the 30th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, pages 6491–6501.

Tianyu Gao, Adam Fisch, and Danqi Chen. 2020. Making pre-trained language models better few-shot learners. *arXiv preprint arXiv:2012.15723*.

Tianyu Gao, Xingcheng Yao, and Danqi Chen. 2021. Simcse: Simple contrastive learning of sentence embeddings. *arXiv preprint arXiv:2104.08821*.

Zhicheng Guo, Sijie Cheng, Yile Wang, Peng Li, and Yang Liu. 2023. Prompt-guided retrieval augmentation for non-knowledge-intensive tasks. *arXiv preprint arXiv:2305.17653*.

Hyukjun Gweon and Matthias Schonlau. 2024. Automated classification for open-ended questions with bert. *Journal of Survey Statistics and Methodology*, 12(2):493–504.

Md Arid Hasan, Shudipta Das, Afiyat Anjum, Firoj Alam, Anika Anjum, Avijit Sarker, and Sheak Rashed Haider Noori. 2024. Zero-and few-shot prompting with llms: A comparative study with fine-tuned models for bangla sentiment analysis. In *Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING 2024)*, pages 17808–17818.

Gautier Izacard and Edouard Grave. 2020. Leveraging passage retrieval with generative models for open domain question answering. *arXiv preprint arXiv:2007.01282*.

Gautier Izacard and Édouard Grave. 2021. Leveraging passage retrieval with generative models for open domain question answering. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*, pages 874–880.

Jiajie Jin, Yutao Zhu, Yujia Zhou, and Zhicheng Dou. 2024. Bider: Bridging knowledge inconsistency for efficient retrieval-augmented llms via key supporting evidence. *arXiv preprint arXiv:2402.12174*.

Jeff Johnson, Matthijs Douze, and Hervé Jégou. 2019. Billion-scale similarity search with gpus. *IEEE Transactions on Big Data*, 7(3):535–547.

Nikhil Kandpal, Haikang Deng, Adam Roberts, Eric Wallace, and Colin Raffel. 2023. Large language models struggle to learn long-tail knowledge. In *International Conference on Machine Learning*, pages 15696–15707. PMLR.

Sujay Khandagale, Han Xiao, and Rohit Babbar. 2020. Bonsai: diverse and shallow trees for extreme multi-label classification. *Machine Learning*, 109(11):2099–2119.

Xin Li and Dan Roth. 2002. Learning question classifiers. In *COLING 2002: The 19th International Conference on Computational Linguistics*.

Ming Liu, He Zhang, Yangjie Tian, Tianrui Zong, Borui Cai, Ruohua Xu, and Yunfeng Li. 2022. Overview of nlpcc2022 shared task 5 track 1: Multi-label classification for scientific literature. In *CCF International Conference on Natural Language Processing and Chinese Computing*, pages 320–327. Springer.

Yinhan Liu. 2019. Roberta: A robustly optimized bert pretraining approach. *arXiv preprint arXiv:1907.11692*, 364.

Yubo Ma, Yixin Cao, YongChing Hong, and Aixin Sun. 2023. Large language model is not a good few-shot information extractor, but a good reranker for hard samples! *arXiv preprint arXiv:2303.08559*.

OpenAI. 2024. Hello, gpt-4o.

Bo Pang and Lillian Lee. 2004. A sentimental education: Sentiment analysis using subjectivity summarization based on minimum cuts. *arXiv preprint cs/0409058*.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J Liu. 2020. Exploring the limits of transfer learning with a unified text-to-text transformer. *Journal of machine learning research*, 21(140):1–67.

Zhihong Shao, Yeyun Gong, Yelong Shen, Minlie Huang, Nan Duan, and Weizhu Chen. 2023. Enhancing retrieval-augmented large language models with iterative retrieval-generation synergy. *arXiv preprint arXiv:2305.15294*.

Freda Shi, Xinyun Chen, Kanishka Misra, Nathan Scales, David Dohan, Ed H Chi, Nathanael Schärli, and Denny Zhou. 2023a. Large language models can be easily distracted by irrelevant context. In *International Conference on Machine Learning*, pages 31210–31227. PMLR.

Weijia Shi, Sewon Min, Michihiro Yasunaga, Minjoon Seo, Rich James, Mike Lewis, Luke Zettlemoyer, and Wen-tau Yih. 2023b. Replug: Retrieval-augmented black-box language models. *arXiv preprint arXiv:2301.12652*.

Richard Socher, Alex Perelygin, Jean Wu, Jason Chuang, Christopher D Manning, Andrew Y Ng, and Christopher Potts. 2013. Recursive deep models for semantic compositionality over a sentiment treebank. In *Proceedings of the 2013 conference on empirical methods in natural language processing*, pages 1631–1642.

Jiejun Tan, Zhicheng Dou, Yutao Zhu, Peidong Guo, Kun Fang, and Ji-Rong Wen. 2024. Small models, big insights: Leveraging slim proxy models to decide when and what to retrieve for llms. *arXiv preprint arXiv:2402.12052*.

Neeraj Varshney, Man Luo, and Chitta Baral. 2022. Can open-domain qa reader utilize external knowledge efficiently like humans? *arXiv preprint arXiv:2211.12707*.

Haoyu Wang, Ruirui Li, Haoming Jiang, Jinjin Tian, Zhengyang Wang, Chen Luo, Xianfeng Tang, Monica Cheng, Tuo Zhao, and Jing Gao. 2024. Blendfilter: Advancing retrieval-augmented large language models via query generation blending and knowledge filtering. *arXiv preprint arXiv:2402.11129*.

Jiaan Wang, Yunlong Liang, Fandong Meng, Beiqi Zou, Zhixu Li, Jianfeng Qu, and Jie Zhou. 2023a. Zero-shot cross-lingual summarization via large language models. *arXiv preprint arXiv:2302.14229*.

Xiao Wang, Weikang Zhou, Can Zu, Han Xia, Tianze Chen, Yuansen Zhang, Rui Zheng, Junjie Ye, Qi Zhang, Tao Gui, et al. 2023b. Instructuie: Multi-task instruction tuning for unified information extraction. *arXiv preprint arXiv:2304.08085*.

Yile Wang, Peng Li, Maosong Sun, and Yang Liu. 2023c. Self-knowledge guided retrieval augmentation for large language models. *arXiv preprint arXiv:2310.05002*.

Zihan Wang, Peiyi Wang, Tianyu Liu, Binghuai Lin, Yunbo Cao, Zhifang Sui, and Houfeng Wang. 2022. Hpt: Hierarchy-aware prompt tuning for hierarchical text classification. *arXiv preprint arXiv:2204.13413*.

A Warstadt. 2019. Neural network acceptability judgments. *arXiv preprint arXiv:1805.12471*.

Janyce Wiebe, Theresa Wilson, and Claire Cardie. 2005. Annotating expressions of opinions and emotions in language. *Language resources and evaluation*, 39:165–210.

Yu Wu, Furu Wei, Shaohan Huang, Yunli Wang, Zhoujun Li, and Ming Zhou. 2019. Response generation by context-aware prototype editing. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 7281–7288.

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

Fangyuan Xu, Weijia Shi, and Eunsol Choi. 2023. Recomp: Improving retrieval-augmented lms with compression and selective augmentation. *arXiv preprint arXiv:2310.04408*.

Linli Xu, Sijie Teng, Ruoyu Zhao, Junliang Guo, Chi Xiao, Deqiang Jiang, and Bo Ren. 2021. Hierarchical multi-label text classification with horizontal and vertical category correlations. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 2459–2468.

Haoyan Yang, Zhitao Li, Yong Zhang, Jianzong Wang, Ning Cheng, Ming Li, and Jing Xiao. 2023. Prca: Fitting black-box large language models for retrieval question answering via pluggable reward-driven contextual adapter. *arXiv preprint arXiv:2310.18347*.

Chenchen Ye, Linhai Zhang, Yulan He, Deyu Zhou, and Jie Wu. 2021. Beyond text: Incorporating metadata and label structure for multi-label document classification using heterogeneous graphs. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 3162–3171.

Ori Yoran, Tomer Wolfson, Ori Ram, and Jonathan Berant. 2023. Making retrieval-augmented language models robust to irrelevant context. *arXiv preprint arXiv:2310.01558*.

Susan Zhang, Stephen Roller, Naman Goyal, Mikel Artetxe, Moya Chen, Shuohui Chen, Christopher Dewan, Mona Diab, Xian Li, Xi Victoria Lin, et al. 2023. Opt: Open pre-trained transformer language models, 2022. *URL https://arxiv. org/abs/2205.01068*, 3:19–0.

Zhuosheng Zhang, Aston Zhang, Mu Li, and Alex Smola. 2022. Automatic chain of thought prompting in large language models. *arXiv preprint arXiv:2210.03493*.

Xiuhao Zhao, Zhao Li, Xianming Zhang, Jibin Wang, Tong Chen, Zhengyu Ju, Canjun Wang, Chao Zhang, and Yiming Zhan. 2022. An interactive fusion model for hierarchical multi-label text classification. In *CCF International Conference on Natural Language Processing and Chinese Computing*, pages 168–178. Springer.

Yang Zhao, Masayasu Muraoka, Issei Yoshida, Bishwaranjan Bhattacharjee, and Hiroshi Kanayama. 2025. A simple-yet-efficient instruction augmentation method for zero-shot sentiment classification. In *Proceedings of the 31st International Conference on Computational Linguistics*, pages 1585–1599.

Chunting Zhou, Graham Neubig, Jiatao Gu, Mona Diab, Francisco Guzmán, Luke Zettlemoyer, and Marjan Ghazvininejad. 2021. Detecting hallucinated content in conditional neural sequence generation. In *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, pages 1393–1404.

Kun Zhu, Xiaocheng Feng, Xiyuan Du, Yuxuan Gu, Weijiang Yu, Haotian Wang, Qianglong Chen, Zheng Chu, Jingchang Chen, and Bing Qin. 2024. An information bottleneck perspective for effective noise filtering on retrieval-augmented generation. *arXiv preprint arXiv:2406.01549*.

Datasets	LR	Lora_alpha	Lora_rank
CoLA	3e-4	16	8
TREC	3e-4	16	8
SST-2	5e-5	16	8
SST-5	5e-5	16	8
MR	5e-5	16	8
MPQA	3e-4	16	8
CR	5e-5	16	8

Table 5: The training parameter of different datasets.

A Implementation

For the external knowledge base, we employ the Wiki-1M (Gao et al., 2021) where each piece of knowledge is presented as individual sentences rather than paragraph-level chunks. For the retrieval stage, we utilize the BGE model[1] as our dense retriever and retain the top-30 most relevant knowledge. During the stage of knowledge filtering, we constrain the amount of knowledge utilized for subsequent augmentation to a maximum of eight. It is noteworthy that the quantity of filtered knowledge may potentially be zero (indicating all retrieved knowledge is harmful), in which case our approach degenerates into a non-retrieval generation.

Our approach adopts LLaMA3-8B [2] as the backbone of our framework. We employ parameter-efficient adaptation through Low-Rank Adaptation (LoRA) to fine-tune the model for each dataset and the detailed training hyperparameters are documented in Table 5. During inference, we employ greedy decoding for the reasons described in Appendix D. All experiments are conducted on an NVIDIA Tesla A100 64G GPU.

For our experiments with GPT-4o, we evaluate the model on our dataset using the OpenAI Fine-tuning API[3]. The prompt "*Read knowledge and the*

[1]https://huggingface.co/BAAI/bge-large-en-v1.5

[2]https://huggingface.co/meta-llama/Llama-3.1-8B-Instruct

[3]https://platform.openai.com/finetune

Datasets	Category	Input	Candidate Labels
CoLA	Linguistic Acceptability	Sentence	Incorrect; Correct
TREC	Question Classification	Sentence	Desc.; Entity; Abbr.; Person; Location; Quantity
SST-2	Sentiment Analysis	Sentence	Negative; Positive
SST-5	Sentiment Analysis	Sentence	Terrible; Bad; Neutral; Good; Excellent
MR	Sentiment Analysis	Sentence	Negative; Positive
MPQA	Sentiment Analysis	Sentence	Negative; Positive
CR	Sentiment Analysis	Sentence	Negative; Positive

Table 6: The description of datasets. **Desc** is short for Description and **Abbr** is short for Abbreviation.

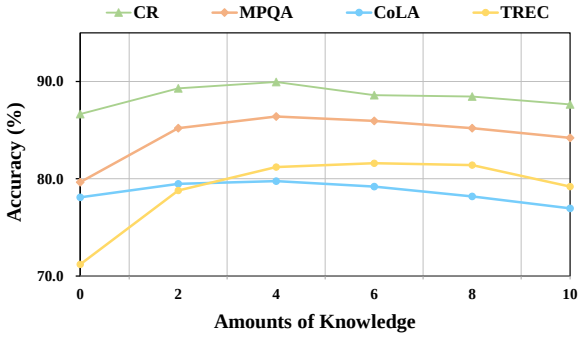

Figure 9: The performance of different amounts of knowledge.

sentence, predict the sentiment of the sentence as < *Candidate Labels* > " is chosen empirically for its strong performance. During the generation stage, the temperature is set to 0.0, ensuring consistency with our approach.

B Impact of Different Amounts of Knowledge

The incorporation of external knowledge has proven beneficial for discriminative tasks. However, an important question arises: *Does increasing the amounts of knowledge consistently enhance model performance?* To address this gap, we investigate the effect of varying the amounts of knowledge on model performance across four datasets.

As illustrated in Figure 9, the performance of the model exhibits the expected upward trend as more retrieved knowledge is gradually incorporated. Nevertheless, when the knowledge quantity surpasses an optimal threshold, further increases in knowledge result not in continued improvement but rather in performance degradation. We hypothesize that this decline may stem from the excessive amounts of knowledge imposing a cognitive burden on the model. This heightened burden could render the model more vulnerable to the misleading influence of certain pieces of knowledge, thereby adversely impacting its overall performance.

Methods	CR	MPQA	CoLA	TREC	Avg.
ICL	86.7	79.7	78.1	71.2	78.9
SFT	89.7	87.7	77.9	90.0	86.3

Table 7: The performance of different optimization strategies.

C Impact of Different Optimization Strategies

We investigate the effectiveness of different optimization strategies for addressing the discriminative task. Since the proposed framework is based on LLMs, we focus on evaluating two widely adopted methods across four datasets: In-Context Learning (ICL) and Supervised Fine-Tuning (SFT). For the ICL method, we provide three task demonstrations for each category, while the detailed information of the training data for SFT is provided in Table 2.

As shown in Figure 7, the experimental results reveal that SFT consistently outperforms ICL in overall performance, with particularly notable improvements observed on the TREC dataset, which contains a larger number of categories. This can be attributed to two primary factors: First, the effectiveness of ICL is highly contingent on the quality of task demonstrations, which often results in suboptimal performance. Second, SFT benefits from removing the limitation of context length, enabling it to utilize richer training data. Consequently, SFT demonstrates superior performance.

D Impact of Different Decoding Strategies

While temperature-based sampling methods are widely employed in generative tasks to enhance output diversity, their applicability to discriminative tasks (e.g., text classification) remains underexplored. To investigate this, we conduct experiments

Methods	CR	MPQA	CoLA	TREC	Avg.
Greedy	86.7	79.7	78.1	71.2	78.9
T=0.5	86.0	78.6	78.4	70.6	78.4
T=0.7	85.4	78.3	76.2	70.8	77.7
T=1.0	81.7	77.2	75.8	68.4	75.8
T=1.5	74.1	72.6	67.9	62.0	69.1

Table 8: The performance of the model under different temperatures.

across four datasets using five distinct decoding configurations. Greedy sampling, which selects the token with the highest probability, serves as a baseline and corresponds to a temperature value of 0.0. In addition, we evaluate the model's performance under four different temperature settings $T \in \{0.5, 0.7, 1.0, 1.5\}$ to analyze its sensitivity to temperature adjustments.

As evidenced by Table 8, model performance demonstrates consistent deterioration with rising temperature values. Through manual analysis of token-level output probabilities across selected samples, we observe that higher temperatures result in the model generating non-candidate labels, which substantially degrades performance. Accordingly, for discriminative tasks, we recommend adopting the greedy sampling strategy to ensure the accuracy and consistency of the model's outputs.

E Prompts for Different Datasets

Here we present the MCQ templates (Figure 10, 11, 12) employed in the LLMs-based knowledge filtering strategies described in Section 3.2.

Multi-Choice Question Prompt

(Demonstrations if available)
Instruction: Given the knowledge and sentence, choose if the knowledge is harmful for predicting the sentiment polarity of the sentence as A or B.
Knowledge: {Retrieved Knowledge}
Sentence: {Query}
Candidate Choices: (A) Harmful (B) Harmless
Answer: {Correct Answer}
…

(Test Sample)
Instruction: Given the knowledge and sentence, choose if the knowledge is harmful for predicting the sentiment polarity of the sentence as A or B.
Knowledge: {Retrieved Knowledge}
Sentence: {Query}
Candidate Choices: (A) Harmful (B) Harmless
Answer:

Figure 10: Multi-Choice Question (MCQ) template for MR, CR, MPQA, SST-2, SST-5.

Multi-Choice Question Prompt

(Demonstrations if available)
Instruction: Given the knowledge and question, choose if the knowledge is harmful for predicting the question category as A or B.
Knowledge: {Retrieved Knowledge}
Sentence: {Query}
Candidate Choices: (A) Harmful (B) Harmless
Answer: {Correct Answer}
…

(Test Sample)
Instruction: Given the knowledge and question, choose if the knowledge is harmful for predicting the question category as A or B.
Knowledge: {Retrieved Knowledge}
Sentence: {Query}
Candidate Choices: (A) Harmful (B) Harmless
Answer:

Figure 11: Multi-Choice Question (MCQ) template for TREC.

Multi-Choice Question Prompt

(Demonstrations if available)
Instruction: Given the knowledge and sentence, choose if the knowledge is harmful for predicting the grammatical correctness of the sentence as A or B.
Knowledge: {Retrieved Knowledge}
Sentence: {Query}
Candidate Choices: (A) Harmful (B) Harmless
Answer: {Correct Answer}
…

(Test Sample)
Instruction: Given the knowledge and sentence, choose if the knowledge is harmful for predicting the grammatical correctness of the sentence as A or B.
Knowledge: {Retrieved Knowledge}
Sentence: {Query}
Candidate Choices: (A) Harmful (B) Harmless
Answer:

Figure 12: Multi-Choice Question (MCQ) template for CoLA.