UniHGKR: Unified Instruction-aware Heterogeneous Knowledge Retrievers

Dehai Min[1], Zhiyang Xu[3], Guilin Qi[1], Lifu Huang[4], Chenyu You[2]

[1]Southeast University, [2]Stony Brook University, [3]Virginia Tech, [4]UC Davis

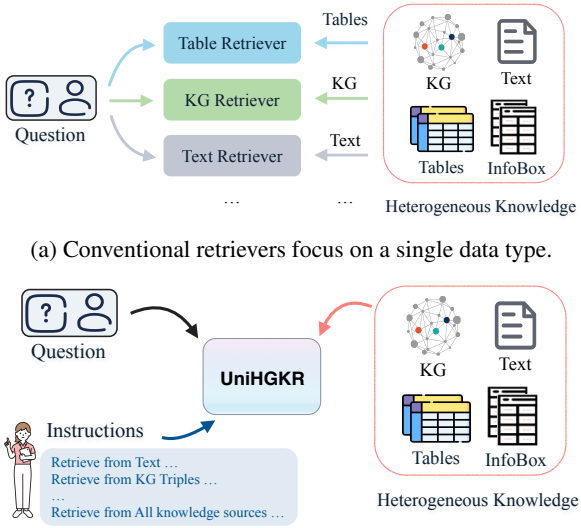
qieqiemin@gmail.com, zhiyangx@vt.edu, gqi@seu.edu.cn
lfuhuang@ucdavis.edu, chenyu.you@stonybrook.edu

Abstract

Existing information retrieval (IR) models often assume a homogeneous structure for knowledge sources and user queries, limiting their applicability in real-world settings where retrieval is inherently heterogeneous and diverse. In this paper, we introduce UniHGKR, a unified instruction-aware heterogeneous knowledge retriever that (1) builds a unified retrieval space for heterogeneous knowledge and (2) follows diverse user instructions to retrieve knowledge of specified types. UniHGKR consists of three principal stages: heterogeneous self-supervised pretraining, text-anchored embedding alignment, and instruction-aware retriever fine-tuning, enabling it to generalize across varied retrieval contexts. This framework is highly scalable, with a BERT-based version and a UniHGKR-7B version trained on large language models. Also, we introduce CompMix-IR, the first native heterogeneous knowledge retrieval benchmark. It includes two retrieval scenarios with various instructions, over 9,400 question-answer (QA) pairs, and a corpus of 10 million entries, covering four different types of data. Extensive experiments show that UniHGKR consistently outperforms state-of-the-art methods on CompMix-IR, achieving up to 6.36% and 54.23% relative improvements in two scenarios, respectively. Finally, by equipping our retriever for open-domain heterogeneous QA systems, we achieve a new state-of-the-art result on the popular ConvMix (Christmann et al., 2022b) task, with an absolute improvement of up to 5.90 points.[1]

1 Introduction

Retrieval-Augmented Generation (RAG (Lewis et al., 2020; Gao et al., 2023; Qi et al., 2024)) has become a pivotal technique for improving the faithfulness of generative large language models (LLMs (Achiam et al., 2023)). By leveraging retrievers to extract relevant knowledge from large-scale knowledge corpus, RAG effectively reduces the hallucinations often produced by LLMs (Ayala and Bechard, 2024; Muennighoff et al., 2024).



(a) Conventional retrievers focus on a single data type.

(b) UniHGKR aims to retrieval from any heterogeneous knowledge source.

Figure 1: Compared to traditional methods, UniHGKR follows user instructions to process queries and retrieves from a heterogeneous knowledge candidates pool.

Although existing information retrieval (IR) methods(Yang et al., 2024; Zhao et al., 2024) have demonstrated effectiveness in retrieving information from homogeneous knowledge corpus, where knowledge is stored in a single structure, such as tables (Kong et al., 2024) or text (BehnamGhader et al., 2024), most of these systems fail to recognize diverse user retrieval intents and retrieve heterogeneous knowledge from multiple sources. In heterogeneous IR, knowledge comes from multiple structures, making retrieval much more complex. Relying solely on homogeneous knowledge often results in partial or incomplete retrieval results, limiting the applicability of these systems to a wider range of downstream tasks (Asai et al.,

[1]Our code, datasets and model checkpoints are available at: https://github.com/ZhishanQ/UniHGKR

Proceedings of the 2025 Conference of the Nations of the Americas Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers), pages 4577–4594
April 29 - May 4, 2025 ©2025 Association for Computational Linguistics

2023; Christmann et al., 2022b). For example, a retriever specialized in table-based retrieval (Herzig et al., 2021) cannot be easily applied to downstream tasks such as question answering (QA) based on knowledge graphs (Huang et al., 2023).

In this paper, we propose the Unified Hetero-Geneous Knowledge Retriever (**UniHGKR**), a novel framework designed to retrieve information from heterogeneous knowledge corpus by following user instructions, as depicted in Figure 1. The UniHGKR framework consists of three training stages: **(1) Unified Embedding Self-Supervised Pre-training**: This stage addresses the lack of structured data in the original pretraining of the language model, laying the foundation for the creation of a unified embedding space. **(2) Text-Anchored Heterogeneous Embedding Alignment**: In this stage, natural language text that shares the same semantic content as heterogeneous data is collected, and their embeddings are aligned using contrastive learning. This process creates a unified embedding space that captures semantic information, independent of the format in which the knowledge is presented. **(3) Instruction-Aware Heterogeneous Retriever Fine-tuning**: At this final stage, the retriever is fine-tuned on heterogeneous knowledge retrieval tasks. To enhance the model's capability to follow user instructions, we introduce two specialized contrastive losses, termed 'type-balanced loss' and 'type-preferred loss', which are designed to optimize retrieval performance according to user instructions.

In addition, existing heterogeneous IR benchmarks have limited knowledge coverage (Petroni et al., 2021; Muennighoff et al., 2023). For example, studies like (Chen et al., 2021b; Zhong et al., 2022) focus only on two types of knowledge: tables and text. To address this gap, we introduce **CompMix-IR**, the first-ever benchmark for heterogeneous knowledge retrieval. CompMix-IR has over 9,400 QA pairs and a corpus of 10 million entries spanning four distinct knowledge types: Text, Knowledge Graphs (KG), Tables, and Infoboxes. Derived from the open-domain QA dataset CompMix (Christmann et al., 2024), CompMix-IR transforms this QA task into a standard IR task (as detailed in Section 3). To better reflect real-world retrieval needs, we define two distinct scenarios in this benchmark: (1) retrieving relevant evidence across all knowledge types, and (2) retrieving evidence of a specific type, as specified by user instructions. Both scenarios utilize the same evidence pool, requiring the retriever to adapt query-evidence similarity based on the instructions. This setup mirrors the complexities of real-world retrieval tasks, offering enhanced practical relevance and utility for diverse applications.

Experimental results demonstrate the effectiveness of our proposed UniHGKR over the existing methods, with relative improvements of up to 6.36% and 54.23% in two different scenarios. In addition to the BERT-based UniHGKR-base model, we also extend our framework to an LLM-based retriever and train the UniHGKR-7B model to verify scalability. Both models achieve state-of-the-art (SOTA) performance on CompMix-IR respective to their parameter scales. Furthermore, in the context of open-domain heterogeneous QA, systems equipped with UniHGKR retriever set a new SOTA on the ConvMix task (Christmann et al., 2022b), with an absolute gain of up to 5.90 points, further validating its real-world applicability.

2 Related Work

IR on Heterogeneous Knowledge. Several efforts have been make in this field, but they come with notable limitations. For example, Li et al. (2021); Kostić et al. (2021) create separate retrieval indices for different data types, retrieving them individually. This approach fails to compare relevance of evidence across knowledge sources, and maintaining multiple indices increases system complexity. On the other hand, UDT-QA (Ma et al., 2022b) introduces a verbalizer-retriever-reader framework, using a finetuned data-to-text generator (Nan et al., 2021) to convert heterogeneous scenarios into homogeneous text scenarios. However, this leads to answer coverage loss and limits downstream reader models from utilizing the structured of data, essential for tasks like KG-based and Table-based QA (Hu et al., 2023; Kweon et al., 2023). Additionally, these retrievers are typically designed for predefined single tasks, failing to accommodate users diverse retrieval needs.

QA over Heterogeneous Knowledge. Each data type has its own characteristics and provides unique benefits. Some studies explores the integration of knowledge sources to QA (Ma et al., 2022a; Min et al., 2024; You et al., 2020a,b, 2021c,a,b; Chen et al., 2021a; You et al., 2022). For instance, HybridQA (Chen et al., 2020b) and OTT-QA (Chen et al., 2021b) investigate the task of extracting answers from the combination of tables and

text. Going further, CONVINSE (Christmann et al., 2022b), Explaignn (Christmann et al., 2023) and FAITH (Jia et al., 2024) consider four knowledge sources like this paper. However, their primary focus is on the answer generation parts of the system. Their retrieval approach is a time-consuming online pipeline: identifying entity IDs in questions, then conducting online searches in Wikipedia and Wikidata (Vrandečić and Krötzsch, 2014), and finally employing BM25 (Robertson et al., 2009) to rank a small set of evidence.

3 CompMix-IR Benchmark

In this section, we provide a detailed description of the construction of CompMix-IR, the definition of retrieval scenarios, and their instruction schema.

3.1 Heterogeneous Knowledge Collection

We introduce **CompMix-IR**, the first native heterogeneous knowledge retrieval dataset, built on the CompMix dataset (Christmann et al., 2024), a recent crowdsourced open-domain QA task spanning four knowledge sources. However, the original dataset lacks a heterogeneous corpus suitable for retrieval tasks. To address this, we construct a heterogeneous knowledge corpus related to the CompMix QA set, extending it for IR tasks. Specifically, we collect and store four types of knowledge using the following methods for each question:

- **KG facts.** We use CLOCQ (Christmann et al., 2022a) to retrieve the top-1000 KG triples related to each question from the Wikidata dump. We also store the disambiguations and wikidata entities information returned by CLOCQ. This information helps us evaluate the relevance between the evidence and the question. To feed the structured data into the language model, the retrieved KG facts are linearized, with entities and relations separated by commas.
- **Text, Tables and Infoboxes.** We use the entities mentioned in questions to retrieve the corresponding Wikipedia pages. Subsequently, a parser is used to extracts natural language paragraphs (text evidence), tables, and infoboxes from the pages. Also, we utilize hyperlinks from Wikipedia pages to map the corresponding entity mentions to Wikidata IDs. This achieves the same labeling format as KG evidence. Following (Oguz et al., 2022), both tables and infoboxes are linearized using simple templates. Specifically, we concatenate the properties and values from the table using the word "is". The entity name described by the infobox and the properties and values are strung together by a comma ",", forming a text string. Additionally, Wikipedia page titles are added at the beginning of the evidence for clearer information.

Types	Avg. length	Count	Percentage
Text	19.86	5,916,596	57.74%
KG	11.40	2,214,854	21.61%
Table	20.32	1,043,105	10.18%
Infobox	11.05	1,072,440	10.47%
Sum	17.18	10,246,995	100.00%

Table 1: Statistics of CompMix-IR. 'Avg. length' refers to the average number of words.

To align with the standard IR task setup, we use automated scripts to label relevant evidence (golden labels) for each question. The relevance between the evidence and the question is of a boolean type (True/False). Specifically, if the entities in the evidence contain the answer to the question, the relevance is marked as True; otherwise, it is marked as False. Each question has at least one piece of evidence that provides the answer. The evidence retrieved for all questions in CompMix is combined into a heterogeneous knowledge pool, forming the corpus for the IR task. This corpus includes over 10 million pieces of evidence, covering knowledge about 137,808 different entities. Detailed statistics of CompMix-IR are presented in Table 1, and examples of linearized evidence and QA pair, as well as their annotation information examples, provided in Appendix A.

3.2 Retrieval Scenarios and Instructions

To address real-world heterogeneous knowledge retrieval needs, we define two distinct retrieval scenarios:

- **Scenario 1:** retrieving evidence from all types of knowledge.
- **Scenario 2:** retrieving type-specific evidence, as instructed by the user.

Both scenarios use the same evidence pool, requiring retrievers to consider not only the relevance of candidates but also whether these candidates match the data type specified in the instructions. Based on these two scenarios, we define an instruction schema (as shown in Table 2), inspired by (Asai et al., 2023; Wei et al., 2023). Users can customize retrieval by adjusting the *[domain]* and *[source]* options, where *[domain]* specifies the topic of evidence and *[source]* defines the type

Template	Given a question in the [domain] domain, retrieve relevant evidence to answer the question from the [source].
[domain] options:	books, movies, music, television series, and football
[source] options:	All Knowledge Sources, Knowledge Graph Triples, Infobox, Table, and Text
Example 1:	Given a question in the music domain, retrieve ... from Knowledge Graph Triples.
Example 2:	Given a question in the football domain, retrieve relevant ... from All Knowledge Sources.
Paraphrased 1:	For a question related to the music domain, find pertinent information from Knowledge Graph Triples.
Paraphrased 2:	For a question in the football domain, extract helpful ... to address it from All Knowledge Sources.

Table 2: Schema and examples of instructions for heterogeneous retrieval. The template contains two placeholders: [domain] and [source]. Users can select options for these based on their specific needs.

of knowledge. Instructions are categorized into five groups: I_{All}, I_{Text}, I_{KG}, I_{Table}, and I_{Info}. Here, I_{All} corresponds to our retrieval scenario 1, while the others correspond to scenario 2. Additionally, to enhance the robustness of the instructions, each instruction was rewritten into 20 different expressions with the help of GPT-4o-mini (OpenAI, 2024).

4 UniHGKR

In this section, we introduce our problem formulation and the UniHGKR framework. Our UniHGKR-base model adopts a single shared-encoder architecture, with parameters initialized from the BERT-base model (Devlin et al., 2019). The [CLS] token from the final hidden layer is trained to serve as the embedding, following (Karpukhin et al., 2020; Xiao et al., 2022a).

4.1 Problem Formulation

Given a vase candidate pool of heterogeneous evidence $\mathcal{E}$, defined as: $\mathcal{E} = \bigcup_{\tau \in \mathcal{H}} \mathcal{E}_\tau$, where $\mathcal{H} = \{\text{Text, Info, Table, KG}\}$ represents the set of evidence types. For each type τ, $\mathcal{E}_\tau = \{e_\tau^i\}_{i=1}^{N_\tau}$ is the set of evidence of type τ. The problem of retrieval with instructions is to find evidence $e \in \mathcal{E}$ that is relevant to q according to the instruction I. The instruction and question are concatenated as $\tilde{q} = [I; q]$, and the evidence e is encoded into embedding vectors by a shared encoder, denoted as Enc. The similarity between $\tilde{q}$ and e, is calculated as follows:

$$f(\tilde{q}, e) = \text{Enc}\,(\tilde{q})^\top \,\text{Enc}(e), \tag{1}$$

where $^\top$ denotes the transpose operation. The retriever returns the top k evidence with the highest similarity as the retrieval results.

4.2 UniHGKR Framework

An overview of our framework is presented in Figure 2, which comprises the following three training stages:

Stage 1: Unified Embedding Self-Supervised Pretraining. Pretrained Language Models (PLMs) are primarily trained on text, making them ineffective at generating embeddings for heterogeneous data, which is critical for IR tasks (Li et al., 2022, 2023b). To this end, we design this stage to train PLMs with a token masking reconstruction task on heterogeneous data-text pairs as inputs. Specifically, we first construct a set of data-text pairs based on the CompMix-IR corpus with the help of LLMs, as illustrated in Figure 3:

$$\mathcal{D} = \left\{ \langle d_i, t_i \rangle \mid d_i \in \hat{\mathcal{E}},\ t_i = \mathcal{F}(d_i) \right\}_{i=1}^{N}, \tag{2}$$

where, $\hat{\mathcal{E}} = \mathcal{E}_{\text{KG}} \cup \mathcal{E}_{\text{Table}} \cup \mathcal{E}_{\text{Info}}$, $\mathcal{F}$ is the data-to-text generator, which in our setting is GPT-4o-mini. The d_i is the linearized structured data, and the text t_i is a well-written natural language sentences with the same semantic information as d_i. At this stage, they are concatenated to form training inputs. This approach enables the model to accept input sequences in heterogeneous formats as self-supervised signals. Furthermore, d_i and t_i can serve as distant supervision signals for each other, providing an indirect supervisory signal that enhances the model's learning from heterogeneous inputs (Sun et al., 2021; Mintz et al., 2009). We adopt the token masking reconstruction task from RetroMAE (Xiao et al., 2022b): an additional single-layer Transformer (Vaswani et al., 2017) as a temporary decoder with a 50% masking ratio, while our model serving as the encoder with a 15% masking ratio. The training objective is:

$$\min_\theta \sum_{x \in \mathcal{X}} -\log \text{Dec}\left(x \mid \text{Enc}\left(\tilde{x}; \theta\right); \theta\right). \tag{3}$$

Here, x represents the original clean input, and $\tilde{x}$ denotes the masked input. After this stage training is completed, only the weights of the encoder are retained for subsequent training.

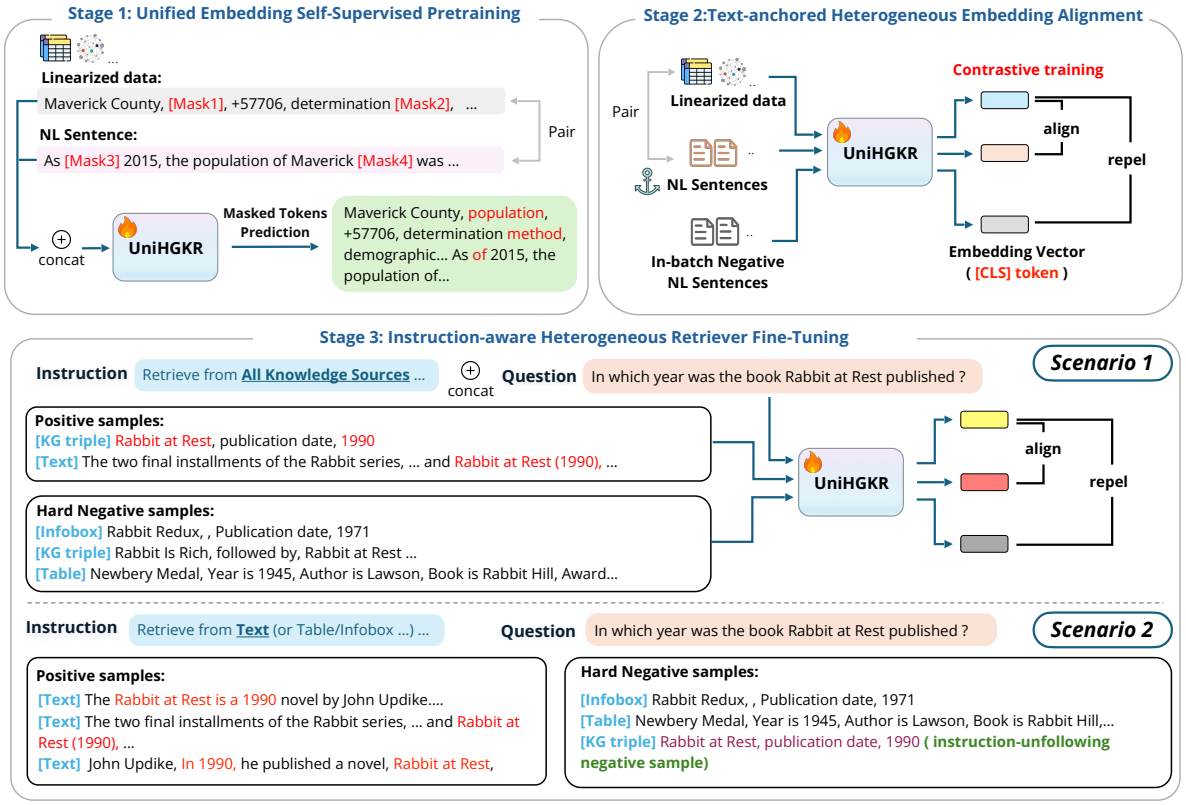


Figure 2: Illustration of our UniHGKR training framework.

Stage 2: Text-anchored Heterogeneous Embedding Alignment. Given that user instructions and questions are typically in text form, we further leverage the collected data-text pairs to optimize the embedding space anchored in text embedding representations. We apply contrastive learning (Chen et al., 2020a) to align the embedding of structured data d_i and text t_i that convey the same semantic information but differ in expression. Meanwhile, we repel embedding with different semantic information using in-batch negative samples B^- (samples that do not share semantic similarity with d_i) (Sohn, 2016). This results in a unified embedding space focused on semantic information rather than the form of knowledge representation. The training objective is to minimize:

$$\sum_{\langle d_i,t_i\rangle\in\mathcal{D}} -\log \frac{e^{f(d_i,t_i)/\tau}}{e^{f(d_i,t_i)/\tau}+\sum\limits_{b^-\in B^-} e^{f(d_i,b^-)/\tau}}, \tag{4}$$

where $f()$ is a similarity function and τ is the temperature parameter.

Stage 3: Instruction-aware Heterogeneous Retriever Fine-Tuning. In this stage, we fine-tune our retriever on the heterogeneous knowledge retrieval task. For each question q and its golden evidence e^+, we generate two training samples: (I_{All}, q, e^+) and (I_λ, q, e^+), where λ is the data type of the positive sample e^+. Additionally, we use the BGE model (Xiao et al., 2024) to mine hard negative samples set, denoted as E^-. For the contrastive training loss $\mathcal{L}$:

$$\begin{aligned}\mathcal{L} &= -\log \frac{e^{f(\tilde{q},e^+)/\tau}}{e^{f(\tilde{q},e^+)/\tau}+\sum_{d^-\in E^-} e^{f(\tilde{q},e^-)/\tau}} \\ &= \underbrace{-f(\tilde{q},e^+)/\tau}_{\mathcal{L}_{\text{align}}} + \underbrace{\log\left(e^{f(\tilde{q},e^+)/\tau}+\mathcal{L}_{\text{repel}}\right)}_{\mathcal{L}_{\text{uniformity}}}\end{aligned} \tag{5}$$

Here, $\mathcal{L}_{\text{align}}$ is the alignment loss encouraging higher similarity between the query and the positive evidence. Meanwhile, $\mathcal{L}_{\text{uniformity}}$ denotes the uniformity loss applied over all samples, aiming to push the query away from negative samples (Wang and Isola, 2020). We can simplify $\mathcal{L}_{\text{repel}}$:

$$\mathcal{L}_{\text{repel}} = \sum_{\tilde{\lambda}\in\mathcal{H}} \sum_{e^-_{\tilde{\lambda}}\in E^-_{\tilde{\lambda}}} e^{f(\tilde{q},e^-_{\tilde{\lambda}})/\tau} \tag{6}$$

where $\mathcal{H}=\{\text{Text, Info, Table, KG}\}$, and $E^-_{\tilde{\lambda}}$ is the set of hard negative samples of type $\tilde{\lambda}$. We define: $k_{\tilde{\lambda}} = |E^-_{\tilde{\lambda}}|, \tilde{\lambda}\in\mathcal{H}$ to represent the number of negative samples for each type.

To enhance the model's ability to follow user instructions, we design distinct contrastive losses:

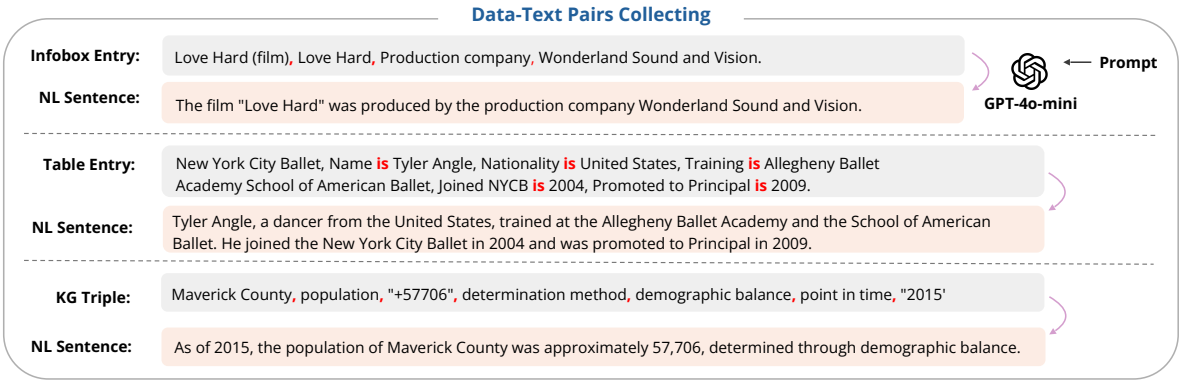


Figure 3: Illustration of Data-Text Pair Collection. The bold red **is** and the comma **,** are used in concatenation template when linearizing structured data. The prompts used for GPT-4o-mini can be found in Appendix B.

a type-balanced loss $\mathcal{L}_{\text{balanced}}$ for training samples wtih instruction I_{All} (Scenario 1), and a type-preferred loss $\mathcal{L}_{\text{preferred}}$ for training samples with instruction I_λ (Scenario 2). Specifically, for type-balanced loss $\mathcal{L}_{\text{balanced}}$, we make $k_{\text{Text}} \approx k_{\text{Info}} \approx k_{\text{Table}} \approx k_{\text{KG}}$ depend on their numbers in E^-. In contrast, for type-preferred loss $\mathcal{L}_{\text{preferred}}$, in order to make the model learn the priority of evidence with specified-type λ, we deliberately make k_λ significantly lower than the quantity of other types. For example, when a training sample with instructions I_{Table}, we set $k_{\text{Text}} \approx k_{\text{Info}} \approx k_{\text{KG}} > k_{\text{Table}} = 0$, by filtering out e^-_{Table} from E^-. By adjusting k_λ, the training samples with I_λ have fewer negative samples of type λ, thereby forming a preference for evidence of type λ in the global heterogeneous candidate pool. Since we also use in-batch negative samples B^- during training, the model can still learn to repel e^-_λ, which are of the correct type but irrelevant evidences. Additionally, we also add a small number of instruction-unfollowing negative samples, which are related to q but not of the type λ, to encourage the model to decrease their similarity with $\tilde{q}$.

5 Experimental Methodology

In our main experiments, we train and evaluate retrievers on the CompMix-IR, following the train, dev, and test set divisions in CompMix.

5.1 Baselines

Zero-shot SoTA Retriever. Referring to the MTEB leaderboard[2], we select some top-ranking and SOTA models as baselines, including Mpnet (Song et al., 2020), Contriever (Izacard et al., 2022), DPR (Karpukhin et al., 2020), GTR-T5 (Ni et al., 2022), SimLM (Wang et al., 2023), BGE (Xiao et al., 2024), and Instructor (Su et al., 2023). For Mpnet, we use the strong version[3] released by Sentence-Transformers (Reimers and Gurevych, 2019). Additionally, we evaluate the classic sparse retriever BM25 (Robertson et al., 2009). For retrievers that undergo instruction fine-tuning (see Table 3), we use the instructions provided in their respective papers for evaluation.

Fine-tuned Baselines. We follow the verbalizer-retriever approach from UDT-QA (Ma et al., 2022b) to fine-tune a BERT-base model, serving as the UDT retriever. Since UDT focuses on homogeneous textual representations of heterogeneous data, we replace d_i with t_i from the data-text pairs $\mathcal{D}$ during its training and evaluation, ensuring this model only interacts with the natural language corpus. This also means that in our experiments, we fine-tune the UDT-retriever baseline using exactly the same GPT-4o-mini synthesized data-text pairs $\mathcal{D}$ as utilized by our UniHGKR. For comparison, we also fine-tune a BERT-base model on the original CompMix-IR. Additionally, we finetune a DPR model using the UniK-QA method (Oguz et al., 2022), serving as the UniK retriever. All fine-tuning uses the same positive and hard negative samples as UniHGKR. For baseline models lacking instruction-following capabilities, we input only the query across all retrieval scenarios to ensure optimal performance.

5.2 Evaluation Metrics

For retrieval scenario 1, we employ common metrics in the IR task: Hit@K (K=5,10,100) and

[2]https://huggingface.co/spaces/mteb/leaderboard

[3]https://huggingface.co/sentence-transformers/all-mpnet-base-v2

			Retrieval Scenario 1 (instruction I_{All})				Retrieval Scenario 2 (instruction I_{τ})			
Method	**Size**	**Ins**	**Hit@5**	**Hit@10**	**Hit@100**	**MRR@100**	**KG-Hit**	**Text-Hit**	**Table-Hit**	**Info-Hit**
BM25	-	✗	11.51	17.40	52.39	8.54	24.20	34.55	8.50	19.79
DPR	109M	✗	24.89	36.32	78.76	17.51	49.13	63.68	15.63	41.57
Mpnet	109M	✗	26.23	37.99	82.67	18.46	63.02	61.11	18.96	52.1
GTR-T5-base	110M	✗	24.46	36.54	80.32	16.73	57.78	59.8	22.87	46.09
Contriever	109M	✗	28.58	40.70	83.79	20.07	62.26	63.86	18.63	55.64
SimLM	109M	✗	25.11	37.08	80.61	17.68	59.59	59.01	17.69	52.06
Instructor-base	110M	✓	24.86	36.22	81.55	17.80	65.63	50.25	16.82	53.36
Instructor-large	336M	✓	25.98	36.87	81.51	18.54	68.78	44.61	17.11	53.98
BGE	109M	✓	26.66	39.04	84.15	19.40	68.42	57.96	22.58	56.58
BERT-finetuned	109M	✗	24.46	35.38	78.51	17.04	57.63	54.67	17.55	48.41
UDT-retriever	109M	✗	24.96	35.49	76.52	18.24	66.10	62.48	25.90	57.05
UniK-retriever	109M	✗	30.68	43.42	85.20	21.22	67.40	63.21	26.74	56.04
UniHGKR-base	109M	✓	**32.38**	**45.55**	**85.75**	**22.57**	**75.43**	**70.30**	**41.24**	**66.21**
▲ Relative gain			+5.54%	+4.91%	+0.65%	+6.36%	+9.67%	+10.08%	+54.23%	+16.06%

Table 3: the experimental results for the two retrieval scenarios on CompMix-IR. The relative gain is calculated based on the performance of UniHGKR-base compared to the best baseline, highlighted by underlines.

MRR@K (Mean Reciprocal Rank, K=100) to evaluate model performance (Zhao et al., 2024). More detailed descriptions are provided in Appendix C. For scenario 2, which uses type-specified instructions I_{τ}, where type $\tau \in \mathcal{H}$, we introduce the metric Type-Hit (Type-Hit@100), indicate whether relevant evidence of the correct type is included in the top 100 retrieval result.

5.3 Implementation Details.

In our experiments, all contrastive training utilizes in-batch negatives across GPU devices. We utilize the maximum batch size that the GPU memory can fit and conduct all our training experiments on 8 A800-80GB GPUs. In the training stage 3, each training sample has a group size of 16, which includes 1 positive sample and 15 hard negative samples. More detailed training settings can be found in Appendix D.

6 Evaluation Results

In this section, we focus on comparing and discussing the performance of UniHGKR with baselines on heterogeneous retrieval tasks and the application of UniHGKR models in the open-domain QA task. We also explore the robustness and zero-shot performance of UniHGKR in Appendix E.

6.1 Main Results

Table 3 presents the retrieval performance of various models on the CompMix-IR test set. Our UniHGKR model outperforms all baselines in both scenarios, with a maximum relative improvement of 6.36% in scenario 1 and 54.23% in scenario 2, demonstrating its effectiveness in heterogeneous knowledge retrieval. Notably, powerful open-source retrievers like BGE (trained on over 200 million high-quality text pairs, (Xiao et al., 2024)) only achieve an MRR@100 below 20.0 in scenario 1 and a Table-Hit of 22.58 in scenario 2, highlighting the challenges of our constructed benchmark. Although the UDT-retriever shows significant improvement over its counterpart model, BERT-finetuned, in Scenario 2, the improvement in Scenario 1 is minimal. Also, it is clearly inferior to our UniHGKR-base, which was trained on the same synthetic data from GPT-4o-mini. Moreover, UniK-retriever, fine-tuned the DPR model on CompMix-IR, performs well across several metrics but is suboptimal on structured data (like Table and Infobox) in scenario 2. In contrast, our UniHGKR shows the greatest improvements on metrics where existing methods struggle, particularly in retrieving structured knowledge in scenario 2. This indicates that our three-stage training approach not only creates an effective representation space for heterogeneous knowledge retrieval but also excels at following diverse user instructions.

6.2 Ablation Study

In this subsection, we conduct ablation studies to examine the roles of different training stages and components in UniHGKR for heterogeneous knowledge retrieval. Table 4 presents the performance of various UniHGKR variants, obtained by removing specific components or a particular training stage. Results show that removing any training stage or component leads to a significant drop in performance. For retrieval scenario 1, training stage 1 (Unified Embedding Self-Supervised

	Retrieval Scenario 1 (instruction I_{All})				Retrieval Scenario 2 (instruction $I_{\mathcal{T}}$)			
Method	**Hit@5**	**Hit@10**	**Hit@100**	**MRR@100**	**KG-Hit**	**Text-Hit**	**Table-Hit**	**Info-Hit**
UniHGKR-base	**32.38**	**45.55**	**85.75**	**22.57**	**75.43**	**70.30**	**41.24**	**66.21**
w/o training stage 1 (pretrain)	29.78	42.80	84.88	21.54	72.97	68.60	34.01	60.06
w/o NL sentence during stage 1	31.30	44.36	85.46	21.83	74.07	69.01	40.13	65.16
w/o training stage 2 (alignment)	31.41	45.02	85.14	21.92	74.75	70.01	37.99	66.04
In the training stage 3 (finetune)								
w/o type-preferred loss $\mathcal{L}_{\text{preferred}}$	31.77	44.97	85.24	22.27	73.26	69.90	33.86	61.22
w/o instructions and $\mathcal{L}_{\text{preferred}}$	31.98	44.39	85.24	22.18	68.20	65.59	29.20	58.65
w/o rewritten instructions	31.11	44.14	84.48	21.86	67.66	65.16	27.82	57.96

Table 4: The results of the ablation study for the UniHGKR-base. We use blue color to indicate the largest decrease.

	Retrieval Scenario 1 (instruction I_{All})				Retrieval Scenario 2 (instruction $I_{\mathcal{T}}$)			
Method	**Hit@5**	**Hit@10**	**Hit@100**	**MRR@100**	**KG-Hit**	**Text-Hit**	**Table-Hit**	**Info-Hit**
UniHGKR-base	32.38	45.55	85.75	22.57	75.43	70.30	41.24	66.21
E5-mistral-7B	31.3	43.49	83.36	22.97	69.03	41.46	33.03	62.92
LLARA-passage	37.45	51.59	86.61	26.11	68.23	70.48	37.88	60.64
LLARA-finetuned	42.19	55.35	87.81	30.83	74.38	69.86	36.40	64.40
UniHGKR-7B	**49.78**	**59.23**	**88.21**	**38.20**	**81.80**	**76.05**	**49.57**	**73.88**
▲Relative gain	+17.99%	+7.01%	+0.46%	+23.91%	+9.98%	+7.90%	+30.86%	+14.72%

Table 5: Retrieval performances of UniHGKR-7B and LLM-based retrievers baselines. The relative gain is calculated based on the performance of UniHGKR-7B compared to the best baseline, highlighted by underlines.

Pretraining) is crucial, while for scenario 2, both rewritten (paraphrased) instructions and instruction-aware type-preferred loss $\mathcal{L}_{\text{preferred}}$ are key. Removing them will result in a performance drop of up to 13.42 points in the Table-Hit metric. Additionally, we present some extra ablation studies in Appendix F, such as exploring the role of different instructions in retrieving from specific sources and the performance gains of different training stages in an unsupervised setting.

6.3 Extending UniHGKR to LLM Retrievers

Recent works, such as E5-mistral-7B (Wang et al., 2024) and LLARA (Li et al., 2023a), have explored converting decoder-only LLMs into dense retrievers, leveraging their extensive pre-trained knowledge to achieve improvements on various IR tasks. Our UniHGKR framework is plug-and-play and can seamlessly adapt to training LLM retrievers by adjusting the training objectives. To demonstrate this, we adapt UniHGKR framework to train our UniHGKR-7B retrievers based on the LLARA architecture. More adaptation details are in Appendix G.

Table 5 presents the evaluation results of UniHGKR-7B alongside other LLM-based baselines, including E5-mistral-7B, LLARA-passage (LLARA-pretrain fine-tuned on MS MARCO passage), and LLARA-finetuned (LLARA-pretrain fine-tuned on CompMix-IR). We can observe that our UniHGKR-7B significantly outperforms the LLM-based baselines and UniHGKR-base, achieving SOTA performance on all metrics across two scenarios. In particular, it achieves a 23.91% relative improvement on the MRR@100 metric in Scenario 1 and reaches 49.57 on the Table-Hit in Scenario 2. These results further validate the effectiveness and scalability of our UniHGKR method, as well as the potential of LLMs as retrievers.

6.4 Employing UniHGKR on QA systems

In this section, we explore the application of UniHGKR retrievers in open-domain QA systems over heterogeneous sources. We select a popular task, ConvMix (Christmann et al., 2022b), which is a conversational format variant of the CompMix. This task is more challenging because it requires systems to consider both the current turn's question and the dialogue history. Baseline models such as QuReTeC (Voskarides et al., 2020), CONVINSE (Christmann et al., 2022b), and EXPLAIGNN (Christmann et al., 2023), along with their results, are sourced from the ConvMix leaderboard[4]. Note that in the QA system experiment, we replace the entire retrieval component (e.g., CLOCQ+BM25) of the baseline with our UniHGKR model, not just BM25. The retrieval component of EXPLAIGNN and CONVINSE can be seen as a combination of coarse retrieval (CLOCQ) and re-ranking (BM25). All baseline methods and

[4] https://convinse.mpi-inf.mpg.de/

Methods	Retriever	Reader	P@1	MRR
BM25+FiD	BM25	FiD	25.3	27.5
QuReTeC	QuReTeC	FiD	28.2	28.9
CONVINSE	CLOCQ+BM25	FiD	34.3	37.8
EXPLAIGNN	CLOCQ+BM25	GNN	40.6	47.1
Ours	**UniHGKR-base**	FiD	42.4	46.6
▲Abs. gain			+8.10	+8.80
	UniHGKR-7B	FiD	**46.5**	**51.4**
▲Abs. gain			+12.20	+13.60
▲SOTA gain			**+5.90**	**+4.30**

Table 6: The QA performance of systems using the UniHGKR retriever and baselines on the ConvMix dataset. 'Abs. gain' represents the absolute improvement brought by the retriever under the same Reader setting (compared to CONVINSE). 'SOTA gain' indicates the absolute improvement over the previous SOTA system.

our UniHGKR use the same corpus to ensure a fair comparison. In the reasoning part after retrieval, we follow CONVINSE and use Fusion-in-Decoder (FiD) (Izacard and Grave, 2021) as the reader. We input the top 100 relevant evidences returned by the retriever into the reader for inference. Then we evaluate the output of the reader as the performance of the QA system using the same metrics as baselines: P@1 (Precision at 1) and MRR.

As shown in Table 6, by replacing retrievers with our UniHGKR models in baseline systems, we observe significant improvements in QA performance. Specifically, compared to CONVINSE, which uses the same reader FiD as we do, using UniHGKR-base as the retriever achieves an absolute improvement of up to 8.80 points in MRR, while UniHGKR-7B achieves an improvement of up to 13.60 points in MRR. Compared to the current SOTA system, EXPLAIGNN, which uses a graph neural network (GNN) as a reader, our system surpasses it by up to 4.30 points in MRR and 5.90 points in P@1, setting a new SOTA performance for the ConvMix dataset. These results further validate the effectiveness of UniHGKR and also indicate that the retrieval component is a significant factor limiting the performance of current open-domain QA systems on heterogeneous data.

7 Conclusion

In this paper, we introduced UniHGKR, an instruction-aware unified heterogeneous knowledge retriever. First, we constructed CompMix-IR, the first heterogeneous information retrieval task dataset containing a corpus of over 10 million entries across four heterogeneous data types. Then, we defined two different heterogeneous information retrieval scenarios to meet the diverse retrieval needs of real-world users. We designed the UniHGKR framework with three training stages. Our experiments showed that UniHGKR achieved state-of-the-art performance on CompMix-IR benchmarks, both with the 110M BERT-based retriever and the 7B LLM-based retriever. Applying our UniHGKR retrievers can significantly enhance the performance of heterogeneous QA systems, achieving new SOTA results on the ConvMix dataset.

8 Limitations

In our study, the CompMix-IR dataset is primarily sourced from Wikidata knowledge graphs and Wikipedia, including infoboxes, tables, and text, but it is limited to five domains: books, movies, music, television series, and football. This may restrict the model's generalization capabilities. Additionally, while UniHGKR incorporates diverse user instructions, it does not cover all scenarios in heterogeneous information retrieval. For instance, users might want to instruct the retriever to return a combination of evidence from multiple knowledge sources, such as text and tables, or a mix of KG triples, tables, and text, as noted in (Christmann et al., 2022b). Exploring these user-defined combinations remains an area for future work. In addition, more modalities such as image, audio and interleaved image and text (Xu et al., 2024b) can be considered and incorporated in the retrieving process of UniHGKR in future. We will open-source our instruction set, CompMix-IR corpus, and UniHGKR model and code, encouraging the community to contribute more retrieval tasks with large-scale human-written instructions (Xu et al., 2024a) to assess whether broader instruction coverage enhances performance.

References

Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, et al. 2023. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*.

Akari Asai, Timo Schick, Patrick Lewis, Xilun Chen, Gautier Izacard, Sebastian Riedel, Hannaneh Hajishirzi, and Wen-tau Yih. 2023. Task-aware retrieval with instructions. In *Findings of the Association for Computational Linguistics: ACL 2023*, pages 3650–3675.

Orlando Ayala and Patrice Bechard. 2024. Reducing hallucination in structured outputs via retrieval-augmented generation. In *Proceedings of the 2024*

Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 6: Industry Track), pages 228–238.

Parishad BehnamGhader, Vaibhav Adlakha, Marius Mosbach, Dzmitry Bahdanau, Nicolas Chapados, and Siva Reddy. 2024. Llm2vec: Large language models are secretly powerful text encoders. *arXiv preprint arXiv:2404.05961*.

Nuo Chen, Chenyu You, and Yuexian Zou. 2021a. Self-supervised dialogue learning for spoken conversational question answering. *arXiv preprint arXiv:2106.02182*.

Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. 2020a. A simple framework for contrastive learning of visual representations. In *International conference on machine learning*, pages 1597–1607. PMLR.

Wenhu Chen, Ming-Wei Chang, Eva Schlinger, William Yang Wang, and William W. Cohen. 2021b. Open question answering over tables and text. In *International Conference on Learning Representations*.

Wenhu Chen, Hanwen Zha, Zhiyu Chen, Wenhan Xiong, Hong Wang, and William Yang Wang. 2020b. Hybridqa: A dataset of multi-hop question answering over tabular and textual data. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 1026–1036.

Philipp Christmann, Rishiraj Saha Roy, and Gerhard Weikum. 2022a. Beyond ned: fast and effective search space reduction for complex question answering over knowledge bases. In *Proceedings of the fifteenth ACM international conference on web search and data mining*, pages 172–180.

Philipp Christmann, Rishiraj Saha Roy, and Gerhard Weikum. 2022b. Conversational question answering on heterogeneous sources. In *Proceedings of the 45th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 144–154.

Philipp Christmann, Rishiraj Saha Roy, and Gerhard Weikum. 2023. Explainable conversational question answering over heterogeneous sources via iterative graph neural networks. In *Proceedings of the 46th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 643–653.

Philipp Christmann, Rishiraj Saha Roy, and Gerhard Weikum. 2024. Compmix: A benchmark for heterogeneous question answering. In *Companion Proceedings of the ACM on Web Conference 2024*, pages 1091–1094.

Yung-Sung Chuang, Rumen Dangovski, Hongyin Luo, Yang Zhang, Shiyu Chang, Marin Soljačić, Shang-Wen Li, Scott Yih, Yoon Kim, and James Glass. 2022. Diffcse: Difference-based contrastive learning for sentence embeddings. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 4207–4218.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

Tianyu Gao, Xingcheng Yao, and Danqi Chen. 2021. SimCSE: Simple contrastive learning of sentence embeddings. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 6894–6910, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

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*.

Faegheh Hasibi, Fedor Nikolaev, Chenyan Xiong, Krisztian Balog, Svein Erik Bratsberg, Alexander Kotov, and Jamie Callan. 2017. Dbpedia-entity v2: a test collection for entity search. In *Proceedings of the 40th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 1265–1268.

Jonathan Herzig, Thomas Mueller, Syrine Krichene, and Julian Eisenschlos. 2021. Open domain question answering over tables via dense retrieval. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 512–519.

Edward J Hu, yelong shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. 2022. LoRA: Low-rank adaptation of large language models. In *International Conference on Learning Representations*.

Nan Hu, Yike Wu, Guilin Qi, Dehai Min, Jiaoyan Chen, Jeff Z Pan, and Zafar Ali. 2023. An empirical study of pre-trained language models in simple knowledge graph question answering. *World Wide Web*, 26(5):2855–2886.

Xiang Huang, Sitao Cheng, Yiheng Shu, Yuheng Bao, and Yuzhong Qu. 2023. Question decomposition tree for answering complex questions over knowledge bases. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 37, pages 12924–12932.

Gautier Izacard, Mathilde Caron, Lucas Hosseini, Sebastian Riedel, Piotr Bojanowski, Armand Joulin, and

Edouard Grave. 2022. Unsupervised dense information retrieval with contrastive learning. *Transactions on Machine Learning Research*.

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

Zhen Jia, Philipp Christmann, and Gerhard Weikum. 2024. Faithful temporal question answering over heterogeneous sources. In *Proceedings of the ACM on Web Conference 2024*, pages 2052–2063.

Ziyan Jiang, Xueguang Ma, and Wenhu Chen. 2024. Longrag: Enhancing retrieval-augmented generation with long-context llms. *arXiv preprint arXiv:2406.15319*.

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

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.

Kezhi Kong, Jiani Zhang, Zhengyuan Shen, Balasubramaniam Srinivasan, Chuan Lei, Christos Faloutsos, Huzefa Rangwala, and George Karypis. 2024. Opentab: Advancing large language models as open-domain table reasoners. In *The Twelfth International Conference on Learning Representations*.

Bogdan Kostić, Julian Risch, and Timo Möller. 2021. Multi-modal retrieval of tables and texts using tri-encoder models. In *Proceedings of the 3rd Workshop on Machine Reading for Question Answering*, pages 82–91.

Sunjun Kweon, Yeonsu Kwon, Seonhee Cho, Yohan Jo, and Edward Choi. 2023. Open-wikitable: Dataset for open domain question answering with complex reasoning over table. In *Findings of the Association for Computational Linguistics: ACL 2023*, pages 8285–8297.

Tom Kwiatkowski, Jennimaria Palomaki, Olivia Redfield, Michael Collins, Ankur Parikh, Chris Alberti, Danielle Epstein, Illia Polosukhin, Jacob Devlin, Kenton Lee, et al. 2019. Natural questions: a benchmark for question answering research. *Transactions of the Association for Computational Linguistics*, 7:453–466.

Patrick S. H. Lewis, Ethan Perez, Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich Küttler, Mike Lewis, Wen-tau Yih, Tim Rocktäschel, Sebastian Riedel, and Douwe Kiela. 2020. Retrieval-augmented generation for knowledge-intensive NLP tasks. *CoRR*, abs/2005.11401.

Alexander Hanbo Li, Patrick Ng, Peng Xu, Henghui Zhu, Zhiguo Wang, and Bing Xiang. 2021. Dual reader-parser on hybrid textual and tabular evidence for open domain question answering. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 4078–4088.

Chaofan Li, Zheng Liu, Shitao Xiao, and Yingxia Shao. 2023a. Making large language models a better foundation for dense retrieval. *arXiv preprint arXiv:2312.15503*.

Xiaonan Li, Yeyun Gong, Yelong Shen, Xipeng Qiu, Hang Zhang, Bolun Yao, Weizhen Qi, Daxin Jiang, Weizhu Chen, and Nan Duan. 2022. Coderetriever: A large scale contrastive pre-training method for code search. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 2898–2910.

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

Zheng Liu, Shitao Xiao, Yingxia Shao, and Zhao Cao. 2023. Retromae-2: Duplex masked auto-encoder for pre-training retrieval-oriented language models. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2635–2648.

Kaixin Ma, Hao Cheng, Xiaodong Liu, Eric Nyberg, and Jianfeng Gao. 2022a. Open-domain question answering via chain of reasoning over heterogeneous knowledge. In *Findings of the Association for Computational Linguistics: EMNLP 2022*, pages 5360–5374.

Kaixin Ma, Hao Cheng, Xiaodong Liu, Eric Nyberg, and Jianfeng Gao. 2022b. Open domain question answering with a unified knowledge interface. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1605–1620.

Macedo Maia, Siegfried Handschuh, André Freitas, Brian Davis, Ross McDermott, Manel Zarrouk, and Alexandra Balahur. 2018. Www'18 open challenge: financial opinion mining and question answering. In *Companion proceedings of the the web conference 2018*, pages 1941–1942.

Dehai Min, Nan Hu, Rihui Jin, Nuo Lin, Jiaoyan Chen, Yongrui Chen, Yu Li, Guilin Qi, Yun Li, Nijun Li, and Qianren Wang. 2024. Exploring the impact of table-to-text methods on augmenting LLM-based question answering with domain hybrid data. In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 6: Industry*

Track), pages 464–482, Mexico City, Mexico. Association for Computational Linguistics.

Mike Mintz, Steven Bills, Rion Snow, and Dan Jurafsky. 2009. Distant supervision for relation extraction without labeled data. In *Proceedings of the Joint Conference of the 47th Annual Meeting of the ACL and the 4th International Joint Conference on Natural Language Processing of the AFNLP*, pages 1003–1011.

Niklas Muennighoff, Hongjin Su, Liang Wang, Nan Yang, Furu Wei, Tao Yu, Amanpreet Singh, and Douwe Kiela. 2024. Generative representational instruction tuning. *arXiv preprint arXiv:2402.09906*.

Niklas Muennighoff, Nouamane Tazi, Loic Magne, and Nils Reimers. 2023. Mteb: Massive text embedding benchmark. In *Proceedings of the 17th Conference of the European Chapter of the Association for Computational Linguistics*, pages 2014–2037.

Linyong Nan, Dragomir Radev, Rui Zhang, Amrit Rau, Abhinand Sivaprasad, Chiachun Hsieh, Xiangru Tang, Aadit Vyas, Neha Verma, Pranav Krishna, et al. 2021. Dart: Open-domain structured data record to text generation. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 432–447.

Tri Nguyen, Mir Rosenberg, Xia Song, Jianfeng Gao, Saurabh Tiwary, Rangan Majumder, and Li Deng. 2017. MS MARCO: A human-generated MAchine reading COmprehension dataset.

Jianmo Ni, Chen Qu, Jing Lu, Zhuyun Dai, Gustavo Hernandez Abrego, Ji Ma, Vincent Zhao, Yi Luan, Keith Hall, Ming-Wei Chang, et al. 2022. Large dual encoders are generalizable retrievers. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 9844–9855.

Barlas Oguz, Xilun Chen, Vladimir Karpukhin, Stan Peshterliev, Dmytro Okhonko, Michael Schlichtkrull, Sonal Gupta, Yashar Mehdad, and Scott Yih. 2022. Unik-qa: Unified representations of structured and unstructured knowledge for open-domain question answering. In *Findings of the Association for Computational Linguistics: NAACL 2022*, pages 1535–1546.

OpenAI. 2024. Gpt-4o mini: advancing cost-efficient intelligence.

Fabio Petroni, Aleksandra Piktus, Angela Fan, Patrick Lewis, Majid Yazdani, Nicola De Cao, James Thorne, Yacine Jernite, Vladimir Karpukhin, Jean Maillard, et al. 2021. Kilt: a benchmark for knowledge intensive language tasks. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 2523–2544.

Jingyuan Qi, Zhiyang Xu, Rulin Shao, Yang Chen, Jin Di, Yu Cheng, Qifan Wang, and Lifu Huang. 2024. Rora-vlm: Robust retrieval-augmented vision language models. *Preprint*, arXiv:2410.08876.

Nils Reimers and Iryna Gurevych. 2019. Sentence-bert: Sentence embeddings using siamese bert-networks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics.

Stephen Robertson, Hugo Zaragoza, et al. 2009. The probabilistic relevance framework: Bm25 and beyond. *Foundations and Trends® in Information Retrieval*, 3(4):333–389.

Kihyuk Sohn. 2016. Improved deep metric learning with multi-class n-pair loss objective. *Advances in neural information processing systems*, 29.

Kaitao Song, Xu Tan, Tao Qin, Jianfeng Lu, and Tie-Yan Liu. 2020. Mpnet: Masked and permuted pre-training for language understanding. *Advances in neural information processing systems*, 33:16857–16867.

Michael Steinbach and Pang-Ning Tan. 2009. knn: k-nearest neighbors. In *The top ten algorithms in data mining*, pages 165–176. Chapman and Hall/CRC.

Hongjin Su, Weijia Shi, Jungo Kasai, Yizhong Wang, Yushi Hu, Mari Ostendorf, Wen-tau Yih, Noah A Smith, Luke Zettlemoyer, and Tao Yu. 2023. One embedder, any task: Instruction-finetuned text embeddings. In *Findings of the Association for Computational Linguistics: ACL 2023*, pages 1102–1121.

Yu Sun, Shuohuan Wang, Shikun Feng, Siyu Ding, Chao Pang, Junyuan Shang, Jiaxiang Liu, Xuyi Chen, Yanbin Zhao, Yuxiang Lu, et al. 2021. Ernie 3.0: Large-scale knowledge enhanced pre-training for language understanding and generation. *arXiv preprint arXiv:2107.02137*.

Nandan Thakur, Nils Reimers, Andreas Rücklé, Abhishek Srivastava, and Iryna Gurevych. 2021. BEIR: A heterogeneous benchmark for zero-shot evaluation of information retrieval models. In *Thirty-fifth Conference on Neural Information Processing Systems Datasets and Benchmarks Track (Round 2)*.

Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, et al. 2023. Llama 2: Open foundation and fine-tuned chat models. *arXiv preprint arXiv:2307.09288*.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Ł ukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc.

Ellen Voorhees, Tasmeer Alam, Steven Bedrick, Dina Demner-Fushman, William R Hersh, Kyle Lo, Kirk Roberts, Ian Soboroff, and Lucy Lu Wang. 2021.

Trec-covid: constructing a pandemic information retrieval test collection. In *ACM SIGIR Forum*, volume 54, pages 1–12. ACM New York, NY, USA.

Nikos Voskarides, Dan Li, Pengjie Ren, Evangelos Kanoulas, and Maarten de Rijke. 2020. Query resolution for conversational search with limited supervision. In *Proceedings of the 43rd International ACM SIGIR conference on research and development in Information Retrieval*, pages 921–930.

Denny Vrandečić and Markus Krötzsch. 2014. Wikidata: a free collaborative knowledgebase. *Communications of the ACM*, 57(10):78–85.

Liang Wang, Nan Yang, Xiaolong Huang, Binxing Jiao, Linjun Yang, Daxin Jiang, Rangan Majumder, and Furu Wei. 2023. Simlm: Pre-training with representation bottleneck for dense passage retrieval. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2244–2258.

Liang Wang, Nan Yang, Xiaolong Huang, Linjun Yang, Rangan Majumder, and Furu Wei. 2024. Improving text embeddings with large language models. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 11897–11916, Bangkok, Thailand. Association for Computational Linguistics.

Tongzhou Wang and Phillip Isola. 2020. Understanding contrastive representation learning through alignment and uniformity on the hypersphere. In *International conference on machine learning*, pages 9929–9939. PMLR.

Cong Wei, Yang Chen, Haonan Chen, Hexiang Hu, Ge Zhang, Jie Fu, Alan Ritter, and Wenhu Chen. 2023. Uniir: Training and benchmarking universal multimodal information retrievers. *arXiv preprint arXiv:2311.17136*.

Shitao Xiao, Zheng Liu, Weihao Han, Jianjin Zhang, Yingxia Shao, Defu Lian, Chaozhuo Li, Hao Sun, Denvy Deng, Liangjie Zhang, et al. 2022a. Progressively optimized bi-granular document representation for scalable embedding based retrieval. In *Proceedings of the ACM Web Conference 2022*, pages 286–296.

Shitao Xiao, Zheng Liu, Yingxia Shao, and Zhao Cao. 2022b. Retromae: Pre-training retrieval-oriented language models via masked auto-encoder. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 538–548.

Shitao Xiao, Zheng Liu, Peitian Zhang, Niklas Muennighoff, Defu Lian, and Jian-Yun Nie. 2024. C-pack: Packed resources for general chinese embeddings. In *Proceedings of the 47th International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '24, page 641–649, New York, NY, USA. Association for Computing Machinery.

Zhiyang Xu, Chao Feng, Rulin Shao, Trevor Ashby, Ying Shen, Di Jin, Yu Cheng, Qifan Wang, and Lifu Huang. 2024a. Vision-flan: Scaling human-labeled tasks in visual instruction tuning. In *Findings of the Association for Computational Linguistics: ACL 2024*, pages 15271–15342, Bangkok, Thailand. Association for Computational Linguistics.

Zhiyang Xu, Minqian Liu, Ying Shen, Joy Rimchala, Jiaxin Zhang, Qifan Wang, Yu Cheng, and Lifu Huang. 2024b. Lateralization lora: Interleaved instruction tuning with modality-specialized adaptations. *Preprint*, arXiv:2407.03604.

Zhen Yang, Zhou Shao, Yuxiao Dong, and Jie Tang. 2024. Trisampler: A better negative sampling principle for dense retrieval. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pages 9269–9277.

Zhilin Yang, Peng Qi, Saizheng Zhang, Yoshua Bengio, William Cohen, Ruslan Salakhutdinov, and Christopher D Manning. 2018. Hotpotqa: A dataset for diverse, explainable multi-hop question answering. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 2369–2380.

Chenyu You, Nuo Chen, Fenglin Liu, Shen Ge, Xian Wu, and Yuexian Zou. 2022. End-to-end spoken conversational question answering: Task, dataset and model. *arXiv preprint arXiv:2204.14272*.

Chenyu You, Nuo Chen, Fenglin Liu, Dongchao Yang, and Yuexian Zou. 2020a. Towards data distillation for end-to-end spoken conversational question answering. *arXiv preprint arXiv:2010.08923*.

Chenyu You, Nuo Chen, and Yuexian Zou. 2020b. Contextualized attention-based knowledge transfer for spoken conversational question answering. *arXiv preprint arXiv:2010.11066*.

Chenyu You, Nuo Chen, and Yuexian Zou. 2021a. Knowledge distillation for improved accuracy in spoken question answering. In *IEEE International Conference on Acoustics, Speech and Signal Processing*.

Chenyu You, Nuo Chen, and Yuexian Zou. 2021b. Mrd-net: Multi-modal residual knowledge distillation for spoken question answering. In *IJCAI*, pages 3985–3991.

Chenyu You, Nuo Chen, and Yuexian Zou. 2021c. Self-supervised contrastive cross-modality representation learning for spoken question answering. *arXiv preprint arXiv:2109.03381*.

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

Wanjun Zhong, Junjie Huang, Qian Liu, Ming Zhou, Jiahai Wang, Jian Yin, and Nan Duan. 2022. Reasoning over hybrid chain for table-and-text open domain question answering. In *IJCAI*, pages 4531–4537.

A CompMix-IR Example

A.1 Heterogeneous Evidence Examples

Table 7 provides linearized heterogeneous examples of evidence for the four types of knowledge. Table 8 provides examples of evidence with full annotation information.

Text evidences:
1. Cousteau (band), Reboot In 2016 it was announced that Liam McKahey and Davey Ray Moor were returning as CousteauX and were back in the recording studio preparing new music.
2. Cousteau (band), To honour the new era the band placed an X at the end of their name.
3. Cousteau (band), Cousteaux is another popular French family name.

KG evidences:
1. Maverick, cast member, Robert Colbert
2. Maverick, original language of film or TV show, English
3. Maverick, cast member, Roxane Berard, name of the character role, 'Comtesse de Barot', name of the character role, 'Comtesse Lizette de La Fontaine', name of the character role, 'Felice de Lassignac', name of the character role, 'Danielle de Lisle'

Table evidences:
1. Stefanie Powers, Year is 1975, Title is Gone with the West, Role is Little Moon, Notes is Alternate title: Little Moon and Jud McGraw
2. Stefanie Powers, Year is 1975, Title is It Seemed Like a Good Idea at the Time, Role is Georgia Price, Notes is.
3. Stefanie Powers, Year is 1976, Title is Invisible Strangler, Role is Candy Barrett, Notes is Alternate titles: The Astral Factor , The Astral Fiend

Infobox evidences:
1. When Harry Met Sally..., When Harry Met Sally..., Directed by, Rob Reiner
2. When Harry Met Sally..., When Harry Met Sally..., Written by, Nora Ephron
3. When Harry Met Sally..., When Harry Met Sally..., Produced by, Rob Reiner Andrew Scheinman

Table 7: Evidence examples from the CompMix-IR corpus.

Text evidence:
{ "linearized evidence text": "Museum of Modern Art, Its first successful loan exhibition was in November 1929, displaying paintings by Van Gogh, Gauguin, Cézanne, and Seurat.",
"wikidata entities": [{ "id": "Q34013", "label": "Georges Seurat" }, { "id": "Q17437796", "label": "featured article" }, ...],
"disambiguations": [["1929", "1929-01-01T00:00:00Z"], ["painting", "Q11629"], ["Van Gogh", "Q17437796"], ...],
"retrieved for entity": { "id": "Q188740", "label": "Museum of Modern Art" },
"source": "text" }

KG evidence:
{"linearized evidence text": "Transformers, genre, action film",
"wikidata entities": [{ "id": "Q171453", "label": "Transformers" }, { "id": "Q188473", "label": "action film" }],
"disambiguations": [["Transformers", "Q171453"],...],
"source": "kg"}

Table evidence:
{ "linearized evidence text": "Surrey Scorchers, Season is 2008–09, Division is BBL, Tier is I, Regular Season is 4th, Post-Season is 33, Trophy is 21, Cup is 12, Head Coach is 42",
"wikidata entities": [{ "id": "Q3645013", "label": "2008–09 British Basketball League season" }, { "id": "Q269597", "label": "Surrey Scorchers" }, { "id": "Q23276", "label": "Surrey"}, ...],
"disambiguations": [["2008", "2008-01-01T00:00:00Z"], ["2008–09", "Q3645013"], ...],
"retrieved for entity": { "id": "Q269597" },
"source": "table"}

Infobox evidence:
{ "linearized evidence text": "Older (Royseven song), from the album " The Art of Insincerity, Genre, Rock",
"wikidata entities": [{ "id": "Q7375238", "label": "Royseven" }, { "id": "Q188451", "label": "music genre" }, { "id": "Q7714454", "label": "The Art of Insincerity" }, ...],
"disambiguations": [["The Art of Insincerity", "Q7714454"], ["Royseven", "Q7375238"], ["Genre", "Q188451"], ...],
"retrieved for entity": { "id": "Q7085553" },
"source": "info" }

Table 8: Evidence examples with full annotation information.

A.2 CompMix-IR QA Examples

We present some question-answer examples from the CompMix-IR dataset in Table 9, while Table 10 provides a QA example with full annotation information. Table 11 shows the statistics of the CompMix-IR QA set.

B Prompt Example

Table 12 shows a prompt example we use in constructing Data-text Pairs, with the help of GPT-4o-mini.

Question 1:
Which is the initial book of the book series Divergent?

Answer 1:
Divergent (novel).

Question 2:
Who is the author of the book Divergent (novel)?

Answer 2:
Veronica Roth.

Question 3:
What is the date of birth of the Divergent's author Veronica Roth?

Answer 3:
19 August 1988

Table 9: QA examples from the CompMix-IR dataset.

Question:
Who was the voice actor for Meg Griffin in Family Guy?

Answer:
Mila Kunis

Annotation information: {"question id": "5136",
"question": "Who was the voice actor for Meg Griffin in Family Guy?",
"domain": "tvseries",
"entities": [{"id": "Q908772", "label": "Meg Griffin"}, {"id": "Q5930", "label": "Family Guy"}],
"answers": [{"id": "Q37628", "label": "Mila Kunis"}],
"answer text": "Mila Kunis",
"answer src": "kg"}

Table 10: A QA example with full annotation information.

C Detailed descriptions of Metrics

In our study, we use the following metrics to measure retrieval performance:

- Hit@K, also known as Top-k Accuracy (Karpukhin et al., 2020), measures the proportion of queries for which the top-k retrieved evidence contains the correct answers. This is a key metric for retrievers in the RAG framework.
- Mean Reciprocal Rank (MRR) (Zhao et al., 2024) computes the average of the reciprocal ranks of the first relevant evidence retrieved across a set of queries.

Dataset		Question word (%)	
Train set	4,966	What	39.28
Dev set	1,680	Who	29.69
Test set	2,764	Which	16.90
Total	9,410	How	5.48
Avg. length		When	5.13
Question	9.19	Where	3.32
Answer	2.17	Other	0.20

Table 11: Question answering Statistics of CompMix-IR. 'Avg. length' refers to the average number of words.

Prompt template:
Evidence data is a triple from the wikidata knowledge graph, representing a factual piece of information. The components of the triple are separated by ', ' and represent the head entity, the relation, and the tail entity, respectively. I hope you understand the content of evidence data, and then use grammatically correct natural language sentence to describe the content in evidence data. Here has some demonstrations: **<Demonstrations>**

Table 12: The prompt example used in KG triples.

D Training setup

In this section, we detail the detailed training settings for training UniHGKR-base and UniHGKR-7B. In training phase 3, a larger number of instruction-unfollowing negative samples could potentially harm the performance of the retriever in retrieval scenario 1. Therefore, in our training, we set a probability of 0.005 to add one instruction-unfollowing negative sample in the training samples of retrieval scenario 2.

D.1 UniHGKR-base Training setup

During training phase 1, we initialize model parameters from BERT-base (Devlin et al., 2019) weights. The learning rate is set to 1×10^{-5}. Training is conducted for one epoch with a batch size of 32 per device. In training phase 2, the learning rate increases to 2×10^{-5}. Training also spans one epoch, but the batch size per device increases to 96. In-batch negative samples can be used across devices, increasing the diversity and number of negative samples used during training. In the subsequent training phase 3, the learning rate remains 2×10^{-5}, but the training duration is extended to 5 epochs. The batch size per device is reduced back to 32 to accommodate a larger hard negative sample group, with a size of 15. In training phases 2 and 3, the temperature parameter is set to 0.02, and both phases use FP16 precision mode to enhance

computational efficiency and conserve memory.

D.2 UniHGKR-7B Training setup

In the initial training phase (stages 1 and 2), we initialize model parameters from LLARA-pretrain (Li et al., 2023a) weights. The learning rate is set to 1×10^{-5}, with a batch size of 384 per device, for one epoch. In these stages, we use the full parameter training method. In the third training phase, we increase the learning rate to 2×10^{-4} and reduce the batch size per device to 64 to accommodate a larger negative sample group size of 7. Training is conducted for one epoch. During this phase, we introduce parameter-efficient training method LoRA (Hu et al., 2022) with a rank of 64 and an alpha value of 16. The dropout rate for LoRa is set to 0.1 to prevent overfitting. Similar to UniHGKR-base, we enable in-batch negative sampling across devices to increase the diversity and number of negative samples during training.

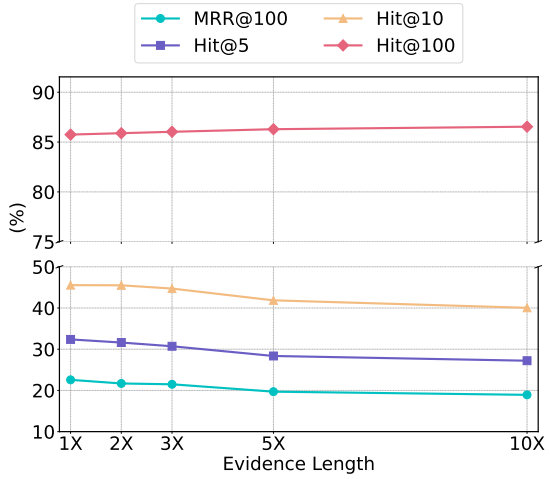


Figure 4: The performance of UniHGKR-base in retrieval Scenario 1 with longer evidences. Here, 10X indicates that the average length of the evidence in the corpus is 10 times the original (1X), and so on.

E Retrieving Robustness of UniHGKR

In this section, we evaluate the performance of the UniHGKR-base model on longer evidence corpora, as well as its zero-shot generalization capabilities.

Robustness for Evidence Length. The robustness of retrievers to varying evidence lengths is crucial, as dense retrievers encounter varying inputs lengths in real-world applications. By increasing the segmentation size of the evidence during the construction of the CompMix-IR corpus, we create several corpus variants,the average length of whose evidence is 2 to 10 times that of the original version. We then evaluate UniHGKR-base, which is trained on the original CompMix-IR corpus, for its retrieval performance on these longer corpus variants, as shown in Figures 4 and 5. From Figure 4, we can see that our UniHGKR-base model shows good robustness with respect to evidence length in retrieval scenario 1. Its performance on metrics like MRR@100 and Hit@5 shows only a slight decline as the evidence length increases, while the Hit@100 metric even shows improvement. This may be because longer evidence can include more information within the fixed number (top-100) evidences, consistent with the findings in (Jiang et al., 2024). On the other hand, Figure 5 shows the retrieval scenario 2 performance of retrieving specified knowledge types on longer evidence. An interesting finding is that the performance of UniHGKR-base in retrieving longer structured data evidence does not decline. Instead, it experiences varying degrees of improvement, most notably on the Table-Hit, where it increases by more than 6 points. This may be because longer evidence can prevent long structured data, such as tables with many rows and columns, from being fragmented into multiple parts, thus avoiding semantic loss.

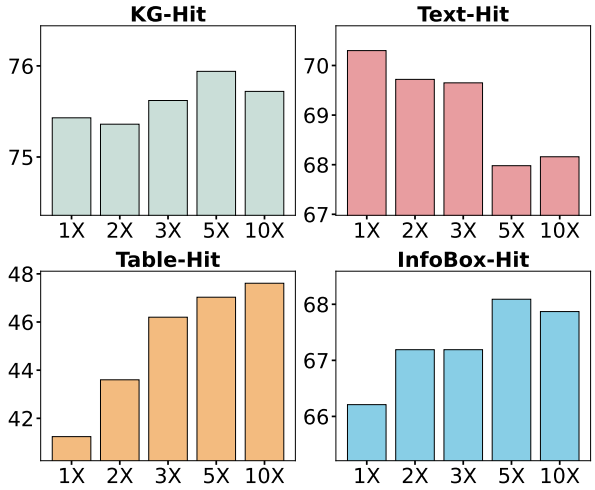


Figure 5: The performance of UniHGKR-base in retrieval Scenario 2 with longer evidences.

Zero-Shot Performance on BEIR. An advantage of instruction-aware universal heterogeneous knowledge retrievers is their enhanced ability to generalize to unseen domains with various types of evidence candidates. To validate this, we evaluate the zero-shot retrieval performance of UniHGKR-base on the popular IR benchmark

BEIR (Thakur et al., 2021). This benchmark includes domains not encountered during UniHGKR's training, such as Bio-Medical and Finance. Following standard setting (Xiao et al., 2022b; Liu et al., 2023), we fine-tune the pretrained model with MS MARCO (Nguyen et al., 2017) and evaluate zero-shot transferability on the other 12 datasets. Following (Thakur et al., 2021), for BEIR, we use NDCG@10 as our primary metric on BEIR. Results for baselines like BERT, SimCSE (Gao et al., 2021), and DiffCS (Chuang et al., 2022) are taken from (Xiao et al., 2022b). As shown in Table 13, our UniHGKR model demonstrates strong zero-shot generalization capabilities. It outperforms baselines on the unseen domain IR datasets, such as the Bio-Medical domain TREC-COVID (Voorhees et al., 2021) and the Finance domain FiQA-2018 (Maia et al., 2018), while maintaining a clear advantage on the familiar task: Wikipedia Entity-Retrieval dataset DBPedia (Hasibi et al., 2017). Additionally, UniHGKR-base also demonstrates clear advantages over the baselines on pure natural language text QA information retrieval datasets, such as NQ (Kwiatkowski et al., 2019) and HotpotQA (Yang et al., 2018). We believe this is because, through our training stages 1 and 2, the model has learned better capabilities to capture the essence of semantic information, which is beneficial for a wide range of retrieval tasks.

Datasets	BERT	SimCSE	DiffCSE	UniHGKR
TREC-COVID	0.615	0.460	0.492	**0.650**
NFCorpus	0.260	0.260	0.259	**0.279**
NQ	0.467	0.435	0.412	**0.490**
HotpotQA	0.488	0.502	0.499	**0.525**
FiQA-2018	0.252	0.250	0.229	**0.261**
ArguAna	0.265	0.413	**0.468**	0.400
Touche-2020	**0.259**	0.159	0.168	0.202
DBPedia	0.314	0.314	0.303	**0.334**
SCIDOCS	0.113	0.124	0.125	**0.133**
FEVER	**0.682**	0.623	0.641	0.670
Climate-FEVER	0.187	**0.211**	0.200	0.205
SciFact	0.533	0.554	0.523	**0.588**
AVERAGE	0.370	0.359	0.360	**0.395**

Table 13: Zero-shot retrieval performances on **BEIR** benchmark (measured by NDCG@10).

F Additional Ablation Studies

F.1 Experiments under the Unsupervised Setting

We conduct experiments under the unsupervised setting (i.e., after training in Stage 1 and Stage 2) in retrieval scenario 1, and the results are shown in Table 14. From these results, we can clearly observe the performance gains brought by each stage to the model's retrieval capabilities. Overall, the alignment training in Stage 2 provides more significant gains compared to the pretraining in Stage 1. After training in Stage 2, the unsupervised model achieves a respectable 73.52 in Hit@100.

Method	Hit@5	Hit@10	Hit@100	MRR@100
Bert-base-uncased	6.55	10.93	37.19	5.04
After Stage 1	9.26	14.47	49.78	6.76
▲Abs. gain	+2.71	+3.54	+12.59	+1.72
After Stage 2	16.03	25.54	73.52	12.10
▲Abs. gain	+6.77	+11.07	+23.74	+5.34
After Stage 3*	**32.38**	**45.55**	**85.75**	**22.57**
▲Abs. gain	+16.35	+20.01	+12.23	+10.47

Table 14: The performance on retrieval scenario 1 after different training stages. Among them, 'After Stage 1' and 'After Stage 2' can be regarded as the performance in the unsupervised setting. 'After Stage 3*' represents our UniHGKR-base model. 'Abs. gain' represents the absolute improvement in performance after each training stage.

F.2 The Impact of Instructions for Retrieving from Specific Sources.

We added experiments on retrieving from specific sources in the I_{All} setting. Based on this, we can compare and observe the improvement in performance when using the instruction I_τ, which specifies the retrieval source, in retrieval scenario 2. In Table 15, we can clearly see that when retrieving specific types of knowledge, our UniHGKR model shows a significant improvement when using the instruction I_τ (where $\tau \in \mathcal{H} =$ Text, Info, Table, KG) compared to using the instruction I_{All}. This is particularly the case for table and infobox-type knowledge. This result indicates that our proposed type-preferred loss ($\mathcal{L}_{\text{preferred}}$) can help the model distinguish data types and capture their differences for flattened inputs with the help of instructions.

F.3 Efficiency of the Proposed Models

For retrieval tasks, efficiency is as important as accuracy. The time cost of retrieval tasks lies in two parts: (1) Embedding, (2) Retrieving. The factor affecting the first part 'Embedding' is the parameter scale of the dense embedder. So, the parameter scales of the baselines and UniHGKR models are shown in Table 3 and Table 5. The efficiency of the second part 'Retrieving' is affected by the dimension of the vector generated by the retriever. We added an experiment to show the time efficiency

Method	Instructions	KG-Hit	Text-Hit	Table-Hit	Info-Hit
UniHGKR-base	I_{All}	68.60	65.70	28.76	57.34
UniHGKR-base	I_{τ}	75.43	70.30	41.24	66.21
Abs. gain		**+6.83%**	**+4.60%**	**+12.48%**	**+8.87%**

Table 15: Performance of retrieving specific knowledge types with different instructions in retrieval scenario 2. 'Abs. gain' refers to the performance improvement brought by using instruction I_{τ} compared to I_{All}.

Model	Size	Vector Dim.	Avg. Embed Time (100 evd)	Avg. Retrieve Time (100 ques)
UniHGKR-Base	109M	768	0.46 s	4.35 s
UniHGKR-7B	7B	4096	1.54 s	53.29 s

Table 16: Time efficiency comparison between UniHGKR-Base and UniHGKR-7B. The experiment was conducted on a single V100-32G GPU on the CompMix-IR. The data are the average values of three runs of the experiment for 100 pieces of evidence or 100 questions.

difference between the UniHGKR-Base model and the UniHGKR-7B model, as shown in Table 16. The embedding and retrieving average time costs for UniHGKR-7B are 3.35 and 12.25 times longer than those for UniHGKR-Base, respectively. Note that during retrieval, we did not use fast vector retrieval libraries such as Faiss (Johnson et al., 2019) but instead performed a naive KNN (Steinbach and Tan, 2009) computation.

G Detailed Description of UniHGKR-7B Adaptation

In our UniHGKR-7B training, we initialize the model weights from the LLaRA-pretrain. LLARA-pretrain model initializes its parameters from LLaMA-2-7B-base (Touvron et al., 2023). The output vector of the last token of the model input sequence S, a special token $\langle\backslash s\rangle$, is used as the embedding representation r of the input sequence:

$$r \leftarrow \text{LLaMA}(S)[\langle\backslash s\rangle].$$

They then apply their proposed Embedding Based AutoEncoder (EBAE) and Embedding Based AutoRegressive (EBAR) techniques for post-training adaptation for dense retrieval. EBAE reconstructs the tokens of the input sentence using r, while EBAR predicts the tokens of the next sentence based on r.

In Stages 1 and 2 of our UniHGKR-7B training, our input sequence S is the linearized structured data d_i. We adapt EBAE to reconstruct d_i and EBAR to predict the corresponding natural language sentence t_i. Here, $\langle d_i, t_i\rangle$ is from the Data-Text Pairs $\mathcal{D}$. This process essentially implements Stages 1 and 2 of our UniHGKR training framework: establishing an effective representation space for heterogeneous knowledge. For task fine-tuning (Stage 3), we use the same training methods as the UniHGKR-base models (BERT-based), including the instruction set and positive/negative sampling strategies (see Section 4.2).