

ZEBRA: Zero-Shot Example-Based Retrieval Augmentation for Commonsense Question Answering

Francesco Maria Molfese*, Simone Conia*, Riccardo Orlando and Roberto Navigli

Sapienza NLP Group, Sapienza University of Rome

{molfese, conia, orlando, navigli}@diag.uniroma1.it

Abstract

Current Large Language Models (LLMs) have shown strong reasoning capabilities in commonsense question answering benchmarks, but the process underlying their success remains largely opaque. As a consequence, recent approaches have equipped LLMs with mechanisms for knowledge retrieval, reasoning and introspection, not only to improve their capabilities but also to enhance the interpretability of their outputs. However, these methods require additional training, hand-crafted templates or human-written explanations. To address these issues, we introduce ZEBRA, a zero-shot question answering framework that combines retrieval, case-based reasoning and introspection and dispenses with the need for additional training of the LLM. Given an input question, ZEBRA retrieves relevant question-knowledge pairs from a knowledge base and generates new knowledge by reasoning over the relationships in these pairs. This generated knowledge is then used to answer the input question, improving the model’s performance and interpretability. We evaluate our approach across 8 well-established commonsense reasoning benchmarks, demonstrating that ZEBRA consistently outperforms strong LLMs and previous knowledge integration approaches, achieving an average accuracy improvement of up to 4.5 points.

1 Introduction

Over recent years, the research community has explored how to improve the reasoning capabilities of language models and the interpretability of their predictions, with many approaches relying on knowledge augmentation (Liu et al., 2022b; Zhang et al., 2022; Yu et al., 2022; Liu et al., 2023). For instance, Das et al. (2021) introduced *case-based reasoning* (Aamodt and Plaza, 1994) into the training process of a knowledge-based question answering system to guide its reasoning. At

* Equal contribution.

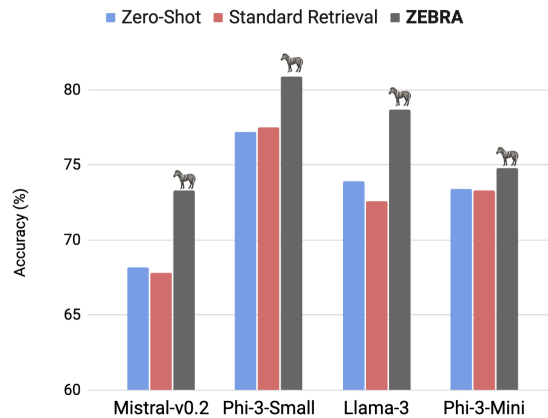


Figure 1: Performance benefits of using ZEBRA against standard retrieval augmentation methods for commonsense reasoning across four Large Language Models.

the same time, Shwartz et al. (2020) and Liu et al. (2022b) investigated how to prompt LLMs to generate useful knowledge via hand-crafted templates or human-written explanations, while Yu et al. (2022) introduced an approach to teach an LLM to reason over a knowledge base of commonsense knowledge, which is accessed via a retriever. Furthermore, Liu et al. (2022a) introduced the concept of *knowledge introspection*, which is the process of generating contextually relevant knowledge in response to given questions.

However, some of these approaches rely on commonsense knowledge bases, which are finite by definition and, therefore, may not include the exact information needed to augment the input for downstream tasks. Moreover, most of the aforementioned approaches require the user to train – or, at least, fine-tune – existing LLMs; not only is this process resource-intensive but it also leads to task-specific models, which may result in sub-optimal performance outside the domain of their training datasets. To address these issues, we introduce ZEBRA, a zero-shot framework for commonsense

reasoning and question answering that aims to combine the benefits of knowledge retrieval, case-based reasoning, and introspection without fine-tuning the underlying LLM. ZEBRA stems from two observations: first, direct retrieval of commonsense facts may provide useful hints, but it is limited by the finite nature of knowledge bases and the noise therein; second, introspection can generate contextually relevant knowledge which is tailored to the input question, but this is limited to what the LLM already “knows”. In contrast, rather than directly retrieving or generating knowledge for the specific input question, ZEBRA: i) retrieves one or more *examples* – also referred to as *cases* in the literature – that elicit a correct reasoning process for the input question based on commonsense, ii) generates commonsense knowledge tailored for the input question by following the relationship in the question-knowledge pairs contained in the retrieved examples, and iii) uses the generated knowledge to answer the input question.

We can summarize the contributions of this work as follows:

- We introduce ZEBRA, a zero-shot example-based retrieval augmentation framework for commonsense reasoning and question answering that combines the benefits of knowledge retrieval and introspection while dropping the need for additional training of the LLM.
- We create ZEBRA-KB, a high-quality silver knowledge base for commonsense question answering, where each entry is composed of a question, a list of choices, and a list of explanations based on commonsense reasoning.
- We evaluate ZEBRA and ZEBRA-KB across 8 commonsense reasoning benchmarks, demonstrating that ZEBRA consistently outperforms the baselines, achieving an average accuracy improvement of up to 4.5 points.

We believe that ZEBRA represents a significant step forward for improving the capabilities of LLMs in commonsense question answering tasks, as outlined in Figure 1. We release our software and data at <https://github.com/sapienzanlp/zebra>.

2 Related Work

Over the years, the research community has proposed several approaches for the integration of commonsense knowledge into language models.

Commonsense generation. There is a large body of work that has investigated how to pre-train or fine-tune language models to generate commonsense knowledge based on the information available in commonsense knowledge bases and commonsense corpora. Notably, [Bosselut et al. \(2019\)](#) introduced COMET, a language model trained using explicit knowledge derived from commonsense knowledge bases, such as ConceptNet ([Speer et al., 2018](#)) and ATOMIC ([Sap et al., 2019](#)). Similarly, [Zhou et al. \(2020\)](#) proposed CALM, a procedure to pre-train language models by integrating generative and contrastive objectives for learning commonsense. In addition, other approaches, exemplified by Unicorn ([Lourie et al., 2021](#)), fine-tune generative language models to solve a wide-array of commonsense-based tasks in a multitask fashion. These methods now represent a milestone in the field of commonsense reasoning, as they have shown themselves to be effective in generating structured commonsense knowledge and reasoning over it. However, these approaches require training and/or fine-tuning on large quantities of data. Furthermore, since they generate the answers directly, they remain hard to interpret for humans.

Knowledge integration. Other efforts shifted their focus onto improving the reasoning abilities of language models by grounding them in external commonsense knowledge graphs. Notable works in this direction include KagNet ([Lin et al., 2019](#)), GRF ([Ji et al., 2020](#)), QA-GNN ([Yasunaga et al., 2021](#)) and GreaseLM ([Zhang et al., 2022](#)), among others, which encode commonsense-based knowledge graphs at the input level to guide the reasoning process, while also increasing its interpretability. However, these approaches not only still require extensive training and fine-tuning, but their performance is also upper-bounded by the quality and completeness of the knowledge graph.

Knowledge retrieval. There have also been several efforts to equip LLMs with mechanisms to retrieve commonsense knowledge from structured and unstructured data ([Lewis et al., 2020](#)). Noteworthy is the work by [Yu et al. \(2022\)](#), who introduced RACo, a retriever-generator approach to retrieve and use commonsense knowledge in knowledge-intensive tasks. More specifically, the knowledge is drawn by the retriever from a large knowledge base comprising twenty million commonsense statements, situations, and events collected from various resources and benchmarks,

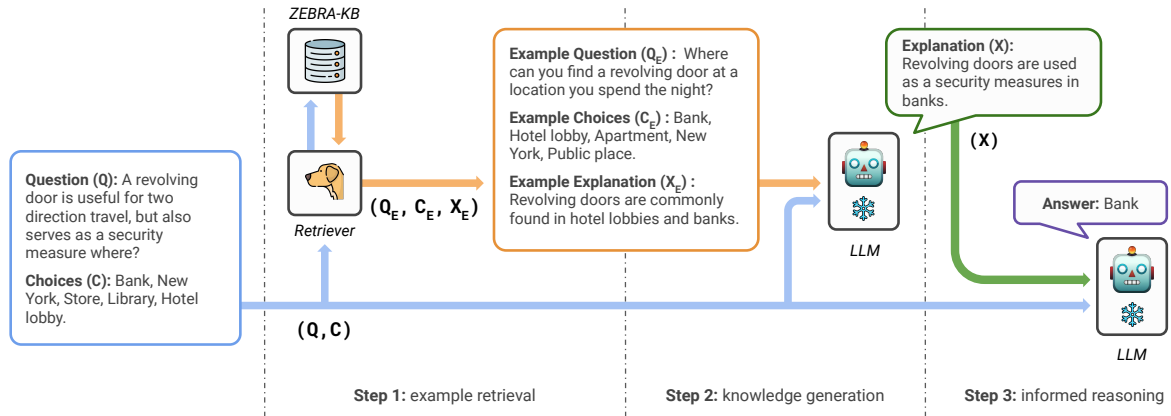


Figure 2: The ZEBRA framework in its entirety. Starting with a question Q and its possible choices C , the first step (*example retrieval*) is to ask the retriever to fetch relevant examples from a collection made of questions along with their choices and associated knowledge explanations (Q_e, C_e, X_e) . Then, the model is asked to generate one or more explanations X for the question Q with choices C emulating the relationship in the elements (Q_e, C_e, X_e) of the examples (*knowledge generation* step). Finally, during the *informed reasoning* step, the same model is asked to perform question answering on the question Q given the choices C and the generated knowledge explanations X .

such as ConceptNet, ATOMIC, and GKB (Bhaktavatsalam et al., 2020), among others. Then, the knowledge is integrated into a reader model via Fusion-in-Decoder (Izacard and Grave, 2020, FiD) that is trained to output the correct answer. Unfortunately, these approaches suffer from the same drawbacks as the previous ones, as they require additional training and fine-tuning, and their performance is limited by the quality and completeness of the knowledge base used.

Knowledge generation. Shwartz et al. (2020, Self-Talk) and Liu et al. (2022b, GKP) were the first to introduce approaches to generate commonsense knowledge about the input question before providing the answer. However, the generalizability and application of such approaches to broader domains is hindered by the need for human intervention: Self-Talk requires hand-crafted templates specific to each dataset, whereas GKP requires human-written explanations designed specifically for each task.

Finally, introspection has been proposed to improve the interpretability of language models by generating contextually relevant knowledge for a given question. Liu et al. (2022a) were the first to introduce the concept of *knowledge introspection*, which paved the way to the development of introspective reasoners, including Rainier (Liu et al., 2022a) and Crystal (Liu et al., 2023). These approaches are based on teaching an LLM to generate knowledge that is specifically helpful in answering a given question and, at the same time,

in optimizing their generated answers conditioned on the knowledge generated during the first phase via reinforcement learning techniques. Although introspective reasoners have achieved remarkable results, they still require additional training and fine-tuning; moreover, the generated knowledge and their reasoning patterns are strongly limited by what they have seen at training time. In conclusion, to the best of our knowledge, no work has yet explored combining retrieval and introspection with a view to enhancing the reasoning capabilities of language models in a zero-shot setting without the need for additional training of the LLM.

3 ZEBRA

In this section, we introduce ZEBRA, our novel zero-shot example-based retrieval augmentation framework for commonsense reasoning that combines the benefits of knowledge retrieval and introspection while dropping the need for additional training of the LLM. ZEBRA’s pipeline is composed of three main steps, as illustrated in Figure 2: i) example retrieval (Section 3.1), ii) example-guided knowledge generation (Section 3.2), and iii) knowledge-informed reasoning (Section 3.3).

3.1 Example Retrieval

The first step of ZEBRA is to retrieve the “examples” from a dedicated knowledge base. The key difference from standard knowledge retrieval is that, instead of retrieving isolated facts or statements, we retrieve complete examples. Each ex-

ample consists of a question, a list of choices, and simple explanations that clarify how commonsense knowledge justifies the correctness or incorrectness of each choice in relation to the input question.¹ Retrieving full examples allows ZEBRA to provide a broader context for the reasoning process that the LLM has to follow to generate the knowledge necessary to answer the question.

Retriever architecture. Our example retriever builds on top of DPR (Karpukhin et al., 2020, Dense Passage Retrieval), which uses an encoder to produce a dense representation of the query and the passages. Given an input query q and a passage $p \in \mathcal{P}$, where \mathcal{P} is a collection of passages, the $\text{Retriever}(\cdot)$ model computes the embeddings of q and p :

$$E_Q(q) = \text{Retriever}(q), E_P(p) = \text{Retriever}(p)$$

Then, we rank the most relevant passages with respect to q using the similarity function $\text{sim}(q, p) = E_Q(q)^\top E_P(p)$, i.e., the dot product between the query and passage embeddings. The retriever returns the top- k passages $\mathcal{P}(q)$ that are most similar to the input query q .

Query and passage representation. For our retriever to work, it is fundamental to encode the queries (input questions and their choices) and the passages (questions of the examples and the corresponding choices) in a way that allows the model to retrieve the most relevant ones, i.e., the positives, and tell them apart from the irrelevant ones, i.e., the negatives. Let the query $q = (Q, C)$ be composed of a question Q and a list of possible choices $C = (c^1, c^2, \dots, c^n)$. Then, we represent the query q as the concatenation of the question Q and each choice c^i separated by a special token:

$$Q \text{ [SEP]} c^1 \text{ [SEP]} c^2 \dots \text{ [SEP]} c^n \quad (1)$$

For each query q , we sample a set of positive passages $\bar{\mathcal{P}}(q)$ from all the queries in the knowledge base of examples. More specifically, a passage p is considered the positive of a query q if the two share the same main topic, i.e., they are about the same concept. In addition, we augment the set of positive passages by permuting, removing, or re-ordering the choices in the queries, as the model should be able to retrieve relevant examples even

¹In the following, we use the terms “explanations” and “knowledge” interchangeably to refer to “explanations that use commonsense knowledge.”

when the choices are presented in a different order or present different distractors.

Training objective. We train the retriever using multi-label noise contrastive estimation (NCE) as the training objective. The $\mathcal{L}_{\text{Retriever}}$ loss for q is defined as:

$$-\log \sum_{p^+ \in \bar{\mathcal{P}}(q)} \frac{e^{\text{sim}(q, p^+)}}{e^{\text{sim}(q, p^+)} + \sum_{p^- \in \hat{\mathcal{P}}(q)} e^{\text{sim}(q, p^-)}} \quad (2)$$

where $\bar{\mathcal{P}}(q)$ are the positives for q , and $\hat{\mathcal{P}}(q)$ is the set of negative samples for q , built using the positives of the other queries in the same batch.

3.2 Example-Guided Knowledge Generation

Given a question Q and a list of choices $C = (c^1, c^2, \dots, c^n)$, the next step in ZEBRA is to generate a list of explanations X that can help in answering the question Q . Unlike introspective reasoners, which generate relevant knowledge directly, we build on top of case-based reasoning. Our approach encourages an LLM to generate knowledge by emulating the relationship in the question-knowledge pairs found in the retrieved examples. More specifically, we retrieve the top k examples E that are conceptually relevant to the given input q :

$$E = \text{top-k}(\text{Retriever}(q)) \quad (3)$$

Each example $e_i \in E \forall i = 1, \dots, k$ is composed of a question Q_{e_i} , choices $C_{e_i} = (c_{e_i}^1, c_{e_i}^2, \dots, c_{e_i}^n)$ and a sequence of gold or silver explanations $X_{e_i} = (x_{e_i}^1, x_{e_i}^2, \dots, x_{e_i}^m)$ that can help answering Q_{e_i} . Subsequently, we construct a prompt containing all of these items of information and ask the LLM to generate a list of explanations X for (Q, C) , following the relationship in the questions $Q_{e_{1:k}}$, choices $C_{e_{1:k}}$, and explanations $X_{e_{1:k}}$ of the top- k retrieved examples:

$$X = \text{Prompt}(Q_{e_{1:k}}, C_{e_{1:k}}, X_{e_{1:k}}, Q, C) \quad (4)$$

We provide more details about the construction of the prompt in Appendix A.1.

3.3 Knowledge-Informed Reasoning

Having generated the list of explanations X for (Q, C) , ZEBRA proceeds to the final step, where the LLM is asked to perform question answering on the input question Q with the list of choices C and the explanations X . Therefore, the model is

asked to predict the correct answer A by conditioning on the input (Q, C, X) :

$$A = \operatorname{argmax}_{c \in C} P(c|Q, C, X) \quad (5)$$

where $P(c|Q, C, X)$ is the probability of the choice c given Q, C , and X . In practice, we compute the probability of the label assigned by the model to each choice c and select the one with the highest probability as the final prediction. We note that this approach can easily be extended to a few-shot setting by providing the model with a few examples before asking the question Q . We provide additional details about the prompt used for this step in Appendix A.2.

3.4 ZEBRA-KB

ZEBRA requires a knowledge base of examples to retrieve and generate commonsense knowledge. Specifically, each example in the knowledge base is composed of a question, a list of choices, and a list of explanations that can help answering the question. These examples can be drawn from the training sets of well-established question answering datasets. A dataset providing such examples is the CommonsenseQA dataset (Talmor et al., 2019), which benefits from a manually-annotated explanations (Rajani et al., 2019; Aggarwal et al., 2021, CoS-E and ECQA). However, not every QA dataset provides such explanations. To obtain an inexpensive but effective solution, we propose ZEBRA-KB, a new knowledge base of examples with gold and silver explanations, the latter being generated using commercially-available LLMs, such as GPT and Gemini. We provide more details about the generation of ZEBRA-KB in Section 4.3 and Appendix A.3.

4 Experimental Setup

In this section, we describe the experimental setup used to train and evaluate ZEBRA. More specifically, we provide an overview on the training process of the retriever (Section 4.1), the datasets used to evaluate our framework (Section 4.2), the process to generate ZEBRA-KB (Section 4.3), as well as the models that we consider for our experiments (Section 4.4).

4.1 Retriever

We build our retriever on top of E5-base-v2 (Wang et al., 2022), a small-sized transformer-based encoder (about 109M parameters) that is pre-trained

HuggingFace model ID	Alias
mistralai/Mistral-7B-Instruct-v0.2	Mistral-v0.2
microsoft/Phi-3-small-8k-instruct	Phi-3-Small
meta-llama/Meta-Llama-3-8B-Instruct	Llama-3
microsoft/Phi-3-mini-128k-instruct	Phi-3-Mini

Table 1: List of LLMs considered for ZEBRA. For better readability, we provide a mapping from the original HuggingFace model IDs to their aliases.

on a large corpus of text. We fine-tune the retriever on the training set of CommonsenseQA (Talmor et al., 2019, CSQA) using the procedure described in Section 3.1. In CSQA, each question is associated with a topic or concept, which we use to construct the set of positive examples for each query. More specifically, we consider two questions q and q' in the dataset as pair-wise positives if they are tagged with the same topic or concept. We select at most 64 positive examples for each query and use up to 200 negatives per batch. We train the encoder for a maximum of 25,000 steps using RAdam (Liu et al., 2020) with a learning rate of $1e-5$ and a linear learning rate decay schedule. At the end of the training, we select the best model based on the loss on the validation set of CSQA. We highlight that the retriever is trained only once and is then used to retrieve examples for all the LLMs and datasets we evaluate. Moreover, the retriever is the only trained component in ZEBRA, as the parameters of the LLMs are kept frozen during the entire process.

4.2 Evaluation Benchmarks

We evaluate our approach against 8 well-established QA datasets: CommonsenseQA (Talmor et al., 2019, CSQA), OpenBookQA (Mihaylov et al., 2018, OBQA), ARC-Easy and ARC-Challenge (Clark et al., 2018), PIQA (Bisk et al., 2019), WinoGrande (Sakaguchi et al., 2019, WG), CommonsenseQA 2.0 (Talmor et al., 2022, CSQA2) and QASC (Khot et al., 2020). To ensure fair and consistent comparisons with recent work (Yu et al., 2022; Liu et al., 2023), we follow standard practice by evaluating on test sets when their labels are publicly available, and otherwise on development sets. Specifically, we use the development sets for CSQA, PIQA, WG, CSQA2, and QASC, and the test sets for ARC and OBQA.

4.3 Creating ZEBRA-KB

CommonsenseQA features a manually annotated set of explanations for each question that was origi-

Size	Model	ZEBRA					Oracle	
		k = 1	k = 3	k = 5	k = 10	k = 20		
7 ~ 8B	Mistral-v0.2	68.2	68.6	<u>72.5</u>	73.3	72.1	71.8	90.3
	Phi-3-Small	77.2	<u>80.8</u>	80.7	80.9	79.5	79.6	95.2
	Llama-3	73.9	77.4	78.7	78.7	<u>78.0</u>	76.6	95.5
~ 4B	Phi-3-Mini	73.4	75.3	<u>74.9</u>	74.8	73.9	72.8	94.9
	Average	73.2	75.5	<u>76.7</u>	76.9	75.9	75.2	94.0
	Δ Improvement	–	+2.3	<u>+3.5</u>	+3.7	+2.7	+2.0	+20.8

Table 2: Results in terms of accuracy on the CSQA development set. Here, k is the number of examples used during the knowledge generation step. “Oracle” indicates the results of the models when they have access to the manually-created explanations of ECQA. Best results are in **bold**, while second-best results are underlined.

nally introduced in prior work (Aggarwal et al., 2021, ECQA). However, this is not a realistic scenario for QA datasets in general and for real-world applications. As a matter of fact, CSQA and OBQA are the only datasets in our evaluation that provide such explanations. However, this limitation does not prevent us from evaluating our framework on the other datasets, as we can generate the required explanations using a silver annotation procedure that is effective and inexpensive.

To create our silver explanations we start from the training set of each dataset and, for each sample, we consider its question Q , choices C , and – most importantly – the correct answer A . Then, we use Google’s GenerativeAPI to prompt Gemini-1.5-Flash² to generate a list of explanations given Q , C , and A , setting the temperature to 0.0 (for reproducibility) and the maximum number of new tokens to 256. We select the top-10 explanations returned by Gemini. The result is a knowledge base of exemplar commonsense knowledge, which we refer to as ZEBRA-KB. We note that relying on commercially-available LLMs limits their output, i.e., the generated explanations, when the questions and/or the possible choices concern sensitive and possibly unsafe topics, e.g., drugs, sex, violence, and race, among others (Tedeschi et al., 2024). Future work may address this limitation in order to generate more comprehensive and diverse explanations. We provide the details about the prompt used for the generation of ZEBRA-KB in Appendix A.3.

4.4 Models

We evaluate ZEBRA using four instruction-tuned LLMs. Table 1 presents the models selected for

evaluation and provides a mapping from their original HuggingFace model IDs to the aliases used in this paper for better readability. Three of these models – Mistral-v0.2, Phi-3-Small, and Llama-3 – feature a similar number of parameters. We also include Phi-3-Mini to evaluate the effectiveness of ZEBRA on a LLM with a significantly lower number of parameters (3.8 billion compared to 7–8 billion of the others). This set of LLMs allows us to evaluate our approach on top of the strongest LLMs available at the time of writing, as well as on smaller models that are computationally less expensive.

5 Results

In this section, we present and discuss the results of ZEBRA on the 8 commonsense question-answering benchmarks introduced in Section 4.2 using the models listed in Section 4.4.

5.1 Results on CSQA

Table 2 provides an overview of the results obtained by different LLMs on CSQA (Talmor et al., 2019). For each model, we report the score obtained in the zero-shot setting, as well as the performance when using the ZEBRA framework with different numbers of retrieved examples k . Note that, here, k is the number of examples used during the knowledge generation step, not the number of in-context examples used for question answering. We also report the “oracle” accuracy of each model when using the manually-created explanations provided in ECQA (right-most column in Table 2), which should represent the upper bound of the performance for an LLM when a human provides one or more explanations to a question having the correct answer available. It is important to underline

²All our explanations are generated using the latest version of Gemini-1.5-Flash available in May 2024.

Model	ARC-C	ARC-E	OBQA	PIQA	WG	CSQA2	QASC	Avg.	Δ
Mistral-v0.2	72.4 / 75.2	85.8 / 87.4	68.8 / 75.8	76.1 / 80.2	55.8 / 60.7	58.5 / 67.5	66.1 / 68.3	69.1 / 73.6	+4.5
Phi-3-Small	90.4 / 91.6	96.9 / 97.7	90.4 / 91.2	86.6 / 88.1	79.1 / 81.0	68.0 / 74.6	83.5 / 81.0	85.0 / 86.4	+1.4
Llama-3	79.4 / 83.5	91.7 / 92.9	73.4 / 79.6	78.3 / 84.0	56.2 / 63.2	64.3 / 69.4	78.2 / 79.1	74.5 / 78.8	+4.3
Phi-3-Mini	85.7 / 88.0	95.4 / 96.0	82.8 / 87.8	80.4 / 84.2	67.3 / 72.9	59.3 / 64.6	74.7 / 73.9	77.9 / 81.0	+3.1

Table 3: Results in terms of accuracy on 7 commonsense benchmarks: ARC-Challenge (ARC-C), ARC-Easy (ARC-E), OpenBookQA (OBQA), PIQA, WinoGrande (WG), CommonsenseQA 2.0 (CSQA2), and QASC. The results are reported in the format *zero-shot* / ZEBRA with $k = 5$ retrieved examples. Best results are in **bold**.

that, because our retriever model is trained specifically on the CSQA training set, these results can also be viewed as an extrinsic evaluation of the in-domain performance of the retriever.

We can immediately see that ZEBRA consistently improves the performance of all the LLMs, with an average increase of 3.7 points of accuracy when $k = 5$. Moreover, we can observe that the performance gain is approximately the same across different LLMs, independently of the architecture and the number of parameters, highlighting the reliability of our approach. Although the results obtained with the oracle knowledge seem to suggest an even higher increase in performance, we stress the fact that those explanations often contain an explicit link between the question and the correct answer, making the models more likely to take a “shortcut” to the correct answer. Indeed, as mentioned above, the explanations are hand-crafted by a human who has access not only to the question and the choices but also to the correct answer, which is not the case in a real-world scenario. We provide examples of this pattern in Appendix A.4.

5.2 Results on Other Benchmarks

The results of ZEBRA on CSQA in Table 2 help us determine the best number of examples to use during the knowledge generation step, which we find to be $k = 5$. Having established the best value for k , we shift our focus to the evaluation of the generalizability of ZEBRA on another 7 benchmarks, which assess different aspects of commonsense reasoning, e.g., physical interactions in PIQA, coreference resolution in WG, science knowledge in ARC and OBQA, etc. Importantly, for each of these benchmarks, we retrieve the k examples for each question from the corresponding training set contained in ZEBRA-KB.

Table 3 summarizes the results, where we can see that ZEBRA consistently outperforms the baselines across all the 7 benchmarks and all the 4 LLMs that we consider, with the exception of

QASC for Phi-3-Small and Phi-3-Mini. The consistent improvement in performance across different datasets and LLMs highlights the reliability of ZEBRA even when the knowledge base is generated through a silver annotation procedure and even when the retriever is trained on a dataset (CSQA) which is potentially very different from the ones used in this evaluation. In Appendix A.5, we present additional experiments demonstrating that ZEBRA remains effective even when the retrieved examples do not come from the same distribution of the evaluation dataset. This highlights ZEBRA’s robustness when both the retriever and the knowledge base are affected by domain shift.

5.3 Comparison with Knowledge Retrieval

To further show the effectiveness of ZEBRA, we carry out a 1-to-1 comparison against an approach that retrieves commonsense knowledge statements directly. Specifically, we train a retriever to fetch commonsense knowledge statements (rather than full examples) that are relevant to the input question using the same retrieval strategy as that introduced in RACo (Yu et al., 2022). Moreover, to ensure a fair comparison, we replace the specialized reader in RACo – a T5 model trained using FiD (Izacard and Grave, 2020) – with the same LLMs we use in our experimental setup. We refer to this approach as RACo-based Retrieval (RBR).

Figure 3 compares the results of the four LLMs that we consider in this work when using ZEBRA and RBR on the CSQA development set. Here, we can see that ZEBRA consistently outperforms RBR across all the LLMs and all the values of k with the exception of Phi-3-Mini when $k = 20$. Interestingly, standard knowledge retrieval often leads to a negative impact on the performance of the LLMs, as we can see in the case of Mistral-v0.2, Llama-3 and Phi-3-Mini, where we observe a decrease in performance when using RBR compared to the vanilla LLMs. Our analysis highlights the limitations of current commonsense knowledge bases

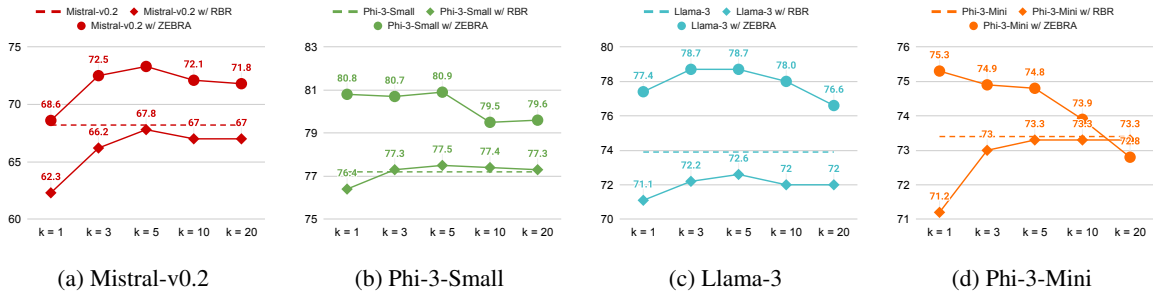


Figure 3: Comparison of the LLMs performance on the CSQA development set using ZEBRA and direct knowledge retrieval (RACo-based Retrieval) as the number of retrieved examples/knowledge statements k increases.

	System	Generator	Reasoner	CSQA	ARC-C	ARC-E	OBQA	QASC	PIQA	CSQA2	WG	AVG
	Llama-3	-	Llama-3	73.9	<u>79.4</u>	91.7	<u>73.4</u>	78.2	<u>78.3</u>	<u>64.3</u>	56.2	<u>74.4</u>
Superv.	Rainier-large	T5 0.77B	Llama-3	72.9	76.0	88.6	71.4	74.5	76.6	57.1	59.3	72.0
	Crystal-3B	T5 3B	Llama-3	72.6	75.5	89.5	72.6	75.9	77.7	58.6	60.1	72.8
	Crystal-11B	T5 11B	Llama-3	<u>75.1</u>	77.3	91.2	72.6	<u>78.4</u>	78.2	60.0	<u>60.5</u>	74.1
Unsuperv.	Self-Talk	Llama-3	Llama-3	70.6	78.7*	91.4*	72.2*	78.3*	77.2	63.4*	58.3	73.8
	GKP	Llama-3	Llama-3	74.0	78.5*	91.5*	70.0*	76.9	76.5*	65.9	60.4*	74.2
	ZEBRA (Ours)	Llama-3	Llama-3	78.7	84.3	90.9	80.0	79.1	84.0	63.2	69.4	78.7

Table 4: Accuracy scores on the CSQA, ARC-Challenge, ARC-Easy, OBQA, QASC, PIQA, CSQA2 and WG benchmarks when using ZEBRA compared to the baselines. Best results are in **bold**, while second-best results are underlined. A green cell indicates an improvement in performance compared to Llama-3 without generated knowledge, while a red cell indicates a decrease. *: results computed using the original methodology on datasets not evaluated by the baseline authors.

and underscores the need for explanations derived from a reasoning process specific to the input question, rather than relying on general commonsense facts, which may not be sufficient to solve the task effectively.

5.4 Comparison with Knowledge Generation

One important aspect of our approach is the manner in which the knowledge is generated. Therefore, here we compare the quality of the knowledge generated by ZEBRA against previous methods: two unsupervised – namely, Self-Talk (Shwartz et al., 2020) and GKP (Liu et al., 2022b) – and two supervised (introspective reasoners) – namely, Rainier (Liu et al., 2022a) and Crystal (Liu et al., 2023).

To test the quality of the generated knowledge of each system in a fair setting, we evaluate the different knowledge *generators* (i.e., the module of the system that generates the knowledge) using the same LLM as a *reasoner* (i.e., the module of the system that answers the question) for all the systems. This allows us to evaluate whether ZEBRA’s example-based retrieval augmentation framework produces knowledge that is of higher quality compared to the hand-crafted templates in Self-Talk and the manually-curated explanations in

GKP. Moreover, this setting enables a direct comparison between ZEBRA, which does not require fine-tuning of the underlying LLM, and models that are specifically trained to generate relevant knowledge, such as Rainier and Crystal.

Table 4 shows the accuracy scores of ZEBRA and all the baselines over the 8 benchmarks introduced in Section 4.2. Interestingly, we observe that most systems – except for ZEBRA – encounter difficulties in scoring higher than a vanilla Llama-3 model in a consistent way across the datasets without any input knowledge (first row in Table 4). These results suggest that current approaches are not suitable for improving the reasoning capabilities of current LLMs. Instead, ZEBRA surpasses the best performing baseline by an average of 4.3 points of accuracy, showcasing the effectiveness of retrieving relevant examples for the input question and allowing the model to generate knowledge by mimicking the relationship in the retrieved question-knowledge pairs. Notably, ZEBRA is also able to outperform supervised techniques that have been explicitly trained to generate and leverage knowledge when answering an input question.³

³For example, the original reasoner (Khashabi et al., 2020, UnifiedQA) in Rainier is trained for question answering but

These results highlight the effectiveness of ZEBRA in providing a simple and effective framework to improve the performance of current LLMs on commonsense reasoning tasks.

5.5 Human Evaluation

Besides better results on standard benchmarks, one of the most important strengths of ZEBRA is the interpretability of the answers provided by the LLMs. Indeed, the knowledge generated by the LLMs can be used by humans to understand the reasoning process that led to the final answer. To evaluate this aspect, we conduct a small-scale manual analysis on the quality of the knowledge generated by the LLMs. We randomly sample 100 instances from the CSQA development set and ask three annotators to validate the quality of the knowledge generated by ZEBRA when using Llama-3. The annotation process is conducted in a blind fashion, i.e., each annotator is not aware of the labels assigned by the other annotators.

Following standard practice in the field (Liu et al., 2022b), each human annotator is assigned the task of evaluating the quality of the generated knowledge in relation to the input question, its answer choices, and the correct answer. The assessment is carried out using three metrics:

- **Relevance:** whether the generated knowledge is relevant to the topic or concepts mentioned in the question; relevant knowledge is not necessarily factual or helpful.
- **Factuality:** whether the generated knowledge is factually correct; factual knowledge is not necessarily relevant or helpful.
- **Helpfulness:** whether the generated knowledge helps in answering the question in a direct or indirect way. We distinguish between three categories: helpful (i.e., supports the correct answer), harmful (i.e., negates the correct answer or supports an incorrect answer), or neutral (neither helpful nor harmful).

Note that an item of helpful knowledge may be factually incorrect, hence all the three dimensions are important for a comprehensive evaluation.

Our human evaluation shows a strong consensus among annotators regarding the quality of the

kept frozen during the training of the knowledge generator, whereas Crystal is jointly trained on question answering and knowledge generation using reinforcement learning.

knowledge generated by ZEBRA using Llama-3. Specifically, all three annotators labeled 96 out of 100 instances as “relevant”, 88 out of 100 instances as “factual”, and 74 out of 100 instances as “helpful”. Moreover, in only 13 out of 100 cases was the knowledge deemed “harmful” by at least one annotator, while only 6 instances were tagged as “harmful” by all three annotators. Finally, there was no instance which was tagged as “not relevant, not factual and harmful” by an annotator, showing the reliability of the explanations generated with ZEBRA. For reference, in Appendix A.6 we provide a number of qualitative examples of the knowledge generated by ZEBRA when using Llama-3 over the CSQA development set.

6 Conclusions

In this paper we presented ZEBRA, a novel zero-shot example-based retrieval augmentation framework for commonsense reasoning. ZEBRA is composed of two main components: an example retriever that fetches examples that are relevant to the input question from a dedicated knowledge base, and an LLM that generates relevant knowledge for the input question by following the relationship in the questions and knowledge provided in the examples. This allows ZEBRA to tackle the limitations of previous methods such as commonsense knowledge retrievers, unsupervised methods relying on human intervention and introspective reasoners, providing a simple and effective way to improve the performance of LLMs on commonsense reasoning and question answering tasks. Since a knowledge base of curated examples may not always be available, we also introduced ZEBRA-KB, a novel knowledge base of examples with silver knowledge explanations for each question, showing how ZEBRA can be used in conjunction with ZEBRA-KB to achieve strong zero-shot results on 8 well-established QA datasets for commonsense reasoning. Our experiments support the effectiveness of ZEBRA over other state-of-the-art methods for knowledge augmented reasoning, such as specialized commonsense retrievers and introspective reasoners. Finally, we investigated the interpretability of the answers provided by the LLMs through a human evaluation of the knowledge generated by ZEBRA. The results demonstrate that the knowledge is often perceived as reasonable by humans, aiding their understanding of the model’s reasoning process leading to the final answer.

Limitations



ZEBRA is not without its limitations. There are several aspects that could be improved in future work. Here, we list and briefly discuss some of the main limitations of our current approach, which may be addressed in future work.

Retriever Performance: The performance of the retriever is crucial for the overall performance of ZEBRA. In this work, our training strategy for the retriever is based on a contrastive learning approach that mainly identifies the positive for a question based on whether the two questions q and q' share or “talk” about the same concept. Despite this approach showing itself to be effective in our experiments, it may not be the most effective strategy for all types of questions. Our positive identification and negative mining strategies are similar to what is commonly used in the literature for dense retrieval, nevertheless, future work may focus on developing more effective and efficient retrieval methods for commonsense reasoning.

Knowledge Base: The quality of the knowledge base is crucial for the overall performance of ZEBRA. In this work, we overcome the lack of manually-crafted explanations for the training sets of many of the evaluation benchmarks we use in our evaluation by using ZEBRA-KB, a knowledge base of examples with silver knowledge explanations created through Google’s Gemini-1.5-Flash, which is relatively fast and inexpensive to interrogate. Future work may focus on the generation of better explanations by using more advanced cleaning and filtering techniques.

Multilinguality and cross-linguality: Our approach is currently limited to English language benchmarks. Future work may focus on extending the framework to multilingual and cross-lingual settings, where the knowledge base is composed of examples in multiple languages, and the LLMs are able to generate knowledge in a language different from the input question, as well as being able to handle input questions written in other languages. This would be interesting to investigate, as the language diversity of the examples in the knowledge base may help or penalize the LLMs on different languages and improve the overall performance of the framework.

Acknowledgements

Simone Conia gratefully acknowledges the support of the PNRR  MUR project PE0000013-FAIR,  which fully funds his fellowship.

Roberto Navigli also gratefully acknowledges the support of the CREATIVE project (CRoss-modal understanding and gENERATIOn of Visual and tEXtual content), which is funded by the MUR Progetti di Rilevante Interesse Nazionale programme (PRIN 2020).

References

- Agnar Aamodt and Enric Plaza. 1994. [Case-based reasoning: Foundational issues, methodological variations, and system approaches](#). *AI communications*, 7(1):39–59.
- Shourya Aggarwal, Divyanshu Mandowara, Vishwa-jeet Agrawal, Dinesh Khandelwal, Parag Singla, and Dinesh Garg. 2021. [Explanations for CommonsenseQA: New Dataset and Models](#). 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 3050–3065, Online. Association for Computational Linguistics.
- Sumithra Bhakthavatsalam, Chloe Anastasiades, and Peter Clark. 2020. [Genericskb: A knowledge base of generic statements](#). *Preprint*, arXiv:2005.00660.
- Yonatan Bisk, Rowan Zellers, Ronan Le Bras, Jianfeng Gao, and Yejin Choi. 2019. [Piqa: Reasoning about physical commonsense in natural language](#). *Preprint*, arXiv:1911.11641.
- Antoine Bosselut, Hannah Rashkin, Maarten Sap, Chaitanya Malaviya, Asli Celikyilmaz, and Yejin Choi. 2019. [COMET: Commonsense transformers for automatic knowledge graph construction](#). In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 4762–4779, Florence, Italy. Association for Computational Linguistics.
- Peter Clark, Isaac Cowhey, Oren Etzioni, Tushar Khot, Ashish Sabharwal, Carissa Schoenick, and Oyvind Tafjord. 2018. [Think you have solved question answering? try arc, the ai2 reasoning challenge](#). *Preprint*, arXiv:1803.05457.
- Rajarshi Das, Manzil Zaheer, Dung Thai, Ameya Godbole, Ethan Perez, Jay Yoon Lee, Lizhen Tan, Lazaros Polymenakos, and Andrew McCallum. 2021. [Case-based reasoning for natural language queries over knowledge bases](#). In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 9594–9611, Online and

- Punta Cana, Dominican Republic. Association for Computational Linguistics.
- Gautier Izacard and Edouard Grave. 2020. [Leveraging passage retrieval with generative models for open domain question answering](#). *arXiv preprint*.
- Haozhe Ji, Pei Ke, Shaohan Huang, Furu Wei, Xiaoyan Zhu, and Minlie Huang. 2020. [Language generation with multi-hop reasoning on commonsense knowledge graph](#). In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 725–736, Online. Association for Computational Linguistics.
- Vladimir Karpukhin, Barlas Oguz, Sewon Min, Patrick Lewis, Ledell Wu, Sergey Edunov, Danqi Chen, and Wen-tau Yih. 2020. [Dense passage retrieval for open-domain question answering](#). In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 6769–6781, Online. Association for Computational Linguistics.
- Daniel Khashabi, Sewon Min, Tushar Khot, Ashish Sabharwal, Oyvind Tafjord, Peter Clark, and Hannaneh Hajishirzi. 2020. [UNIFIEDQA: Crossing format boundaries with a single QA system](#). In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 1896–1907, Online. Association for Computational Linguistics.
- Tushar Khot, Peter Clark, Michal Guerquin, Peter Jansen, and Ashish Sabharwal. 2020. [Qasc: A dataset for question answering via sentence composition](#). *Preprint*, arXiv:1910.11473.
- Patrick 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](#). In *Proceedings of the 34th International Conference on Neural Information Processing Systems, NIPS '20*, Red Hook, NY, USA. Curran Associates Inc.
- Bill Yuchen Lin, Xinyue Chen, Jamin Chen, and Xiang Ren. 2019. [KagNet: Knowledge-aware graph networks for commonsense reasoning](#). In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 2829–2839, Hong Kong, China. Association for Computational Linguistics.
- Jiacheng Liu, Skyler Hallinan, Ximing Lu, Pengfei He, Sean Welleck, Hannaneh Hajishirzi, and Yejin Choi. 2022a. [Rainier: Reinforced knowledge introspector for commonsense question answering](#). In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 8938–8958, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.
- Jiacheng Liu, Alisa Liu, Ximing Lu, Sean Welleck, Peter West, Ronan Le Bras, Yejin Choi, and Hannaneh Hajishirzi. 2022b. [Generated knowledge prompting for commonsense reasoning](#). In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 3154–3169, Dublin, Ireland. Association for Computational Linguistics.
- Jiacheng Liu, Ramakanth Pasunuru, Hannaneh Hajishirzi, Yejin Choi, and Asli Celikyilmaz. 2023. [Crystal: Introspective reasoners reinforced with self-feedback](#). In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 11557–11572, Singapore. Association for Computational Linguistics.
- Liyuan Liu, Haoming Jiang, Pengcheng He, Weizhu Chen, Xiaodong Liu, Jianfeng Gao, and Jiawei Han. 2020. [On the variance of the adaptive learning rate and beyond](#). In *Proceedings of the Eighth International Conference on Learning Representations (ICLR 2020)*.
- Nicholas Lourie, Ronan Le Bras, Chandra Bhagavatula, and Yejin Choi. 2021. [Unicorn on rainbow: A universal commonsense reasoning model on a new multitask benchmark](#). *Preprint*, arXiv:2103.13009.
- Todor Mihaylov, Peter Clark, Tushar Khot, and Ashish Sabharwal. 2018. [Can a suit of armor conduct electricity? a new dataset for open book question answering](#). In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 2381–2391, Brussels, Belgium. Association for Computational Linguistics.
- Nazneen Fatema Rajani, Bryan McCann, Caiming Xiong, and Richard Socher. 2019. [Explain yourself! leveraging language models for commonsense reasoning](#). In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 4932–4942, Florence, Italy. Association for Computational Linguistics.
- Keisuke Sakaguchi, Ronan Le Bras, Chandra Bhagavatula, and Yejin Choi. 2019. [Winogrande: An adversarial winograd schema challenge at scale](#). *Preprint*, arXiv:1907.10641.
- Maarten Sap, Ronan LeBras, Emily Allaway, Chandra Bhagavatula, Nicholas Lourie, Hannah Rashkin, Brendan Roof, Noah A. Smith, and Yejin Choi. 2019. [Atomic: An atlas of machine commonsense for if-then reasoning](#). *Preprint*, arXiv:1811.00146.
- Vered Shwartz, Peter West, Ronan Le Bras, Chandra Bhagavatula, and Yejin Choi. 2020. [Unsupervised commonsense question answering with self-talk](#). In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 4615–4629, Online. Association for Computational Linguistics.
- Robyn Speer, Joshua Chin, and Catherine Havasi. 2018. [Conceptnet 5.5: An open multilingual graph of general knowledge](#). *Preprint*, arXiv:1612.03975.

Alon Talmor, Jonathan Herzig, Nicholas Lourie, and Jonathan Berant. 2019. [CommonsenseQA: A question answering challenge targeting commonsense knowledge](#). 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 4149–4158, Minneapolis, Minnesota. Association for Computational Linguistics.

Alon Talmor, Ori Yoran, Ronan Le Bras, Chandra Bhagavatula, Yoav Goldberg, Yejin Choi, and Jonathan Berant. 2022. [Commonsenseqa 2.0: Exposing the limits of ai through gamification](#). *Preprint*, arXiv:2201.05320.

Simone Tedeschi, Felix Friedrich, Patrick Schramowski, Kristian Kersting, Roberto Navigli, Huu Nguyen, and Bo Li. 2024. [Alert: A comprehensive benchmark for assessing large language models’ safety through red teaming](#). *Preprint*, arXiv:2404.08676.

Liang Wang, Nan Yang, Xiaolong Huang, Binxing Jiao, Linjun Yang, Daxin Jiang, Rangan Majumder, and Furu Wei. 2022. [Text embeddings by weakly-supervised contrastive pre-training](#). *arXiv preprint arXiv:2212.03533*.

Michihiro Yasunaga, Hongyu Ren, Antoine Bosselut, Percy Liang, and Jure Leskovec. 2021. [QA-GNN: Reasoning with language models and knowledge graphs for question answering](#). In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 535–546, Online. Association for Computational Linguistics.

Wenhao Yu, Chenguang Zhu, Zhihan Zhang, Shuohang Wang, Zhuosheng Zhang, Yuwei Fang, and Meng Jiang. 2022. [Retrieval augmentation for commonsense reasoning: A unified approach](#). In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 4364–4377, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Xikun Zhang, Antoine Bosselut, Michihiro Yasunaga, Hongyu Ren, Percy Liang, Christopher D. Manning, and Jure Leskovec. 2022. [Greaselm: Graph reasoning enhanced language models for question answering](#). *Preprint*, arXiv:2201.08860.

Wangchunshu Zhou, Dong-Ho Lee, Ravi Kiran Selvam, Seyeon Lee, Bill Yuchen Lin, and Xiang Ren. 2020. [Pre-training text-to-text transformers for concept-centric common sense](#). *Preprint*, arXiv:2011.07956.

A Appendix

A.1 Knowledge Generation Prompts

In Table 5 we report the prompt used for the knowledge generation step. The number of choices specified under the *System* tag is dynamically adapted according to the number of choices available in the QA dataset.

<p>System: <i>You are given a question and 5 choices. Your task is to write one or more explanations that support the most likely option.</i> <i>Note that:</i> <i>* there is always one option that is correct and more likely than the others.</i> <i>* the explanations must support only the most likely option and refute all the others.</i> <i>* the explanations must be simple and concise (max 15 words).</i> <i>Do you understand the task?</i></p>
<p>Assistant: <i>Yes, I understand. Please provide the question and the possible choices.</i></p>
<p>User: <i>Question:</i> {question} <i>Choices:</i> {choices}</p>
<p>Assistant: <i>List of knowledge:</i></p>

Table 5: Prompt for the knowledge generation step.

A.2 Question Answering Prompts

In Table 6 we report the prompt used for the question answering task without additional input knowledge, which is the one used for the evaluation of the vanilla LLMs. The number and the list of labels appearing under the *System* tag are adjusted according to the number of choices available in each QA dataset. Moreover, in Table 7 we also report the prompt for the informed reasoning step, which is the exact same prompt as the one used for question answering, but with additional knowledge included. This latter can come from the knowledge generation step, from retrieval methods such as RACo, or from manual annotations such as ECQA.

A.3 ZEBRA-KB Generation

In Table 8 we show the prompt used to generate the silver knowledge using the Google GenerativeAPI and the Gemini-1.5-Flash large language model. The number and the list of labels appearing under the *System* tag are adjusted according to the number of choices available in each QA dataset.

A.4 ECQA Explanations

In Table 9 we report five examples of instances coming from the development set of the Com-

<p>System: <i>You are a helpful assistant for question answering.</i> <i>You are given a question and 5 choices (labeled A, B, C, D and E).</i> <i>Your task is to choose the label corresponding to the best answer for the question.</i> <i>Do you understand the task?</i></p>
<p>Assistant: <i>Yes, I understand. Please provide the question and the possible choices.</i></p>
<p>User: <i>Question:</i> {question}</p> <p><i>Choices:</i> {choices}</p>
<p>Assistant: <i>Answer:</i></p>

Table 6: Prompt for the question answering task.

<p>System: <i>You are a helpful assistant for question answering.</i> <i>You are given a question, 5 choices (labeled A, B, C, D and E) and a list of explanations.</i> <i>Your task is to choose the label corresponding to the best answer for the question based on the given explanations.</i> <i>Do you understand the task?</i></p>
<p>Assistant: <i>Yes, I understand. Please provide the question and the possible choices.</i></p>
<p>User: <i>Question:</i> {question}</p> <p><i>Choices:</i> {choices}</p> <p><i>Explanations</i> {knowledge}</p>
<p>Assistant: <i>Answer:</i></p>

Table 7: Prompt for the informed reasoning step.

monsenseQA dataset (Talmor et al., 2019, CSQA), which were manually annotated with a list of explanations by Aggarwal et al. (2021) (ECQA). From the table we can see how the explanations contain an explicit link between the question and the correct answer, like *What would go on top of wood?*

<p>System: <i>You are a helpful assistant for question answering.</i> <i>You are given a question requiring commonsense knowledge to be solved, together with three possible choices (labeled A, B and C) and the label corresponding to the correct answer.</i> <i>For each choice, generate a sentence with explicit commonsense knowledge that supports or refutes the choice.</i> <i>The format of the generated knowledge should be in the following form:</i> A. ... B. ... C. ...</p>
<p>User: <i>Question:</i> {question}</p> <p><i>Choices:</i> {choices}</p>

Table 8: Prompt for the generation of the silver knowledge given a question and its choices.

where the correct answer is *carpet* and the explanation is *It is the carpet that could go on top of wood*. If these explanations were to be used as oracle knowledge during the informed reasoning step, the model would likely exploit the information to select the correct answer, resulting in a high probability of success. Consequently, although the results in Table 2 under the oracle column exhibit a significant performance increase compared to both the baselines and ZEBRA, we contend that this improvement is predominantly attributable to the models leveraging this shortcut.

A.5 Out-of-domain Results

In Table 10 we report the score of ZEBRA in an out-of-domain scenario. Specifically, for each dataset, the examples that we retrieve for the knowledge generation step do not come from the related training set contained in ZEBRA-KB. Instead, we fetch relevant examples from the CSQA training set equipped with the ECQA knowledge explanations (Aggarwal et al., 2021). This setup ensures that the examples that we provide to the LLMs do not share the same distribution as the input questions for evaluation. In the table, under every dataset, the *ECQA* column reports the score of retrieving examples from the CSQA training set equipped

Question	Oracle Knowledge	Choices
The man often made smart remarks, like that any restaurant is a Mexican restaurant where?	Mexican restaurants are found in Mexico. Mexico has many Mexican places.	A. city B. mexica C. san diego D. spain E. mexico
The man in the white suit was very lazy. He did nothing useful. Meanwhile, the man in the blue suit had put in effort and was very what?	The man in the white suit was very lazy. He did nothing useful. Meanwhile, the man in the blue had put in effort and was very productive.	A. restless B. active C. lazybutt D. productive E. hard work
What could go on top of wood?	It is the carpet that could go on top of wood.	A. lumberyard B. synagogue C. floor D. carpet E. hardware store
Where could you find a toilet that only friends can use?	Your friends come to your apartment. A toilet your apartment can only be used by your friends.	A. rest area B. school C. stadium D. apartment E. hospital
The weasel was becoming a problem, it kept getting into the chicken eggs kept in the what?	The weasel was becoming a problem at the barn. The chicken eggs were kept into the barn and weasel was getting into it.	A. forest B. barn C. public office D. out of doors E. freezer

Table 9: Examples of gold explanations from ECQA. The gold answers for the questions are in **bold**.

with the explanations contained in ECQA (out-of-domain), while the *ZKB* column reports the result of retrieving examples from the related training set contained in *ZEBRA-KB* (in-domain). From the table, we can see that the out-of-domain results are lower than the in-domain ones by an average of only 1.6 points across all the LLMs, highlighting the scalability of our approach even in scenarios in which gold or silver knowledge annotations are not available for a certain domain.

A.6 Knowledge Generated with ZEBRA

We provide examples of the knowledge generated by the LLMs under the *ZEBRA* framework. With reference to Table 11, we report 5 examples of knowledge generated by Llama-3 during the knowledge generation step of *ZEBRA*. Specifically, the first four examples show instances of knowledge which were deemed *relevant*, *factual* and *helpful* by all three human annotators, while the last example shows an item of knowledge which was anno-

Model	ARC-C		ARC-E		OBQA		PIQA		QASC		CSQA2		WG		AVG	
	ECQA	ZKB	ECQA	ZKB	ECQA	ZKB	ECQA	ZKB	ECQA	ZKB	ECQA	ZKB	ECQA	ZKB	ECQA	ZKB
Mistral v2	74.4	75.2	86.9	87.4	73.6	75.8	77.6	80.2	68.6	68.3	65.6	67.5	59.1	60.7	72.3	73.6
Phi-3-Small	90.9	91.6	97.2	97.7	89.8	91.2	87.4	88.1	81.1	81.0	72.1	74.6	78.4	81.0	85.3	86.4
Llama-3	82.2	83.5	92.4	92.9	76.8	79.6	79.8	84.0	80.2	79.1	68.2	69.4	60.8	63.2	77.2	78.8
Phi-3-Mini	86.3	88.0	95.4	96.0	84.2	87.8	82.0	84.2	73.7	73.9	64.7	64.6	69.5	79.2	79.4	81.9

Table 10: Out-of-domain results. The *ECQA* column shows the score obtained by retrieving examples from the CSQA training set, augmented with the explanations provided by ECQA (out-of-domain), while the *ZKB* column represents the results of retrieving examples from the corresponding training set within ZEBRA-KB (in-domain).

tated as *relevant*, *factual* and *harmful*. The model demonstrates its capability to generate relevant and high-quality commonsense knowledge for answering questions. However, there are instances where the model produces useful knowledge that, while generally accurate from a commonsense perspective, is not directly related to the correct choice or may harm the final prediction. Consequently, this information, despite its correctness, fails to provide the information to answer the question correctly.

Question	Choices	Generated Knowledge
Where in your home would you keep a ballpoint pen when not in use?	A. backpack B. bank C. desk drawer D. eat cake E. office desk	Desk drawers are used for storing office supplies. Pens are office supplies. Desk drawers are a common place to store pens.
Where is a likely place for an ivy plant?	A. flower pot B. shelf C. windowsill D. outside E. sill	Ivy is a climbing plant that thrives outdoors. Ivy often grows on walls, fences, and other surfaces. Ivy is commonly found growing outside.
Where is a control room needed to prevent wide spread disaster?	A. airbase B. prison C. mill D. nuclear plant E. studio	Nuclear plant is a place where a disaster can spread widely. Control room is needed to prevent widespread disaster in a nuclear plant.
When a person admits his mistakes, what are they doing?	A. accident B. learn to swim C. thank god D. feel relieved E. act responsibly	Admitting mistakes means taking responsibility for one's actions. Taking responsibility is a sign of acting responsibly.
What do the feathers look like on birds found in the rainforest?	A. pretty smart B. singing C. dark D. very colourful E. light	Rainforests are known for their dense and dark environment. Dark colors help birds blend in with their surroundings.

Table 11: Examples of knowledge generated by Llama-3 during the knowledge generation step of ZEBRA. The first four examples feature explanations that have been deemed to be relevant, factual and helpful by all three human annotators; the last example features explanations that have been judged to be relevant, factual and harmful by all three human annotators. The gold answers for the questions are in **bold**.