PRESS: Defending Privacy in Retrieval-Augmented Generation via Embedding Space Shifting

Jiaming He[1,2*], Cheng Liu[3*], Guanyu Hou[2*], Wenbo Jiang[1†], Jiachen Li[4]
[1]School of Computer Science and Engineering, University of Electronic Science and Technology of China, China
[2]College of Computer Science and Cyber Security (Oxford Brookes College), Chengdu University of Technology, China
[3] School of Data Science, The Chinese University of Hong Kong, Shenzhen, China
[4] School of Computer Science and Artificial Intelligence, Wuhan University of Technology, China

***Abstract*—Retrieval-augmented generation (RAG) expands the capabilities of large language models (LLMs) in various applications by integrating relevant information retrieved from external data sources. However, the RAG systems are exposed to substantial privacy risks during the information retrieval process, leading to potential data leakage of private information. In this work, we present a Privacy-preserving Retrieval-augmented generation via Embedding Space Shifting (PRESS), systematically exploring how to protect privacy in RAG systems. Specifically, we first conduct proximal policy optimization (PPO) based training on pre-trained language models to generate target training samples. Then we employ a purposive shift fine-tuning on the text embedding model with the generated samples for guiding the RAG system to map potential privacy leaking queries to safe target in embedding space. Extensive experimental results on representative models and datasets demonstrate that our protection method achieves high defense performance with high efficiency while keeping the normal functionality of the RAG system.**

I. Introduction

Retrieval-augmented generation (RAG) [1]–[3] is an advanced technique that enhances large language models (LLMs) by dynamically retrieving information from external data relevant to input queries. This paradigm has been widely applied in various scenarios, including knowledge QA [4], code completion [5] and outperformed chatbots [6], for improving fact-checking and real-world information retrieval during model inference. In particular, a RAG-integrated LLM system is conducted in two stages: information retrieval and response generation. As the first step, the system retrieves relevant contexts from a large corpus based on the input query. Secondly, LLMs leverage accurate retrieval information to generate detailed responses, enhancing the accuracy and quality of generated responses.

Unfortunately, many studies have demonstrated that RAG systems may face the potential risk of privacy leakage in various applications. For instance, Zeng et al. [7] first propose the concepts of **targeted attack** and **untargeted attack** and find that RAG system is vulnerable to carefully designed prompts for extracting complete contexts (**untargeted attack**) or specific pieces of private information (**targeted attack**) in the retrieved data. Qi et al. [8] propose a data extraction attack against RAG system by injecting the adversarial prompt and exploiting the instruction-following capabilities of LLMs. In the aspect of real-world application, pre-trained chatbots that are used for medical diagnosis may rely on diagnosis cases from real patients as external knowledge but also raise concerns about private individual information. Therefore, it's crucial to build privacy protection on RAG system to enhance the privacy security of RAG and prevent potential privacy leakage and data stealing in regular RAG inference, such as the leakage of sensitive information from patients.

† Corresponding author
* These authors equally contributed to this work.

To mitigate such privacy risks in the RAG system, some pre- and post-processing methods [9]–[11] are proposed for defense, such as similarity distance measurement and context summarization. However, the existing studies present that these methods are weak in the cases of private data existing in retrieved information. Moreover, Zeng et al. [11] present a solution via synthesizing clean data with retrieved contexts during inference, but pure generations on each retrieved-context are inefficient in real-world applications.

To address the above challenges, we propose a **P**rivacy-preserving **R**etrieval-augmented generation via **E**mbedding **S**pace **S**hifting (**PRESS**). To prevent private data leaks from retrieval contexts, it's crucial to ensure that the private-sensitive data is not retrieved as referencing contexts. The key insight of our method is to guide a privacy-safe retrieval process via shifting in embedding space. We propose an automatic framework to generate tuning examples, which are used for shifting fine-tuning on the embedding model. Our experimental results show that adopting our privacy-secured embedding model can achieve remarkable performances with high efficiency while maintaining the accuracy of the normal retrieval process.

II. Proposed Method-**PRESS**

This section details a methodological approach that integrates supervised fine-tuning, reward-based optimization, reinforcement learning and embedding model fine-tuning to implement an effective embedding shifting of which the aim is defending privacy in RAG. Figure 1 outlines the pipeline of our **PRESS**.

A. Supervised Fine-Tuning

Initially, we employ supervised fine-tuning (SFT) to train two pre-trained LLMs to get models π_{θ_p} and π_{θ_q}, π_{θ_p} for generating examples contained responses without private data, and π_{θ_q} for examples with private data. To fine-tune these two models, we adopt positive dataset D_p^{pos} without private data in responses to fine-tune π_{θ_p}, and negative dataset D_p^{neg} with private data in responses to fine-tune π_{θ_q}. Both datasets comprise N_p pairs of prompts x_i^t aiming to reveal specific information and prompts x_i^u aiming to obtain fully restate from the retrieved contexts and responses y_i:

$$D_p = \{(x_i^t, y_i)\}_{i=1}^{\frac{1}{2}N_p} \cup \{(x_i^u, y_i)\}_{i=1}^{\frac{1}{2}N_p} \tag{1}$$

where D_p donates the dataset used to train the LLM (D_p^{pos} and D_p^{neg}), y_i is response of the query x_i^t or x_i^u. The goal of this fine-tuning is to find the weights θ_p and θ_q that minimize the loss L_p on the dataset D_p:

$$L_p = -\frac{1}{N_p} \sum_{(x,y)\in D_p} \log(p(y \mid x)) \tag{2}$$

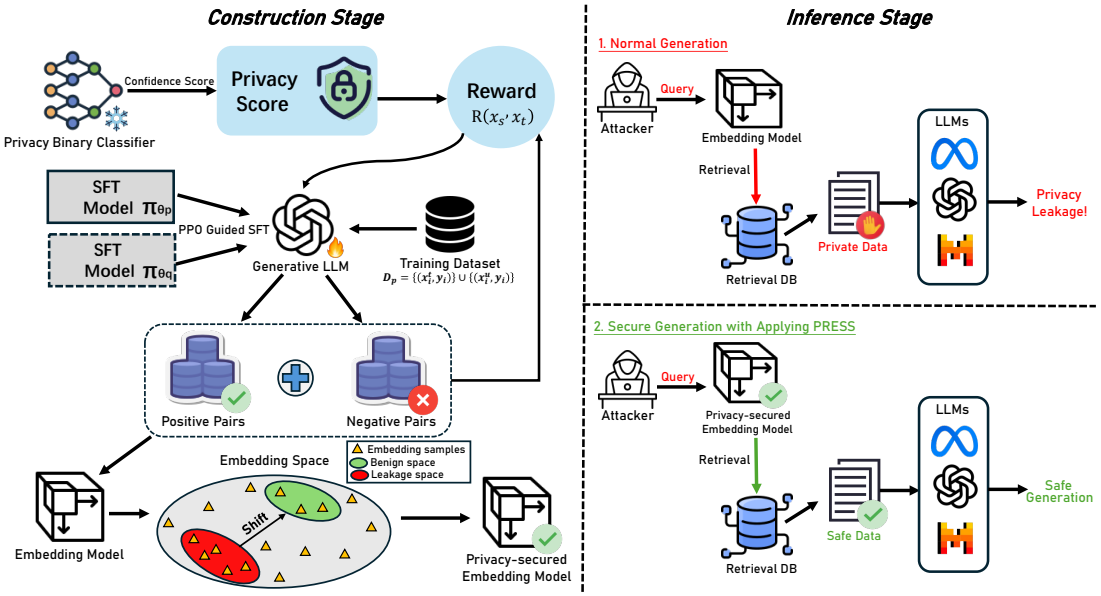


Fig. 1: Overview of our **PRESS** for defending privacy in RAG systems.

where D_p denotes the training dataset, consisting of input-output pairs (x, y), $p(y \mid x)$ is the probability of the model outputting sequence and y denotes an example include a query about private data and a reply without revealing such information, given the input x which is a prompt to instruct the model to generate such example.

B. Reward definition

In order to obtain high-quality examples for fine-tuning the text embedding model, we quantify the quality of optimized examples which is based on the extent to which the response in the example contains no private information. A reward function $R(x, y)$ is set according to the above quantification.

To measure the potential privacy leakage of the responses in the examples (i.e., the extent to which they do not contain private information), we train a Bidirectional Encoder Representations from Transformers model (BERT) [12] by fake_person_info dataset [13] and employ it to perform binary classification, thereby obtaining the privacy score $f_{private}$ that the text is classified as containing private information:

$$f_{private}(x) = E_{P' \sim P(y_p|x)}[BERT_{cfi}(P')] \tag{3}$$

where $P(y_p \mid x)$ is the probability that the input text x belongs to the class y_p, which in this context, is the class indicating the presence of private information, P' is the probability distribution outputted from BERT, $BERT_{cfi}(P')$ denotes the confidence score of predicting the response as privacy leakage.

Subsequently, we adopt this privacy score $f_{private}$ to define the reward function. Additionally, to prevent excessive optimization and over-fitting [14], we have incorporated the Kullback-Leibler (KL) divergence into the reward function:

$$R(x, y) = -f_{private} - \eta log \frac{\mu_\theta(y \mid x)}{\mu_{\theta'}(y \mid x)} \tag{4}$$

where μ_θ is the policy model, $\mu_{\theta'}$ is the supervised fine-tuned model, η is the coefficient to modulate the weight of the KL divergence penalty.

C. Reinforcement learning

We train the policy models via reinforcement learning (RL) to generate high-quality examples which used to fine-turn the embedding model. To improve the quality of generated examples and inspired by Promptist [15], proximal policy optimization (PPO) [16] is adopted to implement this process. With the weights initialized from π_{θ_p} and π_{θ_q}, we train policy models μ_{θ_p} and μ_{θ_q} using the reward function $R(x, y)$ and the negative value of $R(x, y)$. The goal of RL is to maximize the accumulated expected reward over a training set $D_{ins} = \{x_{ins}\}$ which contains several prompts x_{ins} about the examples generation:

$$J = E_{x \sim D_{ins}, y \sim \pi_\theta}[R(x_{ins}, y)] \tag{5}$$

where J is the objective function that the reinforcement learning aims to maximize, and R is the reward function used to guide the process of PPO. x_{ins} is the prompt used to instruct model to output examples, y is the output of the policy models under the instruction x_{ins}.

D. Embedding Shifting Fine-tuning

After gathering all the necessary examples from the policy model, we create a dataset $D_{emb} = \{(x_i^p, y_i^q, y_i^{q'})\}_{i=1}^{N_{emb}}$ where p and q represent a positive example pair, q' represents a negative example, based on these examples with N_{emb} query-response pairs to fine-tune a pre-trained embedding model, which is used to convert the query into a text embedding—a vector with N_{dim} dimensionality that carries the semantic information of the inputted text. The purpose of this fine-tuning is to identify a set of parameters ${\theta_{emb}}'$ that minimize the loss L_{emb} [17] on the dataset D_{emb}, thereby to align the distribution of text embeddings for sensitive queries more closely with those of benign responses within the dataset D_{emb}:

$$L_{emb} = \sum_{i=1}^{N_{emb}} \log \frac{e^{<e_p, e_q>/\tau}}{e^{<e_p, e_q>/\tau} + \sum_{q' \in Q'} e^{<e_p, e_{q'}>/\tau}} \tag{6}$$

where e_p, e_q and $e_{q'}$ donate text embeddings encoded from x_i^p, y_i^q and $y_i^{q'}$, Q' is the set of negative examples, $< e_p, e_q >$ and $< e_p, e_{q'} >$ denotes the dot product between the embeddings of the positive example pair and dot product between the negative example

pair respectively, τ is the temperature parameter, which controls the smoothness of the softmax function.

III. Experiments

A. Experimental Setup

RAG Components. In our experiments, we primarily utilize Llama3-8b, GPT4o, and GPT-3.5-turbo as language models for text generation and performance evaluation. For the embedding models, we employ BGE-base-en-v1.5 and BGE-large-en-v1.5 [18], which are fine-tuned for vector retrieval and embedding space shifting, to facilitate performance comparison.

Retrieval Datasets. To assess the effectiveness of our approach, we consider two privacy-related datasets: the Enron Email dataset, which consists of 500,000 employee emails, and the HealthcareMagic-101 dataset [19], comprising 200,000 doctor-patient medical dialogues. For the HealthcareMagic dataset, each doctor-patient dialogue is treated as an individual data piece for embedding. In the Enron Email dataset, each email is treated as an individual data unit for embedding. Additionally, for each original retrieval dataset, we automatically generated and injected a 1% rejection dataset using an agent. This dataset acts as a defense space to enhance privacy protection.

Implementation Details. In our experiments, we randomly sampled 1% of the original dataset and used a specially designed agent, trained with an RL strategy, to automatically generate generalized pairs in format: {*query: positive_retrieval_text*}. For each sampled instance, the agent generated two targeted attack query pairs, one untargeted query pair, and two safe query pairs, forming the fine-tuning dataset. The learning rate for both embedding models was set to 2e-5, with a batch size of 64. The temperature for generation was set to 0.4.

Baselines. In this paper, We compare three baseline methods: LLM generation-based methods, including **ZeroGen** [20] and **AttrPrompt** [21], and the pure data generation method **SAGE** [11], including attributes-based generation (**Stage-1**) and full generation (**Stage-2**).

B. Evaluation Metrics

We evaluate two main aspects of the experiments: the retrieval performance after embedding model fine-tuning with offset and the in-context generation effectiveness.

Embedding Fine-tuning Offset Evaluation. We use Recall@5 and nDCG@5 to assess the retrieval performance of the embedding model before and after fine-tuning with offset. This ensures that the model maintains its original performance while exhibiting the ability to shift mappings for attack queries.

In-Context Generation Evaluation. We define an untargeted attack as 'successful' if its output contains 10 consecutive tokens that match the original dataset, while a targeted attack is considered 'successful' if its output contains at least 10 repeated tokens from the original dataset. We report the number of successful prompts (**Repeat Prompt**) for each attack type. This metric serves to evaluate the method's defense performance against adversarial prompts, as well as its ability to preserve the quality of responses for normal prompts. In addition, we report the average ROUGE-L [22] and BLEU-L [23] scores for different types of prompts to assess the similarity between the generated responses and the original retrieved text.

C. Experimental Results

Evaluation of Retrieval Defense Effectiveness. The experimental results are presented in Table I and II. These results clearly underscore the effectiveness of embedding space shifting in mitigating privacy risks from both targeted and untargeted attacks. In the baseline models (BGE-Base and BGE-Large), we observe significant recall values for attack queries, with recalls reaching up to 15.22 for targeted and 10.64 for untargeted queries in the HealthCareMagic dataset, and 6.01 and 6.32, respectively, for the Enron Email dataset. This highlights substantial data leakage. However, after fine-tuning (BGE-Base-FT and BGE-Large-FT), recall values for attack queries drop drastically, nearly approaching zero, with values like 1.69 and 1.53 for targeted, and as low as 0.00 for untargeted queries in the Enron dataset. Meanwhile, the recall and nDCG values for safe queries are preserved or even improved, reaching up to 38.14 in recall and 26.10 in nDCG for BGE-Large-FT on the HealthCareMagic dataset, and 41.29 in recall and 32.24 in nDCG on the Enron Mail dataset. These results highlight the method's capability to defend against extraction attacks effectively, all while maintaining high retrieval performance for benign queries, reinforcing the robustness of embedding space shifting as a privacy-preserving solution without sacrificing retrieval quality.

TABLE I: Retrieval results on HealthcareMagic dataset

Method	Target		Untarget		Safe	
	Recall	nDCG	Recall	nDCG	Recall	nDCG
BGE-Base	15.22	12.34	10.64	7.25	27.48	21.24
BGE-Base-FT	1.69	1.22	**3.20**	**2.18**	35.16	24.08
BGE-Large	18.45	14.43	9.13	6.26	28.97	22.55
BGE-Large-FT	**1.53**	**1.11**	3.33	2.34	**38.14**	**26.10**

TABLE II: Retrieval results on Enron Email dataset

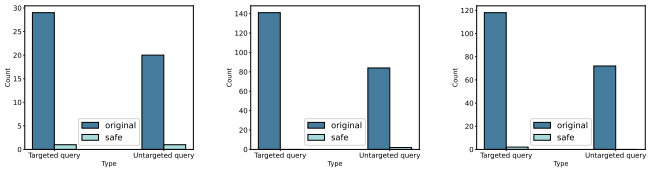
Method	Target		Untarget		Safe	
	Recall	nDCG	Recall	nDCG	Recall	nDCG
BGE-Base	6.01	4.15	6.32	3.61	9.62	6.41
BGE-Base-FT	0.72	0.58	**0.00**	**0.00**	39.80	31.07
BGE-Large	7.53	5.33	8.46	4.67	13.81	9.68
BGE-Large-FT	**0.31**	**0.00**	0.22	0.00	**41.29**	**32.24**

Performance Comparison of Defense Methods. In industrial applications, both defense effectiveness and computational efficiency are crucial. From Table III, we compares PRESS with other methods on these metrics. PRESS significantly reduces the number of failures from the original method's 81 (targeted) and 67 (untargeted) to just 2 and 1, respectively, effectively preventing data leakage. While methods like AttrPrompt and SAGE-stage-2 achieve zero failures, they do so at the expense of substantially higher computational times (1466.35 seconds and 4635.99 seconds, respectively). In contrast, PRESS accomplishes near-optimal defense performance in only 149.05 seconds, drastically reducing computational overhead compared to prior methods such as ZeroGen (1931.78 seconds). These results highlight that PRESS offers a superior balance between defense efficacy and efficiency, making it highly suitable for deployment in privacy-sensitive applications where both security and performance are critical.

In-Context Generation Defense Results. Figure 2 and 3 display the Repeat Prompt, ROUGE, and BLEU scores across different attack types and datasets for our proposed method. The conclusion of targeted attacks is consistent with that of untargeted attacks across various datasets: compared to the original method, PRESS significantly reduces the leakage of private information across all evaluated models, attack types, and datasets. This demonstrates

TABLE III: Comparison of defense performance and implementation time for different methods on HealthcareMagic (300 prompts)

Method	Target		Untarget		Time (seconds)
	Repeat	Prompt	Repeat	Prompt	
Origin	81		67		-
ZeroGen	4		0		1931.78
AttrPrompt	0		0		1466.35
SAGE-stage-1	12		4		2666.38
SAGE-stage-2	0		0		4635.99
PRESS (Ours)	2		1		**149.05**



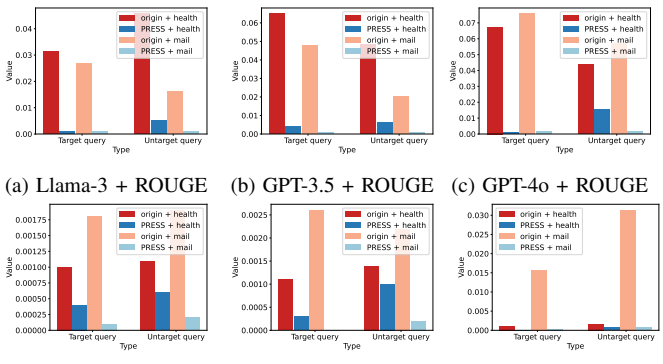
(a) Llama-3 + health (b) GPT-3.5 + health (c) GPT-4o + health

(d) Llama-3 + mail (e) GPT-3.5 + mail (f) GPT-4o + mail

Fig. 2: Repeat prompt results of attack queries on different models and datasets ("safe" indicates results using the PRESS).

PRESS's strong adaptability and robust defense. While the original method experienced a peak leakage of 141 out of 300 prompts, our method, even at its weakest, allowed leakage in only 3 out of 300 prompts, representing a defense improvement by several orders of magnitude. Furthermore, Fig. 3 shows that responses generated by our method contain "purer" information.

From these results, it's evident that the defense performance varies between different generative models. Both in the baseline and with PRESS, the llama-3-8b model outperforms the GPT models in terms of defense, likely due to the generated outputs of llama-3-8b is of poor quality, which fails to retain information from the original data.



(a) Llama-3 + ROUGE (b) GPT-3.5 + ROUGE (c) GPT-4o + ROUGE

(d) Llama-3 + BLEU (e) GPT-3.5 + BLEU (f) GPT-4o + BLEU

Fig. 3: ROUGE and BLEU results of attack queries on different models and datasets.

TABLE IV: In-context generation performance variation on normal queries

Models	Healthcare		Mail	
	ROUGE	BLEU	ROUGE	BLEU
GPT-3.5-turbo	+3.2%	+2.3%	+2.8%	+2.3%
GPT-4o	**+4.2%**	**+3.0%**	+0.3%	+1.8%
Llama-3-8b	+1.7%	+2.0%	+2.1%	+2.8%

Normal-Functionality Preserving. To evaluate the normal performance of models, we prepare normal and safe queries and respective standard responses. For the three candidate models, we conduct an evaluation of the normal functionality of the RAG systems by measuring the metrics such as ROUGE and BLEU scores between the generated responses and standard responses. The results in Table IV demonstrate that equipping our privacy-secured embedding model can even increase the normal performance. It's noteworthy that the normal performance of GPT-4o in the Healthcare dataset increases the most compared with other cases.

(a) Source Embedding Space (b) Shifted Embedding Space

Fig. 4: Visualization of embedding space (the embedding values encoded from input queries, red represents untargeted, purple represents targeted, and blue represents safe.).

Embedding Space Analysis. Theoretically, the defense effectiveness of our method is largely dependent on the shifting in the embedding space. To sufficiently prove this viewpoint, We perform the visualization of embedding space from the original embedding model and the safe-shifted embedding model by using Principal Component Analysis (PCA) [24] for embedding projection. In the right of Fig. 4, we can observe that the embedding distribution of targeted and untargeted queries is shifted while clearly separating from the normal queries, indicating our embedding model exhibits the capability to safely isolate the privacy-sensitive queries, while normally retrieving the relevant contexts for normal queries.

IV. Conclusion

In this paper, we present a defensive method for protecting privacy data in RAG systems, via efficient fine-tuning on the text embedding model. We propose an automatic framework, which includes shifting sample generation, which aims to generate influential positive and negative pairs, and embedding shifting fine-tuning, which aims to guide the query embedding away from leakage space while shifting to privacy-safe space. Extensive experimental results demonstrate that our defensive method achieves high defense performance while maintaining retrieval performance on normal queries. With our work, we hope future research is motivated to investigate privacy concerns in RAG systems.

ACKNOWLEDGMENT

This work is supported by the National Natural Science Foundation of China under Grant 62402087 and the Postdoctoral Innovation Talents Support Program under Grant BX20230060.

REFERENCES

[1] J. Chen, H. Lin, X. Han, and L. Sun, "Benchmarking large language models in retrieval-augmented generation," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 38, no. 16, 2024, pp. 17 754–17 762.

[2] W. Shi, S. Min, M. Yasunaga, M. Seo, R. James, M. Lewis, L. Zettlemoyer, and W.-t. Yih, "Replug: Retrieval-augmented black-box language models," *arXiv preprint arXiv:2301.12652*, 2023.

[3] D. Van Veen, C. Van Uden, L. Blankemeier, J.-B. Delbrouck, A. Aali, C. Bluethgen, A. Pareek, M. Polacin, E. P. Reis, A. Seehofnerova *et al.*, "Clinical text summarization: adapting large language models can outperform human experts; 2023," *Preprint at¡ https://doi. org/10.48550/arXiv*, vol. 2309, 2023.

[4] J. Kim, J. Nam, S. Mo, J. Park, S.-W. Lee, M. Seo, J.-W. Ha, and J. Shin, "Sure: Improving open-domain question answering of llms via summarized retrieval," in *The Twelfth International Conference on Learning Representations*, 2023.

[5] D. Wu, W. U. Ahmad, D. Zhang, M. K. Ramanathan, and X. Ma, "Repoformer: Selective retrieval for repository-level code completion," *arXiv preprint arXiv:2403.10059*, 2024.

[6] M. Kulkarni, P. Tangarajan, K. Kim, and A. Trivedi, "Reinforcement learning for optimizing rag for domain chatbots," *arXiv preprint arXiv:2401.06800*, 2024.

[7] S. Zeng, J. Zhang, P. He, Y. Xing, Y. Liu, H. Xu, J. Ren, S. Wang, D. Yin, Y. Chang *et al.*, "The good and the bad: Exploring privacy issues in retrieval-augmented generation (rag)," *arXiv preprint arXiv:2402.16893*, 2024.

[8] Z. Qi, H. Zhang, E. Xing, S. Kakade, and H. Lakkaraju, "Follow my instruction and spill the beans: Scalable data extraction from retrieval-augmented generation systems," *arXiv preprint arXiv:2402.17840*, 2024.

[9] H. Chase, "Langchain," 2022, accessed on December 26, 2024. [Online]. Available: https://github.com/hwchase17/langchain.

[10] Y. Huang, S. Gupta, Z. Zhong, K. Li, and D. Chen, "Privacy implications of retrieval-based language models," *arXiv preprint arXiv:2305.14888*, 2023.

[11] S. Zeng, J. Zhang, P. He, J. Ren, T. Zheng, H. Lu, H. Xu, H. Liu, Y. Xing, and J. Tang, "Mitigating the privacy issues in retrieval-augmented generation (rag) via pure synthetic data," *arXiv preprint arXiv:2406.14773*, 2024.

[12] J. D. M.-W. C. Kenton and L. K. Toutanova, "Bert: Pre-training of deep bidirectional transformers for language understanding," in *Proceedings of naacL-HLT*, vol. 1, 2019, p. 2.

[13] A. Belkhadir, "Fake person info," 2024, accessed on December 26, 2024. [Online]. Available: https://huggingface.co/datasets/Phantom-II/fake_person_info

[14] L. Ouyang, J. Wu, X. Jiang, D. Almeida, C. Wainwright, P. Mishkin, C. Zhang, S. Agarwal, K. Slama, A. Ray *et al.*, "Training language models to follow instructions with human feedback," *Advances in neural information processing systems*, vol. 35, pp. 27 730–27 744, 2022.

[15] Y. Hao, Z. Chi, L. Dong, and F. Wei, "Optimizing prompts for text-to-image generation," *Advances in Neural Information Processing Systems*, vol. 36, 2024.

[16] J. Schulman, F. Wolski, P. Dhariwal, A. Radford, and O. Klimov, "Proximal policy optimization algorithms," *arXiv preprint arXiv:1707.06347*, 2017.

[17] V. Karpukhin, B. Oguz, S. Min, P. Lewis, L. Wu, S. Edunov, D. Chen, and W.-t. Yih, "Dense passage retrieval for open-domain question answering," in *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, B. Webber, T. Cohn, Y. He, and Y. Liu, Eds. Online: Association for Computational Linguistics, Nov. 2020, pp. 6769–6781. [Online]. Available: https://aclanthology.org/2020.emnlp-main.550

[18] S. Xiao, Z. Liu, P. Zhang, and N. Muennighoff, "C-pack: Packaged resources to advance general chinese embedding," 2023.

[19] Y. Li, Z. Li, K. Zhang, R. Dan, S. Jiang, and Y. Zhang, "Chatdoctor: A medical chat model fine-tuned on a large language model meta-ai (llama) using medical domain knowledge," *Cureus*, vol. 15, no. 6, 2023.

[20] J. Ye, J. Gao, Q. Li, H. Xu, J. Feng, Z. Wu, T. Yu, and L. Kong, "Zerogen: Efficient zero-shot learning via dataset generation," *arXiv preprint arXiv:2202.07922*, 2022.

[21] Y. Yu, Y. Zhuang, J. Zhang, Y. Meng, A. J. Ratner, R. Krishna, J. Shen, and C. Zhang, "Large language model as attributed training data generator: A tale of diversity and bias," *Advances in Neural Information Processing Systems*, vol. 36, 2024.

[22] C.-Y. Lin, "Rouge: A package for automatic evaluation of summaries," in *Text summarization branches out*, 2004, pp. 74–81.

[23] K. Papineni, S. Roukos, T. Ward, and W.-J. Zhu, "Bleu: a method for automatic evaluation of machine translation," in *Proceedings of the 40th annual meeting of the Association for Computational Linguistics*, 2002, pp. 311–318.

[24] I. T. Jolliffe and J. Cadima, "Principal component analysis: a review and recent developments," *Philosophical transactions of the royal society A: Mathematical, Physical and Engineering Sciences*, vol. 374, no. 2065, p. 20150202, 2016.