Seeking Rational Demonstrations for Large Language Models: A Domain Generalization Approach to Unsupervised Cross-Domain Keyphrase Generation

Guangzhen Zhao†, **Yu Yao**‡, **Dechang Kong**†, **Zhenjiang Dong**†*
† School of Computer Science, Nanjing University of Posts and Telecommunications, China
‡ School of Computer Science and Artifical Intelligence, Chaohu University, China
{zhaogz, dongzhenjiang, b21030713}@njupt.edu.cn, 054096@chu.edu.cn

Abstract

Unsupervised cross-domain keyphrase generation (KPG) is crucial in real-world natural language processing scenarios. However, the accuracy of up-to-date approaches is limited by the distribution shift between source and target domain, which stems from the cross-domain field. Large language models (LLMs) offer potential for the cross-domain keyphrase generation tasks due to their strong generalization abilities, facilitated by providing demonstrations relevant to the target task. Nevertheless, it is often difficult to obtain labeled samples from the target domain. To address this challenge, this paper aims to seek rational demonstrations from the source domain, thereby improving the LLMs' ability in the unsupervised cross-domain keyphrase generation setting. Specifically, we design a novel domain-aware retrieval model on the source domain. Guided by insights from domain generalization theory, we introduce two generalization terms, one for cross-domain relevance and another for each domain consistency to better support retrieval of rational demonstrations. By the retrieved source-domain demonstrations and distance-based relevant score, the proposed approach achieves optimal accuracy. Comprehensive experiments on widely used cross-domain KPG benchmarks demonstrate our approach's state-of-the-art performance and effectiveness.

1 Introduction

Keyphrase Generation (KPG) is critical in identifying discriminative information (Meng et al., 2017; Shao et al., 2024; Boudin and Aizawa, 2024). As a fundamental natural language generation (NLG) task, keyphrase generation facilitates a wide variety of downstream applications, including document clustering (Hulth and Megyesi, 2006; Chiu et al., 2020), information retrieval (Ushiku et al., 2017; Boudin et al., 2020) and text summarization (Wang and Cardie, 2013). However, in real world scenarios, e.g. news articles (Gallina et al., 2019), web pages (Xiong et al., 2019), technical question-answer website (Yuan et al., 2020), obtaining labels can be challenging due to the reliance on domain expertise, or even infeasible because of the strict privacy constraints. Deploying keyphrase generation model in these scenarios often occurs in cross-domain KPG setting, e,g., first training KPG models on source tasks and then generalizing to target tasks (Gulrajani and Lopez-Paz; Wiles et al.). What is notable is that distribution shift is an inevitable phenomenon in cross-domain KPG.

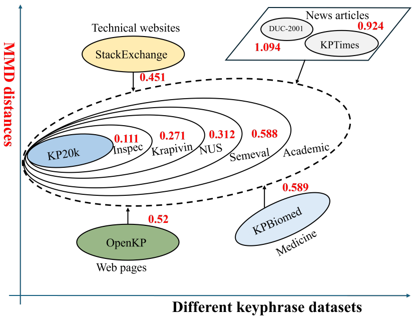

Figure 1: we treat the KP20k train dataset as the source domain, and using MMD to measure the degree of the distribution shift between different datasets.

As depicted in Figure 1, we treat the KP20k train dataset (Meng et al., 2017) as the source domain, and adopt Maximum Mean Discrepancy (MMD) (Kim et al., 2016) to measure the degree of the distribution shift. Higher MMD values between each two datasets indicate increased requirements for the model's generalization capabilities. For example, compared to academic domain datasets, such as Inspec (Hulth, 2003), Krapivin (Krapivin and Marchese, 2009), and NUS (Nguyen and Kan, 2007), the MMD distance is significantly larger for other domain datasets (e.g., OpenKP (Xiong et al., 2019), KPBiomed (Houbre et al., 2022)). This makes generalizing the KPG model in cross-domain setting more challenging than in standard generation,

[*] Corresponding author.

Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers), pages 414–424
July 27 - August 1, 2025 ©2025 Association for Computational Linguistics

where a robust decision boundary is sufficient. An effective strategy to alleviate distribution shift is the use of large language models (LLMs), where labeled samples are used as prompts to facilitate the generation of target domain keyphrases. However, even annotating a limited number of samples can be prohibitively expensive, and often impractical due to the need for expert annotators (Chau et al., 2020; Boudin and Aizawa, 2024). How to effectively leverage existing labeled source domain samples to prompt LLMs in unsupervised cross-domain KPG is a crucial problem that deserves in-depth exploration. In this work, we propose a seeking rational demonstration (SRD) approach for enhancing LLMs' ability in unsupervised cross-domain KPG. The SRD approach is designed based on DPR (Dense Passage Retrieval) model (Karpukhin et al., 2020). Specifically, we randomly select 20% samples from KP20k train datasets and target test dataset as the query set, with the rest as candidate set. We then construct positive and negative samples by computing the relevance of each query-candidate sample pair. However, there is no guarantee of accuracy when mapping the target query samples to the rational KP20k train samples during test time when facing distribution shift. As such, we introduce the domain projection simulation (DPS) with two new losses based on domain generalization theory to alleviate distribution shift between training and test phrases. The first loss is a domain projection loss based on the MMD distance, which helps learn more domain-invariant features and mitigates distribution shift. The second one is a domain characteristic loss that encourages the learnable train distribution to be orthogonal in the representation space, thus increasing the diversity and preventing domain characteristic vanished.

Our contributions are three-fold:

- We quantify the distribution shift phenomenon when applying LLMs to unsupervised cross-domain KPG. To the best of our knowledge, this is the first work that introduces domain generalization with LLMs in KPG.
- We investigate both theoretically and empirically how domain generalization technique can help with distribution projection, and thus alleviating distribution shift.
- Experimental results demonstrate that our SRD approach performs better than up-to-date baselines on widely used cross-domain KPG test datasets. The processed test datasets, codes and experimental results will be upload at https://github.com/chrischowfy/SRD/.

2 Realted Work

Current keyphrase generation tasks primarily focus on academic article domain, encompassing both supervision (Meng et al., 2017; Ye et al., 2021; Shao et al., 2024; Kang and Shin, 2024) and unsupervison (Shen et al., 2022; Boudin and Aizawa, 2024) fashions. AutoKeyGen (Shen et al., 2022) is the first work to explore the unsupervised keyphrase generation task, which trains a sequence-to-sequence model to extract present keyphrases and synthesize absent keyphrases using a phrase corpus. After that, UOKG (Do et al., 2023) extends AutoKeyGen in open-domain setting, which generates keyphrases that represent the core concept of the source text. Another related work is domain-adaptive keyphrase generation (Meng et al., 2023), however, it requires a few labeled samples from the target domain for fine-tuning. Another related direction is keyphrase generation in low-resource settings (Wu et al., 2022; Garg et al., 2023). Existing researches often employ target domain data pretraining (Wu et al., 2022) and data augmentation (Garg et al., 2023) to enhance keyphrases generation capabilities in low-resource contexts. However, in the era of large language models, considering the challenges of fine-tuning LLMs and computational expenses, there is an urgent need for new research approaches to address the problem of unsupervised cross-domain keyphrase generation.

Recently, LLMs perform well in unsupervised keyphrase generation (MartÍınez-Cruz et al., 2024). Notably, when provided with an in-context prompt that contains a few relevant samples from the target domain, LLMs can further enhance their keyphrase generation performance in target domain (Jiang et al., 2024). However, in real-world scenarios, acquiring labeled target samples is non-trivial. This motivates our work: how to effectively prompt LLMs using labeled source domain samples to improve keyphrase generation across different domains.

3 Methodology

3.1 Problem Settings

Formally, the unsupervised cross-domain KPG task is a text generation problem conditioned on both

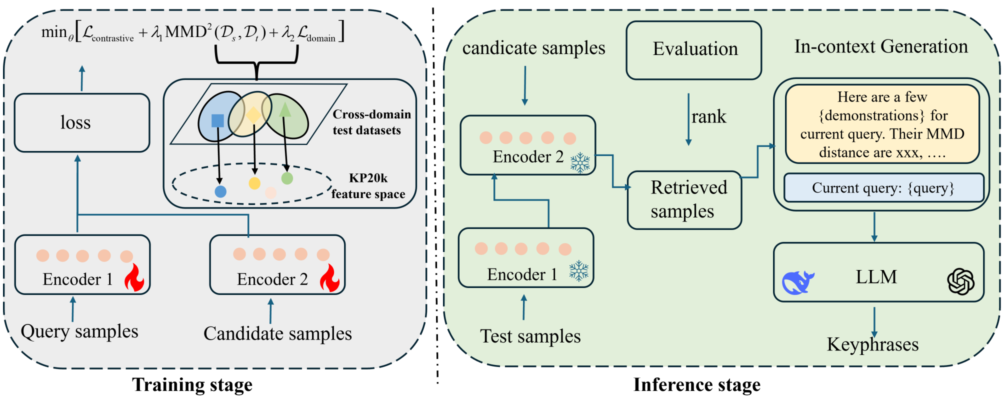
$\min_\theta[\mathcal{L}_{\text{contrastive}} + \lambda_1 \text{MMD}^2(\mathcal{D}_s, \mathcal{D}_t) + \lambda_2 \mathcal{L}_{\text{domain}}]$
loss
Cross-domain test datasets
KP20k feature space
Encoder 1
Encoder 2
Query samples
Candidate samples
Training stage
candicate samples
Evaluation
In-context Generation
rank
Encoder 2
Retrieved samples
Here are a few {demonstrations} for current query. Their MMD distance are xxx, ….
Current query: {query}
Encoder 1
LLM
Test samples
Keyphrases
Inference stage

Figure 2: Overview of the seeking rational demonstrations (SRD) approach for unsupervised cross-domain KPG.

the target domain input and a set of source domain demonstrations:

$$\hat{\mathbf{y}} = \arg\max_{\mathbf{y}} P(\mathbf{y} \mid x_t, \text{Retrieve}(x_t, \mathcal{S}), \Theta), \quad (1)$$

where $x_t \in \mathcal{D}_t$ is the unlabeled target-domain input, $\text{Retrieve}(\cdot,\cdot)$ denotes a retrieval function that selects relevant demonstrations from the labeled source-domain samples $\mathcal{S}$, and Θ denotes the LLMs' parameters. Note that no labeled target-domain data is available, the generalization capacity of the model is crucially depends on the quality of its retrieved source-domain samples under distribution shift.

3.2 Basic Retrieve Module

As shown in Figure 2, to efficiently retrieve rational demonstrations for unsupervised cross-domain KPG, we first build a basic dual-encoder retrieval module based on the DPR model (Karpukhin et al., 2020). The module includes two encoders, which can map both query text and candidate text into vector representations.

Constructing Positive & Negative Pairs. Since the KP20k train dataset does not explicitly provide positive or negative sample pairs, we manually construct them by analyzing the keyphrases of each sample. Specifically, let $\mathbf{k}_x$ and $\mathbf{k}_s$ be the keyphrases of samples x and s, respectively. The relevance score is defined as:

$$\text{Rel}(x, s) = \alpha \cdot \text{sim}_{\text{embed}}(\mathbf{k}_x, \mathbf{k}_s) + (1-\alpha) \cdot \text{Jaccard}(\mathbf{k}_x, \mathbf{k}_s), \quad (2)$$

where $\text{sim}_{\text{embed}}(\mathbf{k}_x, \mathbf{k}_s)$ denotes the semantic similarity and $\text{Jaccard}(\mathbf{k}_x, \mathbf{k}_s)$ is the Jaccard similarity. If $\text{Rel}(x, s)$ exceeds a global threshold γ_1, we classify (x, s) as a positive (matched) pair, and otherwise treat it as a negative (unmatched) pair.

Self-supervised Demonstrations Evaluator. We design a self-supervised evaluator utilizing the KP20k train dataset to assess the efficacy of retrieved demonstrations. We divide KP20k into two disjoint subsets: the pseudo-source domain $\mathcal{D}_{\text{ps}}$ and pseudo-target domain $\mathcal{D}_{\text{pt}}$. The former is used to simulate the source domain for retrieval purposes, and the latter is used to mimic the unlabeled target domain scenario. We then train a regression model to learn a function mapping the triplet $(\mathbf{x}, \mathbf{e}, \mathbf{y}^{\text{gen}})$ to a quality score $\hat{s}$, where $\mathbf{x}$ represents the abstract text in the pseudo-target domain, $\mathbf{e}$ indicates the retrieved examples, and $\mathbf{y}^{\text{gen}}$ are the keyphrases generated by a LLM. Let $\mathbf{y}^{\text{ref}}$ denote the ground-truth keyphrases for $\mathbf{x}$ in the pseudo-target domain. The quality score s of $\mathbf{y}^{\text{gen}}$ is computed by the Eq.2, i.e., $s = \text{Rel}(\mathbf{y}^{\text{gen}}, \mathbf{y}^{\text{ref}})$. More details can be found in Appendix A.1 Implementation Details.

3.3 The Risk of Distribution Shift

We first give the distribution shift measured by $\mathcal{H}$-divergence (Ben-David et al., 2010):

$$d_{\mathcal{H}}(\mathcal{D}_s, \mathcal{D}_t) = 2 \sup_{h \in \mathcal{H}} | \Pr_{x\sim\mathcal{D}_s}[h(x) = 1] - \Pr_{x\sim\mathcal{D}_t}[h(x) = 1]|, \quad (3)$$

where classifier $h : \mathcal{X} \to \{0, 1\}$ is a labeling function on each domain sample. Next, the ideal target domain $\bar{\mathcal{D}}_t$ is assumed that lies in the source domain convex hull Λ_s (Albuquerque et al., 2019). Under this assumption, the risk $\epsilon_t(h)$ on the target domain $\mathcal{D}_t$ is upper-bounded (Albuquerque et al., 2019; Chen et al., 2019) by:

$$\epsilon_t(h) \leq \sum_{i=1}^{K} \eta_i \epsilon_s^i(h) + \gamma + \zeta. \quad (4)$$

The first term is the risks over source domains, which can be minimized by empirical risk minimization. The second term γ is the $\mathcal{H}$-divergence between the ideal target $\overline{\mathcal{D}}_t$ and the real target domain D_t. The third term could be disregarded as it is negligibly small. In this paper, we focus on the first and second terms to alleviate distribution shift in unsupervised cross-domain KPG.

3.4 Distribution Projection Simulation

Since there is distribution shift existed between source and target domain, we attempt to mitigate the ensuing cross-domain risk by domain generalization theory. Specifically, we employ MMD (Kim et al., 2016) as a kernel-based criterion that directly measures the similarly between the source and target samples in its representation space. By guiding the encoder to produce more domain-invariant features, we reduce adverse effects of distribution shift and thus improve generalization in unsupervised cross-domain KPG. In each training iteration, we select a mini-batch of labeled source domain samples $\{x_i^s, y_i^s\}_{i=1}^m \subset \mathcal{D}_s$ and unlabeled target domain samples $\{x_j^t\}_{j=1}^n \subset \mathcal{D}_t$. Using the kernel function $k(\cdot, \cdot)$, the squared MMD is computed as:

$$\mathrm{MMD}^2(\mathcal{D}_s, \mathcal{D}_t) = \frac{1}{m(m-1)} \sum_{\substack{i,j=1 \\ i \neq j}}^{m} k\big(f(x_i^s), f(x_j^s)\big) + \frac{1}{n(n-1)} \sum_{\substack{i,j=1 \\ i \neq j}}^{n} k\big(f(x_i^t), f(x_j^t)\big) - \frac{2}{mn} \sum_{i=1}^{m} \sum_{j=1}^{n} k\big(f(x_i^s), f(x_j^t)\big), \tag{5}$$

where $f(\cdot)$ is the encoder's mapping from input text to a feature vector, $k(\cdot, \cdot)$ is the radial basis function kernel with $k_{i,j} = k(x_i, x_j) = \exp(-\gamma \|x_i - x_j\|)$, γ is a hyperparameter (Kim et al., 2016). MMD imposes a domain-invariance constraint on the encoders' representations. In this way, the features between the source and target domain samples align more closely, which are crucial for accurate keyphrase generation under distribution shift. However, a naive reduction of MMD operates on mean embedding feature of each sample, which may be insufficient to characterize sample mapping (Vayer and Gribonval, 2023). Hence, we also incorporate domain-specific features into the encoders that retain critical domain cues.

Preserving Domain Characteristics via Orthogonality. To further impose domain features into the encoders, we introduce two sets of mean and variance vectors:$\{\mu_b^s, \sigma_b^s\}$ for the source domain samples, and$\{\mu_b^t, \sigma_b^t\}$ for the target domains samples. We design a domain (orthogonality) loss to enforce these vectors to remain domain-specific:

$$\mathcal{L}_{\text{domain}} = \left| \frac{\mu_b^s}{\|\mu_b^s\|} \cdot \frac{\mu_b^t}{\|\mu_b^t\|} \right| + \left| \frac{\sigma_b^s}{\|\sigma_b^s\|} \cdot \frac{\sigma_b^t}{\|\sigma_b^t\|} \right|, \tag{6}$$

where $\|\cdot\|$ is the vector norm, and $\cdot$ is the dot product. By forcing the normalized means and variances from the source and target to be approximately orthogonal, we ensure that while we reduce overall distribution divergence, each domain maintains its own "style" (i.e., domain-specific signals).

3.5 Training

We train the dual-encoder retriever by combining traditional contrastive loss (Karpukhin et al., 2020) with the standard MMD alignment. The overall objective is to minimize:

$$\min_{\theta} \left[\mathcal{L}_{\text{contrastive}} + \lambda_1 \, \mathrm{MMD}^2(\mathcal{D}_s, \mathcal{D}_t) + \lambda_2 \, \mathcal{L}_{\text{domain}} \right], \tag{7}$$

where θ is the parameter of the retriever, and λ_1 and λ_2 are hyperparameters balancing the three terms. By jointly minimizing the contrastive loss and domain relevant loss, we reduce the distribution shift (thus addressing the second risk term in Eq. (4)) while preserving crucial domain characteristics in both source and target domains.

4 Experiments

4.1 Experimental Setup

Datasets We conduct experiments on five public domain KPG datasets, namely DUC-2001, KPTimes, OpenKP, StackExchange and KPBiomed. The detailed statistic is depicted in Table 1.

Datasets	Type	Test docs	#Kps/doc	%Absent	Api_llm
StackExchange (Yuan et al., 2020)	Technology	2000	2.65	48.78	300
DUC-2001 (Wan and Xiao, 2008)	News	308	8.06	2.7	300
KPTimes (Gallina et al., 2020)	News	2000	5.07	54.2	300
OpenKP (Xiong et al., 2019)	Web	2000	2.31	11.62	300
KPBiomed (Houbre et al., 2022)	Medicine	2000	5.38	38.74	300

Table 1: Statistic of cross-domain test datasets.

Baselines and Metrics We compare our approach with AutoKeyGen (Shen et al., 2022) and UOKG (Do et al., 2023), which are the only two standard baselines on unsupervised cross-domain keyphrase generation. Furthermore, we reimplement CopyRNN (Meng et al., 2017), One2set (Ye et al., 2021) and DeepKPG (Wu et al., 2023) as supervised baselines. The subscript in Table 2 indicates the standard deviation (e.g., 9.4_7 denotes 9.40±0.7).We also adopt several LLM backbones to evaluate our approach. Following UOKG, we use the F1@K and Recall@K as evaluation metrics, where K is the number of predicted keyphrases to be considered. We only use KP20k to seek rational demonstrations.

Present keyphrase generation											
Methods	StackExchange		DUC-2001		KPTimes		OpenKP		KPBiomed		Average
	F1@3	F1@5	F1@3	F1@5	F1@3	F1@5	F1@3	F1@5	F1@3	F1@5	
AutoKeyGen	13.7_{6}	14.6_{2}	6.6_{5}	9.1_{1}	11.1_{2}	11.5_{1}	8.9_{5}	9.1_{2}	12.9_{1}	15.2_{4}	13.70
UOKG	16.2_{1}	$\underline{18.3}_{4}$	16.0_{7}	18.7_{1}	12.2_{7}	13.1_{7}	13.3_{5}	11.7_{7}	14.4_{3}	14.6_{4}	14.85
CopyRNN	$\mathbf{24.9}_{2}$	$\mathbf{22.6}_{3}$	7.3_{1}	8.7_{2}	17.3_{5}	17.4_{4}	10.9_{2}	10.3_{1}	21.7_{3}	22.8_{2}	16.39
One2Set	18.2_{3}	15.4_{4}	7.9_{2}	9.1_{7}	10.8_{2}	6.4_{3}	6.5_{3}	6.1_{6}	20.7_{7}	20.9_{2}	12.20
DeepKPG	16.3_{1}	11.4_{6}	13.1_{2}	11.1_{3}	15.7_{5}	11.5_{3}	16.1_{7}	11.4_{2}	30.8_{4}	24.7_{5}	16.21
Qwen2.5-14b	1.8_{1}	1.2_{7}	8.4_{5}	6.8_{3}	4.5_{2}	3.1_{6}	7.6_{1}	5.4_{5}	5.4_{4}	4.1_{2}	4.83
Llama3.3-70b	5.0_{6}	3.4_{4}	21.3_{5}	19.1_{5}	13.1_{4}	9.8_{7}	19.9_{7}	14.5_{2}	20.9_{6}	19.4_{2}	14.64
ChatGPT-3.5	5.4_{4}	4.1_{1}	21.5_{7}	21.7_{3}	12.5_{4}	9.3_{1}	19.1_{3}	13.7_{6}	18.0_{2}	16.8_{5}	14.21
GPT4o	9.9_{1}	7.2_{1}	26.6_{4}	27.2_{3}	20.1_{1}	16.4_{2}	23.9_{3}	17.2_{4}	29.3_{2}	26.1_{3}	20.39
DeepSeek	5.4_{3}	3.7_{4}	21.6_{3}	21.9_{5}	11.0_{2}	8.5_{3}	16.5_{2}	12.2_{1}	18.4_{2}	17.2_{3}	13.64
Ours(Qwen2.5-14b)	17.8_{3}	14.3_{7}	25.2_{3}	26.1_{5}	14.2_{6}	10.4_{5}	22.6_{6}	20.1_{2}	25.8_{7}	24.6_{2}	20.11
Ours(ChatGPT-3.5)	16.8_{2}	12.1_{5}	24.5_{1}	25.1_{2}	16.9_{5}	14.6_{6}	24.7_{4}	18.6_{6}	27.4_{1}	25.1_{7}	20.58
Ours(GPT4o)	$\underline{23.1}_{4}$	16.1_{3}	$\underline{27.2}_{2}$	$\underline{28.8}_{4}$	$\mathbf{28.9}_{1}$	$\mathbf{23.6}_{1}$	$\mathbf{29.2}_{4}$	$\mathbf{22.4}_{2}$	$\mathbf{33.0}_{3}$	$\mathbf{32.7}_{3}$	**26.50**
Ours(DeepSeek)	21.2_{2}	14.9_{4}	$\mathbf{29.1}_{1}$	$\mathbf{29.0}_{1}$	$\underline{27.7}_{2}$	$\underline{22.7}_{3}$	$\underline{27.1}_{4}$	$\underline{19.6}_{4}$	$\underline{31.3}_{1}$	$\underline{30.0}_{3}$	$\underline{25.26}$
Absent keyphrase generation											
Methods	R5	R10	R5	R10	R5	R10	R5	R10	R5	R10	Average
AutoKeyGen	0.9_{5}	1.0_{2}	–	–	0.2_{1}	0.2_{1}	0.1_{3}	0.3_{1}	0.4_{2}	0.8_{2}	0.49
UOKG	1.9_{2}	2.8_{5}	–	–	1.1_{4}	1.1_{3}	0.3_{2}	0.5_{3}	1.4_{2}	2.2_{5}	1.41
CopyRNN	2.5_{2}	3.7_{4}	–	–	0.6_{4}	0.8_{1}	0.1_{4}	0.1_{4}	0.8_{3}	1.2_{3}	1.22
One2Set	1.1_{1}	1.5_{5}	–	–	0.3_{5}	0.3_{1}	0.1_{5}	0.2_{3}	0.8_{2}	0.8_{5}	0.64
DeepKPG	2.2_{1}	2.3_{1}	–	–	1.3_{5}	1.4_{5}	2.3_{1}	2.4_{2}	2.9_{4}	3.0_{3}	2.23
Qwen2.5-14b	0.2_{2}	0.2_{1}	–	–	0.5_{2}	0.7_{4}	3.1_{3}	3.0_{3}	0.7_{4}	0.7_{5}	1.14
Llama3.3-70b	2.9_{4}	3.2_{1}	–	–	2.3_{1}	2.3_{1}	7.6_{3}	7.8_{2}	3.2_{2}	3.5_{1}	4.10
ChatGPT-3.5	2.3_{2}	3.0_{4}	–	–	4.1_{6}	4.2_{4}	9.9_{6}	9.4_{4}	1.8_{2}	1.8_{6}	4.56
GPT4o	6.5_{3}	7.5_{4}	–	–	7.7_{1}	8.1_{2}	9.8_{2}	10.6_{3}	4.2_{1}	4.3_{3}	7.34
DeepSeek	3.1_{1}	3.2_{3}	–	–	5.8_{4}	6.0_{2}	9.8_{4}	9.7_{1}	2.9_{1}	2.9_{2}	5.42
Ours(Qwen2.5-14b)	6.7_{2}	6.3_{4}	–	–	2.1_{4}	1.9_{3}	8.4_{2}	8.9_{3}	4.0_{3}	3.9_{4}	5.28
Ours(ChatGPT-3.5)	4.9_{4}	5.2_{1}	–	–	6.8_{2}	7.2_{2}	9.9_{2}	12.4_{3}	3.9_{2}	4.7_{2}	6.88
Ours(GPT4o)	$\mathbf{8.8}_{4}$	$\underline{9.1}_{4}$	–	–	$\mathbf{12.3}_{4}$	$\mathbf{13.4}_{3}$	$\underline{12.1}_{1}$	$\underline{12.7}_{5}$	$\mathbf{5.3}_{3}$	$\mathbf{5.8}_{5}$	**9.94**
Ours (DeepSeek)	$\underline{8.7}_{1}$	$\mathbf{9.4}_{3}$	–	–	$\underline{10.9}_{2}$	$\underline{11.1}_{2}$	$\mathbf{12.3}_{1}$	$\mathbf{12.8}_{3}$	$\underline{4.6}_{3}$	$\underline{4.8}_{2}$	$\underline{9.33}$

Table 2: Performances of cross-domain test datasets (%).

4.2 Comparisons to the State-of-the-art Methods

Table 2 shows the performance of different methods on the five cross-domain test datasets, the proposed SRD method consistently performs better than the SOTA methods across multiple datasets. Ours(GPT4o) method exceeds ChatGPT-3.5 by 12.29% and 5.38% in absolute average values on present and absent keyphrase generations. The metrics of ChatGPT-3.5 and Ours (ChatGPT-3.5) (14.21 vs. 20.58) indicate reasonable demonstrations significantly enhance the accuracy in unsupervised cross-domain KPG. Furthermore, we also find that LLMs tend to weaken their generation capabilities on free-form text, while reasonable demonstrations effectively mitigate this negative impact. Besides, we can see that the DeepSeek model is competitive with GPT4o (26.50/9.94 vs. 25.26/9.33) on present and absent keyphrase, indicating the high cost-effectiveness of the former.

4.3 Sampling Ratio Analysis

Figure 3 shows the performance of various proportions of sample selections: 5%, 10%, 20%, 30%, and 40%. It is observed that a lower proportion of the query set results in insufficient training diversity, thereby constraining the overall effectiveness of KPG. As the proportion increases, the performance of KPG improves accordingly, reaching its peak around the 30% inclusion mark. This suggests that the optimal query set proportion likely lies between 20% and 30%. However, when the proportion is increased to 40%, the query set tends to accumulate redundant or highly similar samples.

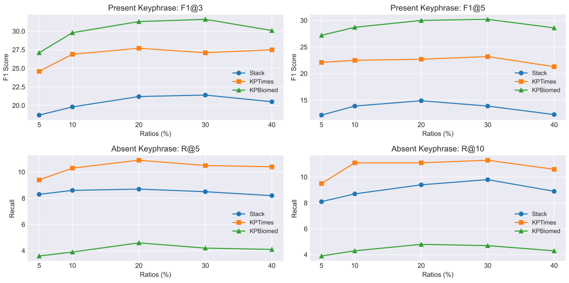

Figure 3: Performances under different ratios (%).

5 Conclusion

In this work, we present a new perspective for unsupervised cross-domain KPG with prompts LLMs to enhance its generalizability. We propose a novel seeking rational demonstrations (SRD) approach, which is the first work that introduces domain generalization on retrieve model for unsupervised cross-domain KPG. We design MMD driven distribution project loss and orthogonality loss beyond the mapping of positive and negative samples. Extensive experiments demonstrate the better performance and the effectiveness of the proposed seeking rational demonstrations approach.

6 Limitations

In addressing the practical distribution shift challenges faced by large language models in unsupervised cross-domain PKG, i.e., the difficulty of obtaining labeled target domain samples for prompting, this paper proposes a seeking rational demonstration approach. Using the Maximum Mean Discrepancy (MMD) distance as a metric for disparity, this approach retrieves highly relevant samples from the source domain to serve as demonstrations for the target samples. The main limitations of this paper are as follows: 1. The setup of the evaluator is rather rudimentary. Due to the unavailability of labeled samples from the target domain, only source domain samples are used. When encountering domains that significantly differ from the source domain, this may impact the selection of demonstrations and thus affect the quality of unsupervised cross-domain keyphrase generation. Future work could explore joint optimization of retriever and evaluator. 2. This approach does not involve optimizations for domain generalization of the large language model itself. If domain-specific instruction fine-tuning could be applied to the language model, it is believed that the model performance could be further enhanced.

7 Acknowledgements

The authors thank the anonymous reviewers for their valuable comments. This work is supported by the Scientific Research Fund of Nanjing University of Posts and Telecommunications under project numbers XK0040924146 and XK0040925024, and in part by the Scientific Research Fund of Chaohu University under project number KYQD-2025029.

References

Isabela Albuquerque, João Monteiro, Mohammad Darvishi, Tiago H Falk, and Ioannis Mitliagkas. 2019. Generalizing to unseen domains via distribution matching. *arXiv preprint arXiv:1911.00804*.

Shai Ben-David, John Blitzer, Koby Crammer, Alex Kulesza, Fernando Pereira, and Jennifer Wortman Vaughan. 2010. A theory of learning from different domains. *Machine learning*, 79:151–175.

Florian Boudin and Akiko Aizawa. 2024. Unsupervised domain adaptation for keyphrase generation using citation contexts. In *Findings of the Association for Computational Linguistics: EMNLP 2024 (Findings of EMNLP)*, pages 598–614, Miami, Florida, USA.

Florian Boudin, Ygor Gallina, and Akiko Aizawa. 2020. Keyphrase generation for scientific document retrieval. In *Proceedings of the Annual Meeting of the Association for Computational Linguistics, (ACL-20)*, pages 1118–1126, Online.

Hung Chau, Saeid Balaneshin, Kai Liu, and Ondrej Linda. 2020. Understanding the tradeoff between cost and quality of expert annotations for keyphrase extraction. In *Proceedings of the 14th Linguistic Annotation Workshop*, pages 74–86, Barcelona, Spain.

Xinyang Chen, Sinan Wang, Mingsheng Long, and Jianmin Wang. 2019. Transferability vs. discriminability: Batch spectral penalization for adversarial domain adaptation. In *International Conference on Machine Learning, (ICLR-19)*.

Billy Chiu, Sunil Kumar Sahu, Derek Thomas, Neha Sengupta, and Mohammady Mahdy. 2020. Autoencoding keyword correlation graph for document clustering. In *Proceedings of the Annual Meeting of the Association for Computational Linguistics (ACL-20)*, pages 3974–3981, Online.

Lam Do, Pritom Saha Akash, and Kevin Chen-Chuan Chang. 2023. Unsupervised open-domain keyphrase generation. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 10614–10627, Toronto, Canada.

Ygor Gallina, Florian Boudin, and Beatrice Daille. 2019. KPTimes: A large-scale dataset for keyphrase generation on news documents. In *Proceedings of the 12th International Conference on Natural Language Generation*, pages 130–135, Tokyo, Japan.

Ygor Gallina, Florian Boudin, and Béatrice Daille. 2020. Large-scale evaluation of keyphrase extraction models. In *Proceedings of the ACM/IEEE Joint Conference on Digital Libraries in 2020*, JCDL '20, pages 271–278, New York, NY, USA. Association for Computing Machinery.

Krishna Garg, Jishnu Ray Chowdhury, and Cornelia Caragea. 2023. Data augmentation for low-resource keyphrase generation. In *Findings of the Association for Computational Linguistics: ACL 2023 (Findings of ACL)*, pages 8442–8455, Toronto, Canada.

Arthur Gretton, Karsten Borgwardt, Malte Rasch, Bernhard Schölkopf, and Alex Smola. A kernel method for the two-sample-problem. In *Advances in Neural Information Processing Systems*. MIT Press.

Ishaan Gulrajani and David Lopez-Paz. In search of lost domain generalization. In *Proceedings of the International Conference on Learning Representations, (ICLR-20)*.

Maël Houbre, Florian Boudin, and Beatrice Daille. 2022. A large-scale dataset for biomedical keyphrase generation. In *Proceedings of the 13th International Workshop on Health Text Mining and Information Analysis (LOUHI)*, pages 47–53, Abu Dhabi, United Arab Emirates (Hybrid).

Anette Hulth. 2003. Improved automatic keyword extraction given more linguistic knowledge. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing, (EMNLP-03)*, pages 216–223.

Anette Hulth and Beáta B. Megyesi. 2006. A study on automatically extracted keywords in text categorization. In *Proceedings of the International Conference on Computational Linguistics and Annual Meeting of the Association for Computational Linguistics, (COLING/ACL-06)*, pages 537–544, Sydney, Australia.

Zhongtao Jiang, Yuanzhe Zhang, Kun Luo, Xiaowei Yuan, Jun Zhao, and Kang Liu. 2024. On the in-context generation of language models. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 10169–10187, Miami, Florida, USA.

Byungha Kang and Youhyun Shin. 2024. Improving low-resource keyphrase generation through unsupervised title phrase generation. In *Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING 2024)*, pages 8853–8865.

Vladimir Karpukhin, Barlas Oguz, Sewon Min, Patrick Lewis, Ledell Wu, Sergey Edunov, Danqi Chen, and Wen-tau Yih. 2020. Dense passage retrieval for open-domain question answering. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 6769–6781, Online.

Been Kim, Rajiv Khanna, and Oluwasanmi O Koyejo. 2016. Examples are not enough, learn to criticize! criticism for interpretability. In *Advances in Neural Information Processing Systems*, volume 29.

M. Krapivin and M. Marchese. 2009. Large dataset for keyphrase extraction. *Technical report, University of Trento*.

R. MartÍłnez-Cruz, A.J. LÍőpez-LÍőpez, and Portela. 2024. Chatgpt vs state-of-the-art models: a benchmarking study in keyphrase generation task. *Appl. Intell.*, 55(50).

Rui Meng, Tong Wang, Xingdi Yuan, Yingbo Zhou, and Daqing He. 2023. General-to-specific transfer labeling for domain adaptable keyphrase generation. In *Findings of the Association for Computational Linguistics: ACL 2023*, pages 1602–1618, Toronto, Canada.

Rui Meng, Sanqiang Zhao, Shuguang Han, Daqing He, Peter Brusilovsky, and Yu Chi. 2017. Deep keyphrase generation. In *Proceedings of the Annual Meeting of the Association for Computational Linguistics (ACL-17)*, pages 582–592, Vancouver, Canada.

Thuy Dung Nguyen and Min-Yen Kan. 2007. Keyphrase extraction in scientific publications. In *Asian Digital Libraries. Looking Back 10 Years and Forging New Frontiers*, pages 317–326.

Liangying Shao, Liang Zhang, Minlong Peng, Guoqi Ma, Hao Yue, Mingming Sun, and Jinsong Su. 2024. One2Set + large language model: Best partners for keyphrase generation. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing (EMNLP-24)*, pages 11140–11153, Miami, Florida, USA.

Xianjie Shen, Yinghan Wang, Rui Meng, and Jingbo Shang. 2022. Unsupervised deep keyphrase generation. In *Proceedings of the AAAI Conference on Artificial Intelligence (AAAI-22)*, pages 11303–11311.

Atsushi Ushiku, Shinsuke Mori, Hirotaka Kameko, and Yoshimasa Tsuruoka. 2017. Game state retrieval with keyword queries. In *Proceedings of the International ACM SIGIR Conference on Research and Development in Information Retrieval, (SIGIR-17)*, pages 877–880, Shinjuku, Tokyo, Japan.

Titouan Vayer and Rémi Gribonval. 2023. Controlling wasserstein distances by kernel norms with application to compressive statistical learning. *J. Mach. Learn. Res.*, 24(1).

Xiaojun Wan and Jianguo Xiao. 2008. Single document keyphrase extraction using neighborhood knowledge. In *Proceedings of the 23rd National Conference on Artificial Intelligence, (AAAI-08)*, AAAI'08, pages 855ÍC–860. AAAI Press.

Lu Wang and Claire Cardie. 2013. Domain-independent abstract generation for focused meeting summarization. In *Proceedings of the Annual Meeting of the Association for Computational Linguistics, (ACL-13)*, pages 1395–1405, Sofia, Bulgaria.

Olivia Wiles, Sven Gowal, Florian Stimberg, Sylvestre-Alvise Rebuffi, Ira Ktena, Krishnamurthy Dj Dvijotham, and Ali Taylan Cemgil. A fine-grained analysis on distribution shift. In *Proceedings of the International Conference on Learning Representations, (ICLR-22)*.

Di Wu, Wasi Ahmad, and Kai-Wei Chang. 2023. Rethinking model selection and decoding for keyphrase generation with pre-trained sequence-to-sequence models. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing (EMNLP-23)*, pages 6642–6658, Singapore.

Di Wu, Wasi Ahmad, Sunipa Dev, and Kai-Wei Chang. 2022. Representation learning for resource-constrained keyphrase generation. In *Findings of the Association for Computational Linguistics: EMNLP 2022 (Findings of EMNLP)*, pages 700–716, Abu Dhabi, United Arab Emirates.

Lee Xiong, Chuan Hu, Chenyan Xiong, Daniel Campos, and Arnold Overwijk. 2019. Open domain web keyphrase extraction beyond language modeling. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 5175–5184, Hong Kong, China.

Jiacheng Ye, Tao Gui, Yichao Luo, Yige Xu, and Qi Zhang. 2021. One2Set: Generating diverse keyphrases as a set. In *Proceedings of the Annual Meeting of the Association for Computational Linguistics (ACL-21)*, pages 4598–4608, Online.

Xingdi Yuan, Tong Wang, Rui Meng, Khushboo Thaker, Peter Brusilovsky, Daqing He, and Adam Trischler. 2020. One size does not fit all: Generating and evaluating variable number of keyphrases. In *Proceedings of the Annual Meeting of the Association for Computational Linguistics, (ACL-20)*, pages 7961–7975, Online.

A Appendix

A.1 Implementation Details

Basic Retrieve Module. To efficiently retrieve suitable demonstrations for unsupervised cross-domain keyphrase generation (KPG), we first build a basic dual-encoder retrieval module based on the DPR model[1]. The module consists of two encoders, which can map both query text and candidate text into vector representations.

Constructing Positive & Negative Pairs. Since the KP20k train dataset does not explicitly provide positive or negative sample pairs, we manually construct them by analyzing the keyphrases of each sample in the dataset. Specifically, let $\mathbf{k}_x$ and $\mathbf{k}_s$ be the keyphrase sets of two samples x and s, respectively. We first apply linguistic preprocessing such as case normalization and stemming[2] to reduce word-level discrepancies. We define the Jaccard similarity as:

$$\text{Jaccard}(\mathbf{k}_x, \mathbf{k}_s) = \frac{|\mathbf{k}_x \cap \mathbf{k}_s|}{|\mathbf{k}_x \cup \mathbf{k}_s|}. \quad (8)$$

A potential limitation of the Jaccard similarity is that keyphrases with low or zero lexical overlap may still have high semantic similarity (e.g., “deep learning” vs. “neural networks”). To better capture deep semantic relationships, we additionally employ an embedding-based metric. Each keyphrase is encoded into a vector $\mathbf{e}_{k_x}$ or $\mathbf{e}_{k_s}$ with *gte-large*[3] embedding model. We compute the cosine similarity between $\mathbf{e}_{k_x}$ and $\mathbf{e}_{k_s}$. The relevance score is defined as:

$$\text{Rel}(x, s) = \alpha \cdot \text{sim}_{\text{embed}}(\mathbf{k}_x, \mathbf{k}_s) + (1-\alpha) \cdot \text{Jaccard}(\mathbf{k}_x, \mathbf{k}_s), \quad (9)$$

where $0 \leq \alpha \leq 1$ is a balancing coefficient. If $\text{Rel}(x, s)$ exceeds a global threshold γ_1, we classify (x, s) as a positive (matched) pair, and otherwise treat it as a negative (unmatched) pair.

Self-supervised Demonstrations Evaluator. To assess the efficacy of retrieved demonstrations in enhancing keyphrase generation within an unlabeled target domain, we design a self-supervised evaluator utilizing the KP20k train dataset. We divide KP20k into two disjoint subsets: the pseudo-source domain $\mathcal{D}_{\text{ps}}$ and pseudo-target domain $\mathcal{D}_{\text{pt}}$ with K-means algorithm[4]. The former is used to simulate the source domain for retrieval purposes, and the latter is used to mimic the unlabeled target domain scenario. We then train a regression model with RoBERTa[5] to learn a function mapping the triplet $(\mathbf{x}, \mathbf{e}, \mathbf{y}^{\text{gen}})$ to a quality score $\hat{s}$, where $\mathbf{x}$ represents the abstract text in the pseudo-target domain, $\mathbf{e}$ indicates the retrieved examples, and $\mathbf{y}^{\text{gen}}$ are the keyphrases generated by LLM. Let $\mathbf{y}^{\text{ref}}$ denote the ground-truth keyphrases for $\mathbf{x}$ in the pseudo-target domain. The quality score s of $\mathbf{y}^{\text{gen}}$ is computed by the defined evaluation metric in Eq.9, i.e., $s = \text{Rel}(\mathbf{y}^{\text{gen}}, \mathbf{y}^{\text{ref}})$. Formally, let f_θ represent the evaluator, the training objective is to minimize the mean squared error (MSE) between $\hat{s}$ and s:

$$\min_{\theta} \mathbb{E}_{(\mathbf{x},\mathbf{e},\mathbf{y}^{\text{gen}},\mathbf{y}^{\text{ref}})\sim\mathcal{D}} \Big[\big(f_\theta(\mathbf{x}, \mathbf{e}, \mathbf{y}^{\text{gen}}) - s \big)^2 \Big], \quad (10)$$

where $\mathcal{D}$ is the dataset constructed from both $\mathcal{D}_{\text{ps}}$ and $\mathcal{D}_{\text{pt}}$. We use multiple checkpoints of the trained retrieval module to obtain various $\mathbf{y}^{\text{gen}}$ to better train the regression model.

Expermental Details. We use the open-source instruction fine-tuned LLMs as backbone. That is Qwen-2.5-14b[6] and LLama-3.3-70b-4bit[7]. We also use several LLMs with commercial API, that is ChatGPT-3.5[8], GPT4o[9] and DeepSeek[10]. The temperature is set to be 0.4. We set the hyperparameters with the KPTimes validation dataset. The learning rate is 3e-5 and the optimizer is AdamW. We retain only the top 2 demonstrations to prompt the LLMs, avoiding that an excessive number of examples would introduce significant additional overhead. We train the model on 2 A40 48GB GPUs and use a cosine scheduler with a 2% warm-up period for 3 epochs. The ratio α in Eq.9 is set to be 0.8, the global threshold γ_1 is set to be 0.70.

A.2 Ablation Study

We perform comprehensive ablation studies based on the DeepSeek model to show the effectiveness of the components through the performance on the three different domain test datasets.

- Zero-shot: This variant only generates keyphrases without demonstrations.

[1] https://github.com/facebookresearch/DPR/

[2] We use the NLTK toolkit to do processing. `https://www.nltk.org/`

[3] https://huggingface.co/thenlper/gte-large

[4] https://scikit-learn.org/

[5] https://huggingface.co/FacebookAI/roberta-base

[6] https://huggingface.co/Qwen/Qwen2.5-14B-Instruct

[7] https://huggingface.co/ibnzterrell/Meta-Llama-3.3-70B-Instruct-AWQ-INT4

[8] gpt-3.5-turbo-0613

[9] gpt-4o-2024-11-20

[10] deepseek/deepseek-chat

- Random-D: This variant randomly selects samples from the KP20k train dataset to prompt the DeepSeek model.

- Embed-D: This variant solely uses semantic similarity to select samples from the KP20k train dataset to prompt the DeepSeek model.

- w/o MMD: this variant removes the MMD loss during the training of the retrieval module.

- w/o Orthogonality: this variant removes the Orthogonality loss during the training of the retrieval module.

- w/o MDD + Orthogonality: This variant only uses contrastive loss to train the retrieval module.

From Table 3, we can conclude that: The performance of zero-shot is the worst, indicating that while large language models exhibit strong generalization ability, they struggle to leverage this advantage when facing samples from new domains. The performance of random-D shows only a limited improvement over zero-shot, suggesting that inaccurate prompts are ineffective in enhancing the performance of large language models. Embed-D performs better than random-D. It indicates that semantically similar samples provide meaningful assistance to large language models.

Compared to the Embed-D method, the results of w/o MMD, w/o Orthogonality, and w/o MMD+Orthogonality are even better, highlighting the effectiveness of the proposed "seek rational demonstration" approach in this paper. This approach, through the retrieval and evaluation modules, is able to provide large language models with the most reasonable sample examples, thereby significantly enhancing their unsupervised cross-domain keyphrase generation ability.

Present keyphrase generation						
Methods	StackExchange		KPTimes		KPBiomed	
	F1@3	F1@5	F1@3	F1@5	F1@3	F1@5
Zero-shot	5.4	3.7	11.0	8.5	18.4	17.2
Random-D	6.3	3.4	17.5	10.5	8.6	18.1
Embed-D	18.9	8.9	23.2	20.8	28.7	25.5
Ours	**21.2**	**14.9**	**27.7**	**22.7**	**31.3**	**30.0**
Ours(w/o MMD)	19.6	10.2	26.8	21.1	27.9	27.3
Ours(w/o Orthogonality)	19.8	12.8	27.4	21.5	30.1	28.3
Ours(w/o MDD+Orthogonality)	18.2	9.4	25.4	17.6	27.4	26.6
Absent keyphrase generation						
Methods	StackExchange		KPTimes		KPBiomed	
	R@5	R@10	R@5	R@10	R@5	R@10
Zero-shot	3.1	3.2	5.8	6.0	2.9	2.9
Random-D	3.7	3.2	5.7	5.9	3.2	2.9
Embed-D	6.6	7.1	7.4	9.2	3.8	3.1
Ours	**8.7**	**9.4**	**10.9**	**11.1**	**4.6**	4.8
Ours(w/o MMD)	7.5	7.7	9.2	9.4	3.9	4.6
Ours(w/o Orthogonality)	8.1	7.5	9.8	10.5	4.2	**5.1**
Ours(w/o MDD+Orthogonality)	6.8	6.5	8.4	8.9	4.2	4.3

Table 3: Ablation study of our SRD approach across three benchmark datasets. Best scores in **bold**, second-best underlined.

Present keyphrase generation							
Methods	StackExchange		KPTimes		KPBiomed		Average
	F1@3	F1@5	F1@3	F1@5	F1@3	F1@5	
Ours	21.2	14.9	27.7	22.7	31.3	30.0	24.63
Random-T	22.1	15.3	27.5	23.6	30.8	29.9	24.87
Ours+few-shot	**23.2**	**16.5**	**29.5**	**24.4**	**33.1**	**31.6**	**26.38**
Absent keyphrase generation							
Methods	StackExchange		KPTimes		KPBiomed		Average
	R@5	R@10	R@5	R@10	R@5	R@10	
Ours	8.7	9.4	10.9	11.1	4.6	4.8	8.25
Random-T	8.6	10.1	10.5	11.8	4.5	5.7	8.53
Ours+few-shot	**10.5**	**11.2**	**11.7**	**12.5**	**6.3**	**6.5**	**9.78**

Table 4: In-domain evaluation across three benchmark datasets.

A.3 In-domain Sample Evaluation

Our SRD method is specifically designed for the unsupervised cross-domain KPG task. For small-scale labeled target domain samples (e.g., 10 labeled target domain samples), we can directly use the target domain supervised samples as prompts, integrating them into our SRD method. To evaluate the experimental effects, we designed the following experiments with DeepSeek model:

- Random-T: It selects the first 10 samples from the validation sets of their respective datasets, then randomly choose 2 samples as target domain demonstrations to evaluate the generative capability of the LLMs.

- Ours+few-shot: We add 2 randomly selected labeled target domain samples as demonstrations.

From Table 4, we can observe that: The SRD is slightly inferior to supervised target domain KPG, indicates that demonstrations from the same domain have a significant impact on the KPG via LLMs. The SRD+few-shot yields the best results, suggesting that equipping SRD with labeled target domain samples further enhances its KPG capabilities. We infer that, in the context of unsupervised cross-domain KPG, the absence of relevant target domain information usually make it challenging to surpass supervised cross-domain KPG tasks. The assistance of a few-shot approach undoubtedly enhances our method significantly.

A.4 MMD for evaluating distribution shift between two different datasets

The maximum mean discrepancy (MMD) is a measure of the difference between distributions $\mathcal{P}$ and

$\mathcal{Q}$, given by the supremum over a function space $\mathcal{F}$ of differences between the expectations with respect to two distributions. Given n samples from $\mathcal{P}$ as X $= \{x_i \sim P, i \in [n]\}$, and m samples from $\mathcal{Q}$ as $\mathbf{Z} = \{z_i \sim Q, i \in [m]\}$, the following is a finite sample approximation:

$$\begin{aligned} \mathrm{MMD}(\mathcal{F}, P, Q) = \sup_{f \in \mathcal{F}} (&\mathrm{E}_{X \sim P}[f(X)] \\ &- \mathrm{E}_{Y \sim Q}[f(Y)]). \end{aligned} \tag{11}$$

Inspired by the sampling technique for interpretable machine learning (Kim et al., 2016), we adopt the squared maximum mean discrepancy (MMD) between $\mathcal{S}$ and $\mathcal{P}$ with a kernel function k to measure the discrepancy between them:

$$\begin{aligned} \mathrm{MMD}_k^2(\mathcal{F}, \mathcal{S}, \mathcal{P}) = &\frac{1}{|\mathcal{S}|^2} \sum_{\mathbf{s}_i, \mathbf{s}_j \in \mathcal{S}} k\left(\mathbf{s}_i, \mathbf{s}_j\right) \\ &- \frac{2}{|\mathcal{S}||\mathcal{P}|} \sum_{\mathbf{s}_i \in \mathcal{S}, \mathbf{p}_j \in \mathcal{T}} k\left(\mathbf{s}_i, \mathbf{p}_j\right) \\ &+ \frac{1}{|\mathcal{P}|^2} \sum_{\mathbf{p}_i, \mathbf{p}_j \in \mathcal{P}} k\left(\mathbf{p}_i, \mathbf{p}_j\right). \end{aligned} \tag{12}$$

It is clear that $\mathrm{MMD}^2(\mathcal{F}, P, Q) \geq 0$ and $\mathrm{MMD}^2(\mathcal{F}, P, Q) = 0$ iff. $\mathcal{P}$ is indistinguishable from $\mathcal{Q}$ on the RHKS $\mathcal{F}$.

When $\mathcal{F}$ is a reproducing kernel Hilbert space (RKHS) with kernel function $k : \mathcal{X} \times \mathcal{X} \mapsto \mathbb{R}$, the supremum is achieved at (Gretton et al.):

$$f(x) = \mathrm{E}_{X' \sim P}\left[k\left(x, X'\right)\right] - \mathrm{E}_{X' \sim Q}\left[k\left(x, X'\right)\right], \tag{13}$$

and the witness function is approximated as:

$$f(x) = \frac{1}{n} \sum_{i \in [n]} k\left(x, x_i\right) - \frac{1}{m} \sum_{j \in [m]} k\left(x, z_j\right). \tag{14}$$

The function 14 is also known as the witness function as it measures the maximum discrepancy between the two expectations in $\mathcal{F}$.

A.5 Prompt Templates

For the zero-shot setting of LLMs for unsupervised cross-domain PKG, we use the following prompt template.

```
=======
Please generate accurate present and
absent keyphrases in lowercase for
the given sample.
=======
###Sample###:
"text": "{text}"
**Please only output at least five
present and absent keyphrases with
standard JSON format.**
```

Given the retrieved demonstrations, we can use the following prompt template for unsupervised cross-domain PKG.

```
=======
Please refer to the demonstrations
generate accurate present and absent
keyphrases with lowercase for the given
sample.
{demonstrations_case}
=======
###Sample###:
"text": "{text}"
The MMD distances between the
demonstrations and the sample are {MMD}.
**Only output present and absent
keyphrases with standard JSON format.
```