Effective and Efficient Schema-aware Information Extraction Using On-Device Large Language Models

Anonymous ACL submission

Abstract

Information extraction (IE) plays a crucial role in natural language processing (NLP) by converting unstructured text into structured knowledge. Deploying computationally intensive large language models (LLMs) on resource-constrained devices for information extraction is challenging, particularly due to issues like hallucinations, limited context length, and high latency—especially when handling diverse extraction schemas. To address these challenges, we propose a two-stage information extraction approach adapted for on-device LLMs, called Dual-LoRA with Incremental Schema Caching (DLISC), which enhances both schema identification and schema-aware extraction in terms of effectiveness and efficiency. In particular, DLISC adopts an Identification LoRA module for retrieving the most relevant schemas to a given query, and an Extraction LoRA module for performing information extraction based on the previously selected schemas. To accelerate extraction inference, Incremental Schema Caching is incorporated to reduce redundant computation, substantially improving efficiency. Extensive experiments across multiple information extraction datasets demonstrate notable improvements in both effectiveness and efficiency.

1 Introduction

Information extraction (IE) is a core task in natural language processing (NLP) that aims to extract structured knowledge—such as entities, relations, and events—from unstructured text (Xu et al., 2023; Deng et al., 2024; Yang et al., 2022). Large language models (LLMs), with their powerful generalization abilities, have shown considerable promise in improving IE tasks (Xu et al., 2023; Deng et al., 2024). However, deploying LLMs on resource-constrained edge devices for information extraction is more challenging (Xu et al., 2024), including hallucinations, limitations in context length, and high latency. In particular, on-device LLMs face hallucinations due to insufficient task-specific tuning, while the need to include all extraction schemas in broad scenarios results in long inputs and high latency.

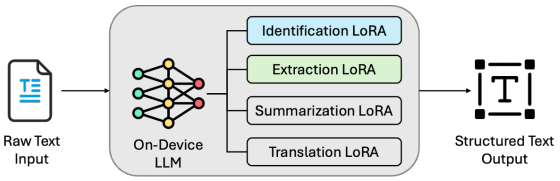

Figure 1: The LLM-Adapters architecture for deploying LLMs on edge devices with a single on-device LLM and multiple plug-in LoRA modules.

To mitigate these challenges, retrieval-augmented generation (RAG) methods have emerged as a promising solution, enhancing extraction accuracy by incorporating external knowledge (Gao et al., 2023; Li et al., 2024). For instance, Shiri et al. (2024) decompose event extraction into two subtasks: Event Detection (ED), which retrieves relevant event examples, and Event Argument Extraction (EAE), which extracts events based on the retrieved examples. In addition, Liang et al. (2025) propose Adaptive Schema-Aware Event Extraction (ASEE), a two-stage paradigm that decomposes the extraction task into schema matching and schema-augmented extraction. ASEE leverages an extensive library of event extraction schemas, adaptively retrieving relevant schemas and assembling extraction prompts to improve accuracy and scalability.

Despite the progress of RAG-based methods in information extraction, fully leveraging the unique advantages of on-device LLMs (Xu et al., 2024; Mehta et al., 2024)—such as the LLM-Adapters architecture (Hu et al., 2023)—while enhancing extraction effectiveness and efficiency remains underexplored. Figure 1 illustrates the LLM-Adapters architecture for deploying large language models on edge devices. In this design, a single on-device

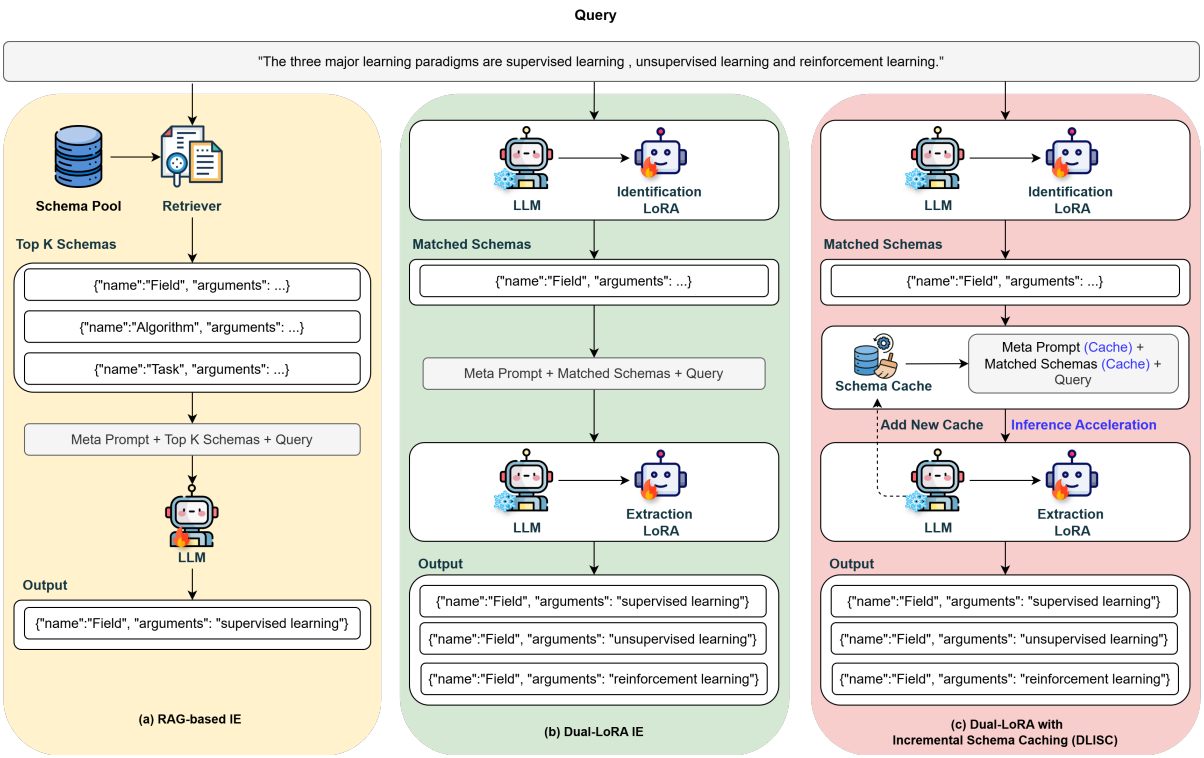

Figure 2: An illustrative comparison of (a) RAG-based IE with schema retrieval and top-K schema-aware extraction; (b) Dual-LoRA IE paradigm with schema identification and schema-aware extraction; (c) Dual-LoRA with Incremental Schema Caching (DLISC) for further enhancing inference efficiency.

LLM remains persistently loaded to maintain responsiveness, while multiple plug-in LoRA modules are selectively activated to support various task-specific adaptations with minimal resource overhead.

In this paper, we propose a two-stage information extraction approach adapted for on-device LLMs, called Dual-LoRA with Incremental Schema Caching (DLISC), which enhances both schema identification and schema-aware extraction in terms of effectiveness and efficiency. In particular, DLISC adopts an Identification LoRA module for retrieving the most relevant schemas to a given query, and an Extraction LoRA module for performing information extraction based on the previously selected schemas. To accelerate extraction inference, Incremental Schema Caching is incorporated to reduce redundant computation, notably improving efficiency.

2 Methodology

To enhance the effectiveness and efficiency of on-device information extraction, we propose a two-stage information extraction approach adapted for on-device LLMs, called Dual-LoRA with Incremental Schema Caching (DLISC). Figure 2 presents the architecture of DLISC with an Identification LoRA for identifying the most relevant schemas to the query, an Extraction LoRA for performing the information extraction with the matched schemas, and Incremental Schema Caching for accelerating the extraction inference.

Dual-LoRA Architecture. The Dual-LoRA information extraction architecture (as shown in Figure 2 (b)) follows the RAG-based two-stage paradigm (Liang et al., 2025) with an Identification LoRA ($\boldsymbol{\theta}_I$) and an Extraction LoRA ($\boldsymbol{\theta}_E$) based on the same LLM ($\boldsymbol{\theta}$). During inference, these two parameter sets are "merged" with the LLM ($\boldsymbol{\theta}$), producing two distinct LLMs ($\boldsymbol{\theta}'_I, \boldsymbol{\theta}'_E$),

$$\boldsymbol{\theta}'_I, \boldsymbol{\theta}'_E = Merge(\boldsymbol{\theta}_I, \boldsymbol{\theta}), Merge(\boldsymbol{\theta}_E, \boldsymbol{\theta}) \quad (1)$$

each serving identification and extraction functions. In addition, the Identification LoRA ($\boldsymbol{\theta}_I$) and Extraction LoRA ($\boldsymbol{\theta}_E$) can be optimized for improving the identification and extraction accuracy, respectively.

The raw text data, Query Q, is input into $\boldsymbol{\theta}'_I$ with Identification Meta Prompt M_I to identify the most relevant schemas, i.e. Matched Schemas S,

$$S = \boldsymbol{\theta}'_I(M_I + Q). \quad (2)$$

The Matched Schemas S are then concatenated with Extraction Meta Prompt M_E and Query Q as

the prompt of $\boldsymbol{\theta}'_E$,

$$R = \boldsymbol{\theta}'_E(M_E + S + Q), \tag{3}$$

where the returned structured results R are the extracted information from the raw text data Q.

Incremental Schema Caching Acceleration. To accelerating the extraction inference, we introduce Incremental Schema Caching (ISC) to the extraction process (as shown in Figure 2 (c)), inspired by the Key-Value (KV) Cache mechanism (Luohe et al., 2024) and Prompt Cache (Gim et al., 2024),

$$R = \boldsymbol{\theta}'_E(M_{E(Cache)} + S_{(Cache)} + Q), \tag{4}$$

where $M_{E(Cache)}$ and $S_{(Cache)}$ are the cached Extraction Meta Prompt and the cached Matched Schemas. Specifically, we store the Extraction Meta Prompt M_E in the cache when it first appears, and the Incremental Schema Caching (ISC) mechanism for the Matched Schemas works as follows:

- When a Matched Schema S is identified, the schema is checked if it is already in the schema cache pool.
- If the schema cache **is** found, the cached schema is directly returned and used for accelerating the extraction inference.
- If the schema cache **is not** found, the Matched Schema is concatenated in text for running the inference process, while the computed Matched Schema cache is then stored in the schema cache pool.

Overall, by decomposing the information extraction task into two stages—Identification and Extraction—the multi-LoRA structure of on-device LLMs can be fully leveraged to optimize each stage separately, thereby improving overall extraction performance. By incrementally caching previously inferenced schemas, we can avoid redundant calculations, thereby boosting the extraction inference efficiency.

3 Experiments

3.1 Experimental Settings

RAG-based Baselines. We compare our proposed DLISC approach with the following retrieval-augmented baselines (as shown in Figure 2 (a), K=5):

- **BM25** (Robertson and Zaragoza, 2009) is a probabilistic information retrieval algorithm that scores document-query relevance by weighting term frequency, inverse document frequency, and document length.
- **BGE-Reranker-V2-M3** (Chen et al., 2024) is a lightweight reranker model that possesses strong multilingual capabilities, is easy to deploy, and supports fast inference.
- **LLM-Embedder** (Zhang et al., 2024a) comprehensively support diverse retrieval augmentation scenarios for LLMs with a unified embedding model, addressing the limitations of both general-purpose and task-specific retrievers.

On-Device LLMs. We consider the following state-of-the-art on-device LLMs for information extraction:

- **Llama-3.2-1B** (Dubey et al., 2024) delivers powerful language model capabilities on edge and mobile devices with its lightweight 1B parameter model.
- **Qwen2.5-3B** (Yang et al., 2024) demonstrate exceptional performance across a wide range of tasks and benchmarks, showcasing its strength in instruction following, generating long texts, understanding structured data, and producing structured outputs.
- **TinyLlama-1.1B-Chat-v1.0** (Zhang et al., 2024b) is a lightweight conversational model based on the TinyLlama project, which aims to pretrain a 1.1 billion parameter Llama model on 3 trillion tokens and is suitable for applications with limited computational and memory resources.

Datasets. We conduct experiments on several collected datasets, including:

- CrossNER_AI (Liu et al., 2020): CrossNER is a well-known **English** open-source project in the field of NLP, specifically focusing on cross-domain Named Entity Recognition (NER). In particular, CrossNER_AI mainly focuses on the **artificial intelligence (AI)** domain.
- DuEE-Fin (Gui et al., 2024): a large-scale dataset designed for document-level Event Extraction (EE) tasks, particularly focusing on the **Chinese financial** domain.

Evaluation Metrics. We use three metrics to evaluate IE performance in terms of both effectiveness and efficiency:

- Precision – Assesses the accuracy of schema matching.
- F1 Score – Measures the overall quality of information extraction.

Method	Llama-3.2-1B	Qwen2.5-3B
BM25	0.2341	0.3968
BGE-Reranker-V2-M3	0.2341	0.3819
LLM-Embedder	0.2341	0.3819
DLISC	**0.4179**	**0.4311**

Table 1: Effectiveness of DLISC with baselines on CrossNER_AI for NER in terms of F1.

Method	Llama-3.2-1B	Qwen2.5-3B
Dual-LoRA	1.38	6.02
DLISC	**0.66**	**4.59**

Table 2: Efficiency of Dual-LoRA without caching and DLISC in seconds (s) per sample.

Method	Identification (Precision)	Extraction (F1)
BM25	0.2120	0.3220
BGE-Reranker-V2-M3	0.2580	0.5714
LLM-Embedder	0.2680	0.7484
DLISC	**0.2982**	**0.7746**

Table 3: Ablation exploration on DuEE-Fin dataset with TinyLlama-1.1B-Chat-v1.0 as the base LLM ($\boldsymbol{\theta}$). "Identification" represents the schema matching phase, and "Extraction" represents the end-to-end information extraction phase with the corresponding matched schemas.

- Latency (seconds, s) – Captures efficiency by recording average extraction time over 100 samples.

3.2 Experimental Results

Effectiveness Comparison. Table 1 presents a comparative evaluation between our proposed DLISC method and several retrieval-augmented information extraction (IE) baselines that differ in their retrieval capabilities. Specifically, the baselines incorporate three retrieval models—BM25, BGE-Reranker-V2-M3, and LLM-Embedder—while employing the same information extraction backbone model $\boldsymbol{\theta}'_E$ as used in DLISC, ensuring a fair comparison. As shown in the results, DLISC consistently outperforms all RAG-based baselines across both Llama-3.2-1B and Qwen2.5-3B, demonstrating superior extraction effectiveness. These findings highlight that DLISC's on-device IE task decomposition and schema caching strategies contribute to more accurate and robust information extraction, especially in complex or schema-rich scenarios.

Efficiency Comparison. To assess the impact of Incremental Schema Caching on extraction efficiency, we measure the end-to-end processing time (in seconds) required for information extraction over a set of 100 test samples. As shown in Table 2, we compare the baseline Dual-LoRA method without caching against our proposed DLISC approach enhanced with Incremental Schema Caching. Notably, DLISC, when implemented with both Llama-3.2-1B and Qwen2.5-3B, achieves a substantial reduction in latency, demonstrating a more efficient inference process. The results highlight that Incremental Schema Caching notably lowers the average extraction time per sample, thereby improving overall system responsiveness.

Ablation Exploration. Our proposed DLISC adopts a two-stage information extraction paradigm involving two distinct LLMs: $\boldsymbol{\theta}'_I$ for identifying the most relevant schemas, and $\boldsymbol{\theta}'_E$ for schema-aware extraction. Table 3 analyzes the contribution of each component—namely the "Identification" and "Extraction" phases—on the DuEE-Fin dataset, using TinyLlama-1.1B-Chat-v1.0 as the base LLM ($\boldsymbol{\theta}$). The results show that DLISC outperforms all three RAG-based IE baselines, achieving the highest Precision score in the Identification phase and the highest F1 score in the Extraction phase. Overall, DLISC delivers the best end-to-end performance, validating the effectiveness of the Dual-LoRA information extraction paradigm.

4 Conclusion

In this paper, we propose Dual-LoRA with Incremental Schema Caching (DLISC), a novel two-stage information extraction approach tailored for on-device LLMs. Specifically, DLISC employs an Identification LoRA module to retrieve the most relevant schemas for a given query, and an Extraction LoRA module to perform information extraction conditioned on the selected schemas. We conduct extensive experiments on multiple benchmark datasets, demonstrating that DLISC achieves state-of-the-art performance in both schema identification and schema-aware extraction when compared with three RAG-based IE baselines. To further improve inference efficiency, DLISC integrates Incremental Schema Caching, which effectively reduces redundant computation. Future work will explore integrating more fine-grained schema representations and dynamic on-demand generation to further enhance adaptability across diverse extraction tasks.

5 Limitations

Our Dual-LoRA with Incremental Schema Caching (DLISC) framework has several limitations. First, DLISC currently employs only the LoRA adapters, so it remains to be explored whether the approach can generalize to other adapter types (Hu et al., 2023), such as Prefix Tuning, Series Adapter, or Parallel Adapter, to improve adaptability and generalization. Second, we were unable to deploy on real edge devices due to computational resource constraints. Additionally, our current implementation lacks support for more complex multilingual and cross-lingual scenarios, posing additional challenges for building scalable and versatile information extraction systems. We hope to address the above limitations in the follow-up work.

References

Jianlv Chen, Shitao Xiao, Peitian Zhang, Kun Luo, Defu Lian, and Zheng Liu. 2024. Bge m3-embedding: Multi-lingual, multi-functionality, multi-granularity text embeddings through self-knowledge distillation. *Preprint*, arXiv:2402.03216.

Shumin Deng, Yubo Ma, Ningyu Zhang, Yixin Cao, and Bryan Hooi. 2024. Information extraction in low-resource scenarios: Survey and perspective. In *2024 IEEE International Conference on Knowledge Graph (ICKG)*, pages 33–49. IEEE.

Abhimanyu Dubey, Abhinav Jauhri, and Abhinav Pandey. 2024. The llama 3 herd of models. *ArXiv*, abs/2407.21783.

Yunfan Gao, Yun Xiong, Xinyu Gao, Kangxiang Jia, Jinliu Pan, Yuxi Bi, Yi Dai, Jiawei Sun, and Haofen Wang. 2023. Retrieval-augmented generation for large language models: A survey. *arXiv preprint arXiv:2312.10997*.

In Gim, Guojun Chen, Seung-seob Lee, Nikhil Sarda, Anurag Khandelwal, and Lin Zhong. 2024. Prompt cache: Modular attention reuse for low-latency inference. *Proceedings of Machine Learning and Systems*, 6:325–338.

Honghao Gui, Lin Yuan, Hongbin Ye, Ningyu Zhang, Mengshu Sun, Lei Liang, and Huajun Chen. 2024. IEPile: Unearthing large scale schema-conditioned information extraction corpus. pages 127–146, Bangkok, Thailand.

Zhiqiang Hu, Lei Wang, Yihuai Lan, Wanyu Xu, Ee-Peng Lim, Lidong Bing, Xing Xu, Soujanya Poria, and Roy Lee. 2023. LLM-adapters: An adapter family for parameter-efficient fine-tuning of large language models. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 5254–5276, Singapore. Association for Computational Linguistics.

Xiaoxi Li, Jiajie Jin, Yujia Zhou, Yuyao Zhang, Peitian Zhang, Yutao Zhu, and Zhicheng Dou. 2024. From matching to generation: A survey on generative information retrieval. *ArXiv*, abs/2404.14851.

Sheng Liang, Hang Lv, Zhihao Wen, Yaxiong Wu, Yongyue Zhang, Hao Wang, and Yong Liu. 2025. Adaptive schema-aware event extraction with retrieval-augmented generation. *arXiv preprint arXiv:2505.08690*.

Zihan Liu, Yan Xu, Tiezheng Yu, Wenliang Dai, Ziwei Ji, Samuel Cahyawijaya, Andrea Madotto, and Pascale Fung. 2020. Crossner: Evaluating cross-domain named entity recognition. In *AAAI Conference on Artificial Intelligence*.

Shi Luohe, Hongyi Zhang, Yao Yao, Z. Li, and Zhao Hai. 2024. Keep the cost down: A review on methods to optimize llm' s kv-cache consumption. *ArXiv*, abs/2407.18003.

Sachin Mehta, Mohammad Hossein Sekhavat, Qingqing Cao, Maxwell Horton, Yanzi Jin, Chenfan Sun, Iman Mirzadeh, Mahyar Najibi, Dmitry Belenko, Peter Zatloukal, et al. 2024. Openelm: An efficient language model family with open training and inference framework. *arXiv preprint arXiv:2404.14619*.

Stephen E. Robertson and Hugo Zaragoza. 2009. The probabilistic relevance framework: BM25 and beyond. *Found. Trends Inf. Retr.*, 3(4):333–389.

Fatemeh Shiri, Van Nguyen, Farhad Moghimifar, John Yoo, Gholamreza Haffari, and Yuan-Fang Li. 2024. Decompose, enrich, and extract! schema-aware event extraction using llms. *Preprint*, arXiv:2406.01045.

Derong Xu, Wei Chen, Wenjun Peng, Chao Zhang, Tong Xu, Xiangyu Zhao, Xian Wu, Yefeng Zheng, and Enhong Chen. 2023. Large language models for generative information extraction: A survey. *arXiv preprint arXiv:2312.17617*.

Jiajun Xu, Zhiyuan Li, Wei Chen, Qun Wang, Xin Gao, Qi Cai, and Ziyuan Ling. 2024. On-device language models: A comprehensive review. *arXiv preprint arXiv:2409.00088*.

An Yang, Baosong Yang, and Binyuan Hui. 2024. Qwen2 technical report. *ArXiv*, abs/2407.10671.

Yang Yang, Zhilei Wu, Yuexiang Yang, Shuangshuang Lian, Fengjie Guo, and Zhiwei Wang. 2022. A survey of information extraction based on deep learning. *Applied Sciences*, 12(19):9691.

Peitian Zhang, Zheng Liu, Shitao Xiao, Zhicheng Dou, and Jian-Yun Nie. 2024a. A multi-task embedder for retrieval augmented LLMs. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 3537–3553, Bangkok, Thailand. Association for Computational Linguistics.

Peiyuan Zhang, Guangtao Zeng, Tianduo Wang, and Wei Lu. 2024b. Tinyllama: An open-source small language model. *ArXiv*, abs/2401.02385.