Complex Reasoning over Logical Queries on Commonsense Knowledge Graphs

Anonymous ACL submission

Abstract

Reasoning about events, their relationships, and inferring implicit context are crucial abilities of event commonsense reasoning, which state-of-the-art language models still struggle to perform. However, data scarcity makes it challenging to learn systems that can generate commonsense inferences for contexts and questions involving interactions between complex events. To address this demand, we present COM^2 (**COM**plex **COM**monsense), a new dataset created by sampling multi-hop logical queries (e.g., *the joint effect or cause of both event A and B, or the effect of the effect of event C*) from an existing commonsense knowledge graph (CSKG), and verbalizing them using handcrafted rules and Large Language Models into multiple-choice and text generation questions.

Our experiments show that Language models trained on COM^2 exhibit significant improvements in complex reasoning ability, resulting in enhanced zero-shot performance in both in-domain and out-of-domain tasks for question answering and generative commonsense reasoning, without expensive human annotations.

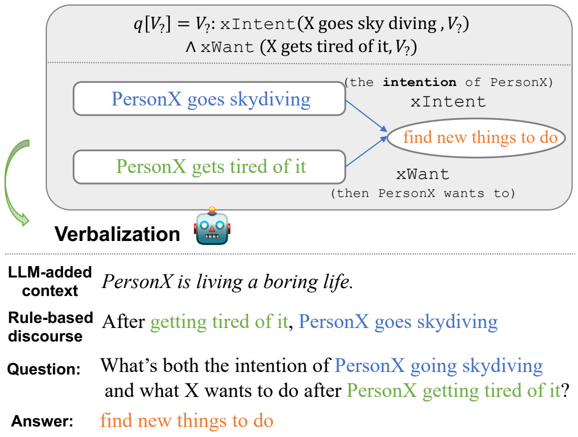

Figure 1: An example of conjunctive logical queries and the verbalization to complex commonsense inferences.

1 Introduction

Despite achieving remarkable performance in many commonsense reasoning tasks, LLMs still face challenges when it comes to more complex scenarios, such as reasoning about multiple events and their relationships, as well as inferring implicit context to facilitate subsequent reasoning. This is due to the inherent difficulty of reasoning over multiple pieces of information and a lack of adequate-scale supervised training datasets for learning (Zhao et al., 2023). Unfortunately, complex and multi-hop commonsense reasoning benchmarks (Gabriel et al., 2021) are both technically challenging and financially expensive to curate. Consequently, previous efforts either constructed datasets (a) with simpler reasoning structures, such as single-hop chains (Mostafazadeh et al., 2020), (b) using distant supervision based on one-hop inference (Gabriel et al., 2021), or (c) with human-annotations, but at relatively small scale (Ravi et al., 2023).

To alleviate this training data bottleneck, recent works have explored extracting and formulating questions from existing CommonSense Knowledge Graphs (CSKGs; Hwang et al., 2021), which store commonsense triples. However, using CSKGs to produce high-quality reasoning datasets poses several challenges. First, while the shared entities in commonsense triples encode a complex, interconnected graph structure, the sparsity of this structure limits the number of potential questions that encode more than one reasoning hop (Sap et al., 2019b; Kim et al., 2023). Second, triples in CSKGs are represented in a context-free manner, such as the event "PersonX gets tired of it" in Fig. 1, yielding ambiguous (and sometimes incorrect) human annotations in the CSKG, e.g., ATOMIC (Sap et al., 2019a) has an error rate of over 10%. These errors propagate quadratically when triples are naively combined to construct reasoning questions. Finally, also because triples in CSKGs are represented in a context-free manner, additional context must be added to make questions fluent, a problem exacerbated in multi-hop settings where the entities of multiple reasoning hops must be coherently verbalized together.

In this paper, we construct COM^2 (**COM**plex **COM**monsense), a novel commonsense reasoning dataset using multi-hop queries in commonsense knowledge graphs to construct question answer pairs requiring complex narrative reasoning to solve. To build a dataset that integrates more complex reasoning signals, we resort to *conjunctive logical queries* (Hamilton et al., 2018), a subset of First-Order Logical queries that use existential quantifiers and conjunction. The multi-hop projection operation involves inferring hidden contexts, while the intersection operation enables reasoning among multiple events, encompassing common cause or effect, and abduction. For example, in Fig. 1, an intersection of two triples can be verbalized to a short narrative, and the process of inferring the sampled common tail can be seen as an *abduction* of the hidden cause between the two heads.

To address the challenges above, we propose to first *densify* the CSKG to merge nodes with high semantic similarity, increasing the connectivity of the graph. Then, we use an off-the-shelf plausibility scorer to filter out low quality triples, avoiding error propagation as we construct more complicated queries. Finally, we verbalize the queries to a natural language context with handcrafted rules and Large Language Models to derive coherent and informative narrative contexts for our questions. Our final COM^2 dataset comprises 790K question-answer pairs (both with multiple-choice and generative answer settings), including 1.3K examples that we manually verify for evaluation.

Our results demonstrate the challenges faced by even powerful LLMs and supervised question answering models on the COM^2 dataset, underscoring the difficulty of performing complex multi-hop reasoning. Moreover, fine-tuning question answering models and generative commonsense inference models on COM^2 leads to substantial improvements across four commonsense reasoning datasets, showing the effecacy of our framework for boosting commonsense reasoning ability.

To conclude, our contributions are three-fold. First, we present a pipeline for effectively sampling and verbalizing complex logical queries from CSKGs, to form a complex commonsense reasoning benchmark, COM^2, with minimum human effort. Second, we benchmark the complex reasoning ability of various state-of-the-art language models and question answering models on COM^2. Third, we conducted comprehensive experiments to validate the beneficial impact of fine-tuning on COM^2 for subsequent commonsense reasoning tasks across eight datasets.

2 Related Work and Background

Complex Logical Queries Recent years have witnessed significant progress in reasoning on one-hop relational data (Bordes et al., 2013; Sun et al., 2019; Lin et al., 2023). In addition to one-hop reasoning, efforts are also put into handling complex logical structures, involving *reasoning on unobserved edges and multiple entities and variables* (Ren et al., 2020; Wang et al., 2021, 2023b; Bai et al., 2023a). In this paper, we focus on conjunctive logical queries (Hamilton et al., 2018), a subset of first-order logic that is defined with logical operators such as existential quantifiers $\exists$ and conjunctions $\wedge$. There is a set of anchor entities, $\mathcal{V}$, a unique target entity $V_?$ representing the answer to the query, and a set of existential quantified variables $V_1, \cdots, V_m$. Conjunctive queries are defined as the conjunction of literals $e_1, \cdots, e_n$:

$$q = V_?, \exists V_1, \cdots, V_m : e_1 \wedge e_2 \wedge \cdots \wedge e_n \quad (1)$$

where e_i is an edge involving variable nodes and anchor nodes, satisfying $e_i = r(v_j, V_k), V_k \in \{V_?, V_1, \cdots, V_m\}$, $v_j \in \mathcal{V}, r \in \mathcal{R}$, or $e_i = r(V_j, V_k), V_j, V_k \in \{V_?, V_1, \cdots, V_m\}, j \neq k, r \in \mathcal{R}$. $\mathcal{R}$ is the set of relations defined in the KB.

Previous efforts focus on constructing box embeddings (Ren et al., 2020), embeddings based on beta distribution (Ren and Leskovec, 2020), particle simulations (Bai et al., 2022), and computation tree optimization (Bai et al., 2023b). Instead of relying on embeddings or limited query types for matching synthetic logical queries, we leverage the concept of logical queries to effectively acquire complex reasoning data from CSKGs with minimum human efforts.

Complex Commonsense Reasoning Recent advances in commonsense reasoning growed starting from the construction human-annotated of CommonSense Knowledge Graphs (CSKG), including ConceptNet (Speer et al., 2017), ATOMIC (Sap et al., 2019a), ATOMIC$^{20}_{20}$ (Hwang et al., 2021), and GLUCOSE (Mostafazadeh et al., 2020). A common approach to create challenges for commonsense reasoning involves constructing tasks in the form of question-answering (Talmor et al., 2019; Sap et al., 2019b), knowledge base completion (Malaviya et al., 2020), grounding (Gao et al., 2022), and daily dialogue (Kim et al., 2023), based on CSKGs. However, most of those previous benchmarks are based on one-hop triples.

In contrast, real-world situations in dialogues and narratives usually involve more complicated reasoning across multiple events, sentences, and paragraphs (Schank and Abelson, 1975). Previous works are devoted to learn representations of narrative chains (Chambers and Jurafsky, 2008; Pichotta and Mooney, 2014) and draw inferences (Fang et al., 2022; Yuan et al., 2023). To address more complicated paragraph-level or multi-event reasoning, ParaCOMET (Gabriel et al., 2021) is proposed to pre-train on distantly supervised one-hop paragraph-level commonsense inferences, and COMET-M (Ravi et al., 2023) is proposed to be fine-tuned on a crowdsourced corpus focusing on reasoning on multiple events. Instead of crowdsourcing or using language models to distill complex inferences, we provide narrative-level inference by verbalizing complex logical queries over CSKGs, to effectively acquire grounded inferences at scale. Moreover, besides involving multiple pieces of information in the context, the question to the context also involves multiple relations.

3 Methodology

In this section, we introduce the construction details of COM^2, including pre-processing, sampling of complex queries, verbalization, and the details of human annotations.

3.1 Pre-processing

We use ATOMIC$^{20}_{20}$ (Hwang et al., 2021), a comprehensive Commonsense Knowledge Graph covering social, physical, and event-level everyday knowledge, as the base CSKG. Before sampling, we deal with the sparsity and quality issue first.

Sparsity CSKGs are usually highly sparse compared to factual KGs due to the nature of human annotation and flexibility of commonsense (Malaviya et al., 2020), making it hard to sample diverse complex queries. To alleviate the sparsity issue, we first conduct normalization to the tails. In ATOMIC, heads are pre-defined complete sentences (for example, "PersonX says sorry") while tails are usually short phrases without a subject (for example, "to say sorry"). This discrepancy produces many duplicated nodes and make the graph sparser. We develop simple rules to add "PersonX" or "PersonY" in front of the tails to make them a complete sentence, if the tail does not have a subject. This process merged 3.7% nodes in ATOMIC together.

Second, as the nodes in ATOMIC are free-text, some nodes with the same semantic meaning are represented as separated nodes due to some minor annotation distinctions and errors, e.g., "PersonX buys a ticket" versus "PersonX buys a ticket.". We use a state-of-the-art sentence embedding model[1], to merge nodes with cosine similarity score over 0.95. In this process, 20.0% nodes are merged together and the average degree increases by 25.3%.

Quality The error rate of ATOMIC itself is over 10% (Sap et al., 2019a). This error rate can be problematic when we consider the intersection and projection of more than two triples as errors propagate quadratically. We use an off-the-shelf plausibility scorer Vera (Liu et al., 2023), a 5B T5-based plausibility scorer fine-tuned on 2 CSKGs and 19 QA datasets, to score every triple in terms of commonsense plausibility (between 0 to 1). We filter out triples with a plausibility score lower than 0.5, the threshold provided as a tipping point in Vera between plausible and implausible statements. Around 10% of the triples are filtered out.

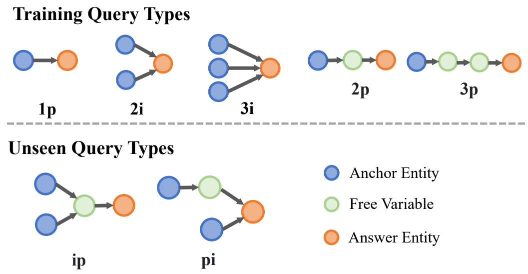

Figure 2: Visualization of query structures. The anchor entities and relations are specified to instantiate the query. 'p' and 'i' represent *projection* and *intersection*, and the number ahead of p and i indicates the number of anchor entities and free variables.

3.2 Query Sampling

The query structures that we study are visualized in Fig. 2. Following Ren et al. (2020), we use projections (1p, 2p) and intersections (2i, 3i) as training queries, and leave more complex queries ip and pi as the zero-shot evaluation queries. To examine scenarios involving negation and differentiate them

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

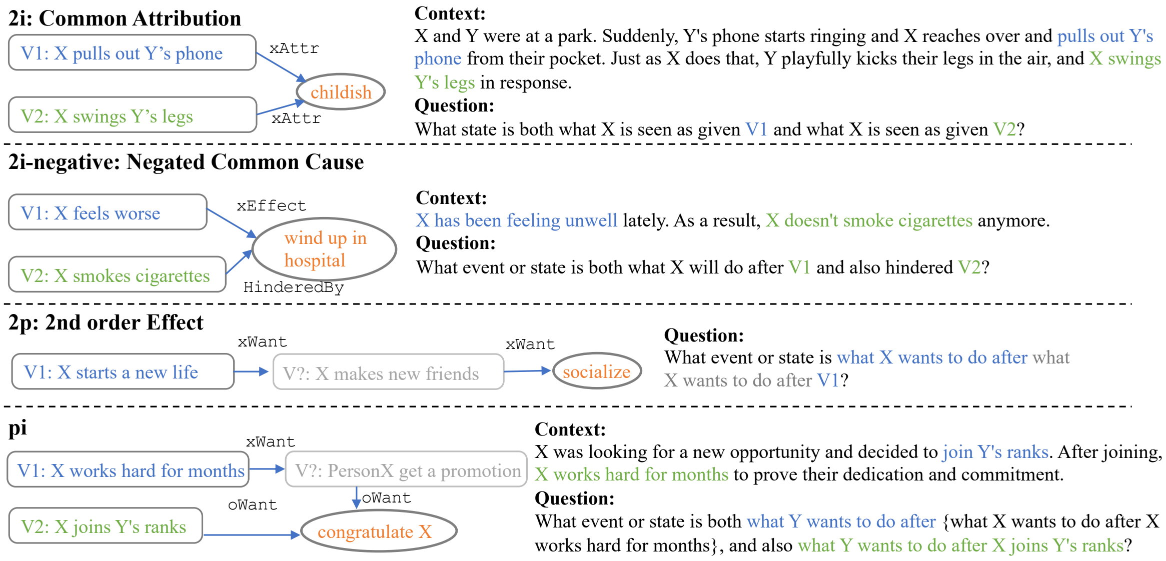

Figure 3: Examples of different query types, the verbalization, and corresponding questions.

from regular 2i queries, we use the term "2i-neg" to represent 2i queries where one of the relations is "HinderedBy".

Given a query structure, we use pre-order traversal to sample free variables and anchor entities starting from an answer entity. We sample predecessors uniformly based on (relation, entity) pairs. During sampling, to avoid over-sampling on nodes with extremely high degree, we empirically set a cut-off degree $\mathcal{T} = 10$ to only sample from top $\mathcal{T}$ neighbors of a node scored by Vera. In the end, we conduct a post-order traversal starting from the anchor entities to find all the answers of the query, in addition to the starting answer entity.

Option Sampling We sample 4 additional candidate distractors for each query, where 2 of them are randomly sampled across the whole CSKG, and 2 of them are sampled from the neighbors of the anchor entities that are not the answers to the whole query, represented as confusing negative examples. In case of fine-tuning a question answering language model, the negative examples are used as synthetic question answering pairs for training. In the evaluation set, these candidate negative examples, together with the sampled answer, are manually annotated to form a gold evaluation set.

3.3 Verbalization

CSKGs are constructed in a context-free manner. To make the logical queries on such context-free triples more human-interpretable, we introduce an additional step of verbalizing the anchor entities to a narrative, to effectively acquire fluent and plausible narrative-inference pairs.

Anchor Entity Verbalization We consider a rule-based verbalizer and a ChatGPT-driven verbalizer. In the rule-based verbalizer, we add a discourse marker between the two or three anchor entities depending on the semantics of the query relations. For example, a simple situation would be adding an "and" or "then" between two anchor entities in a 2i query. To make the query even more human-understandable, we consider using ChatGPT to synthesize necessary contexts to make the query an actual narrative. We include the detailed rules for adding discourse connectives (denoted as *rule-based verbalization*), and prompts for using ChatGPT to verbalize complex queries (denoted as LLM-based) in Appx. §A.3.

Relation Verbalization The multiple relations in complex queries can be deterministically converted to a question using the natural language descriptions of the relations, which are presended in Appx. §A.3.

3.4 Human Annotation

We formalize the problem of complex commonsense reasoning as a multi-choice question answering task, to support reliable automatic evaluation. There are only one true answer and three distractors, together with an option indicating "None of the answers are correct". We crowdsourced the answers using Amazon Mechanical Turk (AMT). The workers are given the verbalized query as the context, the corresponding question by converting the relations in the query using a prompt template, and the sampled (negative) answers. If no sampled answers are correct, then the worker is asked to se-

Method	**2i**	**2i-neg**	**3i**	**2p**	**ip**	**pi**	**All**
API-based LLMs							
`gpt-3.5-turbo-0613`	33.56	43.12	42.01	38.66	38.05	28.40	37.74
- 1-shot	43.31	35.31	58.45	57.73	51.33	62.96	48.22
- 1-shot w/ CoT	45.80	36.43	54.34	57.73	50.44	66.67	48.75
- 8-shot (2i, 2p)	48.52	41.26	57.08	67.53	53.10	74.07	53.22
- 8-shot (2i, 2p) w/ CoT	52.61	46.10	60.27	59.79	52.21	65.43	54.37
`gpt-4-1106-preview`	44.67	46.47	52.05	32.47	40.71	53.08	44.64
- 1-shot	47.85	42.01	50.68	38.66	44.25	50.62	45.63
- 1-shot w/ CoT	48.97	46.46	52.96	49.48	52.21	58.02	50.04
- 8-shot (2i, 2p)	54.87	46.47	58.90	45.88	52.21	66.67	53.00
- 8-shot (2i, 2p) w/ CoT	57.82	49.07	62.56	61.34	52.21	66.67	57.40
Open-source (QA) Language Models							
HyKAS (Ma et al., 2021, zero-shot)	34.92	39.41	27.85	41.75	37.17	33.33	35.76
CAR (Wang et al., 2023a, zero-shot)	37.41	30.48	37.44	57.73	32.74	53.09	39.56
UnifiedQA-v2 (Khashabi et al., 2022)	56.23	39.41	62.56	58.76	51.33	62.96	54.21
Flan-T5 (11B) (Chung et al., 2022)	58.28	47.21	65.30	**76.29**	56.64	79.01	60.97
Llama2 (7B) (Touvron et al., 2023)	35.15	21.93	39.27	35.57	28.32	51.85	33.64
Vera (Liu et al., 2023)	47.62	27.51	40.18	66.49	52.21	58.02	46.09
Fine-tuned on CoM^2							
DeBERTa-v3-Large (+CoM^2)	60.09	**58.36**	69.41	61.86	**59.29**	81.48	62.79
CAR-DeBERTa-v3-Large (+CoM^2)	**61.22**	56.13	**69.86**	68.56	56.64	**85.19**	**63.78**

Table 1: Model performance (%) on the multiple-choice question answering evaluation set of CoM^2.

lect an additional "None of the answers are correct" option. If the verbalization itself does not make sense, the worker can also click another option "The context doesn't make sense or is meaningless." and we will discard the data. Each question is annotated by three workers, and the overall per-option Inter Annotator Agreement (IAA) is 78%, and the fleiss kappa is 0.445, indicating moderate agreement. The workers are paid on average 16 US Dollar per hour.

We refer readers to Appx. §A for technical details and dataset statistics regarding §3.

4 Experiments

We conduct experiments on the evaluation set of CoM^2, a Multi-Choice Question Answering (MCQA) task. Specifically, we examine the performance of state-of-the-art off-the-shelf language models on CoM^2, and also study the effect of training a question answering model on the distantly supervised training set of CoM^2.

4.1 Setup

We study popular API-based LLMs and some Open-source Language Models as baselines. Following the standard practice of prompting LLMs for QA (Robinson et al., 2022), we use a prompt-based method that takes "[Context] [Question] [Options]" as the input and ask the model to only output the associated symbol (e.g., 'A') in the QA pair as the prediction. For open-source language models like Flan-T5 and Llama2, we use same prompt, and compute the logits received by each of the options in the first prediction token.

We also study the effect of fine-tuning a question-answering model on the synthetic training queries discussed in §3.2. We follow the most effective pipeline by HyKAS (Ma et al., 2021), which fine-tunes language models on QA pairs synthesized from one-hop knowledge in CSKGs, and extend it to complex queries. For one-hop (1p) triples, the head and relation are transformed into a question with pre-defined prompts. For complex queries, the verbalized queries (as illustrated in §3.3) are regarded as the context, and questions are also transformed with a different prompt template depending on the relations. The tails to the one-hop triple or the sampled answer to the query are regarded as the correct answer, and the negative examples are randomly sampled across the whole CSKG following a keyword overlapping filtering (Ma et al., 2021; Wang et al., 2023a). We use DeBERTa-v3-large as the backbone encoder[2].

4.2 Results and Analysis

We present the results in Tab. 1. In terms of performance on commercial LLMs, GPT-4 generally outperforms ChatGPT with a notable margin. The incorporation of Chain-of-Thought (CoT) proves crucial in enhancing LLM reasoning capabilities, as it fosters a step-by-step thinking approach that first focuses on inducing the causes or effects of

[2]We refer readers to Appx. §B for detailed implementations and prompt templates.

Model	CSKG	Out-of-domain						In-dom.
		a-NLI	CSQA	PIQA	SIQA	WG	Avg.	COM^2
Random	-	50.0	20.0	50.0	33.3	50.0	40.7	20.0
DeBERTa-v3-L (He et al., 2023)	-	59.9	25.4	44.8	47.8	50.3	45.6	14.7
Self-talk (Shwartz et al., 2020)	-	-	32.4	70.2	46.2	54.7	-	-
COMET-DynGen (Bosselut et al., 2021)	ATOMIC	-	-	-	50.1	-	-	-
SMLM (Banerjee and Baral, 2020)	*	65.3	38.8	-	48.5	-	-	-
MICO (Su et al., 2022)	ATOMIC	-	44.2	-	56.0	-	-	-
STL-Adapter (Kim et al., 2022)	ATOMIC	71.3	66.5	71.1	64.4	60.3	66.7	-
Large Language Models								
GPT-3.5 (`text-davinci-003`)	-	61.8	68.9	67.8	68.0	60.7	65.4	-
GPT4 (`gpt-4-1106-preview`)	-	75.0	43.0	73.0	57.0	77.0	65.0	44.6
ChatGPT (`gpt-3.5-turbo`)	-	69.3	74.5	75.1	69.5	62.8	70.2	37.7
+ zero-shot CoT	-	70.5	<u>75.5</u>	79.2	**70.7**	63.6	71.9	28.9
Backbone: DeBERTa-v3-Large *435M*								
HyKAS (Ma et al., 2021)	ATM-10X	75.1	71.6	79.0	59.7	71.7	71.4	27.7
HyKAS (Ma et al., 2021)	ATOMIC	76.0	67.0	78.0	62.1	76.0	71.8	35.8
CAR (Wang et al., 2023a)	ATOMIC	78.9	67.2	78.6	63.8	78.1	73.3	36.8
CAR (Wang et al., 2023a)	ATM^C	79.6	69.3	78.6	64.0	78.2	73.9	39.8
HyKAS + COM^2(Ours)	ATM, COM^2	78.4	69.9	78.7	64.1	78.3	73.9	62.8
CAR + COM^2(Ours)	ATM^C, COM^2	81.2	70.9	80.3	65.6	77.4	**75.1**	**63.8**
Human Performance	-	91.4	88.9	94.9	86.9	94.1	91.2	-

Table 2: Zero-shot evaluation results (%) on five out-of-domain commonsense question answering benchmarks, and the in-domain evaluation set of COM^2. The best results are **bold-faced**, and the second-best ones are <u>underlined</u>.

individual events in intersection-based queries, or inducing the hidden variables in projection-based queries. The eight-shot CoT, which encompasses both 2i and 2p queries as exemplars, yields the highest performance naturally due to the coverage of all base query types.

When it comes to fine-tuning on complex queries using the HyKAS and CAR paradigm, we observe that the synthetic training pairs, despite lacking manual annotation, serve as valuable distant supervision signals. They effectively enhance the complex reasoning capability of a QA model, even surpassing the performance of an 8-shot GPT-4 model with CoT by 6%. CAR + COM^2 can also outperform the 11B version of UnifiedQA-v2 and Flan-T5, which are both fine-tuned on numerous (commonsense) question answering datasets by 9% and 3%, respectively. We also include the zero-shot transferability experiments of this QA model to some other commonsense QA datasets, which will be presented in §5.1.

5 Downstream Evaluation

In addition to benchmarking Complex Commonsense Reasoning, we also study the effect of leveraging COM^2 as training data and the generalization to other downstream commonsense reasoning tasks. In detail, we study zero-shot CommonSense Question Answering (CSQA), and Generative Commonsense Inference, including one-hop, multi-event, and paragraph-level settings.

5.1 Commonsense Question Answering

Setup The task of zero-shot commonsense QA involves selecting the most plausible option for commonsense questions without any supervision signals from the training set of the benchmark data. We directly leverage the model we trained in §4, the DeBERTa-v3-large-based model fine-tuned on synthetic question pairs in both ATOMIC and COM^2, and check the performance on five popular commonsense question answering datasets: Abductive NLI (aNLI; Bhagavatula et al., 2020), CommonsenseQA (CSQA; Talmor et al., 2019), PhysicalIQA (PIQA; Bisk et al., 2020), SocialIQA (SIQA; Sap et al., 2019b), and WinoGrande (WG; Sakaguchi et al., 2021). We report the accuracy of each dataset and the average accuracy among five datasets.

Results and Analysis We report the model performance in Tab. 2. The first batch of baselines are zero-shot CSQA models that leverages CSKGs as supervision signals, and we surpass them by a large margin. We also report the zero-shot performance of API-based LLMs including GPT-3.5, ChatGPT, and GPT-4. The inclusion of COM^2 and

Model	Training Data	Multi-Event			Paragraph-Level			Single-Event			COM^2		
		B-2	R-L	BERT	R-L	CIDE	BERT	R-L	CIDE	BERT	R-L	CIDE	BERT
(Distantly) Supervised Learning													
COMET-M (BART-L)	MEI	25.1	33.6	64.9	-	-	-	-	-	-	-	-	-
COMET-M (GPT-2-L)	MEI	16.2	25.7	55.1	-	-	-	-	-	-	-	-	-
ParaCOMET (GPT-2-L)	ParaCOMET	-	-	-	18.8	27.8	60.2	-	-	-	-	-	-
Zero-shot Learning								**Supervised**					
COMET	1p	1.20	2.73	38.9	3.5	6.4	25.7	50.0	66.1	75.1	10.0	20.7	44.3
COMET-distill	ATM10x	1.20	3.55	12.7	11.8	16.8	29.5	1.6	4.8	24.3	8.3	11.9	36.1
COM^2-COMET	1p, 2i	8.87	15.2	46.4	13.8	22.1	53.7	50.7	68.0	77.1	13.6	26.1	39.8
COM^2-COMET	1p, 2p, 2i, 3i	5.41	10.4	44.8	9.2	16.6	44.1	50.4	66.9	77.1	**14.7**	**33.0**	**46.3**
LLama2-7b	-	1.81	4.14	45.7	2.2	2.2	48.6	5.4	2.9	51.5	3.9	6.7	44.9
COMET-LLama2-7b	1p	7.62	14.4	44.2	9.1	12.3	51.0	27.5	26.4	64.2	10.9	22.3	44.9
COM^2-LLama2-7b	1p, 2i	**8.82**	**16.4**	**47.5**	14.6	**22.1**	55.3	**31.6**	**31.1**	**66.0**	**35.7**	**107.2**	**61.3**
COM^2-LLama2-7b	1p, 2p, 2i, 3i	8.22	15.4	47.0	**15.9**	21.3	**55.3**	31.3	29.8	65.5	35.6	105.0	60.1

Table 3: Experimental results on downstream narrative commonsense reasoning, including in a multi-event (Ravi et al., 2023) setting, and a paragraph-level setting (Gabriel et al., 2021). In-domain settings include single-event generation and complex inference in COM^2. We use BLEU-2 (B-2), ROUGE-L (R-L), CIDEr (CIDE), and BERTScore (BERT) as the evaluation metrics.

one-hop triples from ATOMIC as training data for CAR and HyKAS yields significant improvements in question answering ability. This improvement is observed in both in-domain complex reasoning tasks and out-of-domain CSQA tasks. Notably, the combination of CAR and COM^2 achieves the highest performance among all models, surpassing even ChatGPT and GPT-4, despite having a parameter size at least two orders of magnitude smaller.

5.2 Generative Commonsense Inference

Setup We study generative commonsense inference as an additional evaluation task. We include multi-event commonsense generation (COMET-M; Ravi et al., 2023) and paragraph-level commonsense generation (ParaCOMET; Gabriel et al., 2021) as two out-of-domain evaluation tasks. We also include the vanilla COMET (Bosselut et al., 2019) as an additional in-domain evaluation, which actually focuses on 1p queries that requires generating the tail given head and relation as the input. Besides, we report the generation performance on generative COM^2 in the last columns.

We study the effect of fine-tuning COMET (GPT-2-large) on ATOMIC and different query types of COM^2, following the settings in Bosselut et al. (2019). We also study fine-tuning on an LLM, Llama2-7b, by converting triples and queries to an instruction-tuning format, following the prompt template in §3.3 and Appx. §B.2. We leverage the framework of Chen et al. (2023)[3] to fine-tune Llama2-7b. We fine-tune on a mixture of different query types as detailed in the "Training Data" column. To ensure diversity and prevent overfitting to common tails, complex queries are selected using an n-gram based diversity filter (Yang et al., 2020).

Results and Analysis We report the performance of various models on three datasets in Tab. 3. First, compared to fine-tune on only one-hop triples, COMET models based on both GPT2-large and Llama2-7b will have an improved generative commonsense inference ability on both multi-event, paragraph-level, and single-event commonsense inference. The first two settings are out-of-domain complex commonsense reasoning tasks that require reasoning on longer context and more complicated event-event relations. Second, among different query types, 2i is the most useful query type that help improve the reasoning ability. This may be due to the fact that both the task from COMET-M and ParaCOMET doesn't require second-order inference, while only requires the reasoning ability brought by intersection-based queries.

6 Discussions and Analysis

6.1 Ablation Study

We analyze the impact of various data filters, query types, and verbalization methods in Tab. 12 in the appendix on generative inference in COM^2.

Filtering We include two types of filters, a Vera-based plausibility filter and a diversity filter. Evaluating the performance of generative commonsense

[3]https://github.com/epfLLM

Model	#Plau.	#1-hop	#False
LLama2-7b	26	2	28
COMET-LLama2-7b	29	8	23
COM^2-LLama2-7b (2i)	47	2	11
COM^2-LLama2-7b (all)	45	3	12

Table 4: Human evaluation results on the generative sub-task in COM^2 using Llama2-7b as the backbone. '1-hop' indicates the answer is plausible in terms of only one-hop relations.

inferences on COM^2, we examine the impact of removing both filters while employing GPT2-Large as the backbone model. Removing the plausibility filter results in a significant performance decline, highlighting its critical role. On the other hand, the diversity filter exhibits a minor positive influence on enhancing performance.

Type of Queries We investigate the impact of training our models on different types of logical queries. The model trained only on 1p and 2i queries does not generalize well to other query types such as pi and ip, leading to a worse performance than the model trained on all query types. However, according to Tab. 1 and Tab. 3, models trained on only 2i queries have a better generalization ability to downstream commonsense reasoning tasks. This is probably because most existing commonsense benchmarks focusing on interactions regarding multiple events are actually structured as an intersection-based manner, instead of projections and more complicated structures.

Verbalization We investigate the effect of using a rule-based verbalizer or ChatGPT-enabled verbalizer. The ChatGPT-verbalized queries help produce better inference system a tad bit on both ParaCOMET and COM^2. In COM^2, the presence of ChatGPT-verbalization intuitively improves performance since the training context aligns with the evaluation set's format. On the other hand, the context in the ParaCOMET dataset is long and comprised of five sentences. Verbalization not only adds more contexts to the training but also aligns better with the ParaCOMET format.

6.2 Difficulty of Different Query Types

Based on Tab. 1, there is a significantly higher accuracy of pi queries than others. This is mainly because of the sparsity issue, such that we cannot sample enough pi queries. Within the limited pi queries, the number of unique answers is also small and they are usually common nodes with high degrees in the CSKG, making it easier for models to make accurate predictions. The same situation applies to 2i and 3i queries. Though 3i queries possess a more complex structure, they are constrained by the sparse structures of ATOMIC, resulting in a relatively narrower answer set. This narrower set of possible answers makes predictions easier[4].

6.3 Error Analysis

We present a human-annotated quality evaluation of the Llama-7b-based model on the generation sub-task of COM^2. To ensure diverse coverage of query types, we randomly sampled 60 queries, with 10 from each of the 6 categories. Manual inspection revealed a common error where the generated output was partially correct, either providing the answer to one of the triples in an intersection query or only the one-hop answer instead of the two-hop answer in 2-projection queries. Tab. 4 includes the number of such '1-hop' partially correct answers. Our results demonstrate that the zero-shot Llama model already produces 26 out of 60 plausible inferences. Fine-tuning the model on one-hop ATOMIC further increases the number of plausible generations while more frequently generating inferences that are one-hop correct. Moreover, fine-tuning on the synthetic training set of COM^2 significantly improves the model's ability to generate complex commonsense inferences and reduces the occurrence of partially correct answers. We leave the some case studies in the Appx. §D.

7 Conclusion

In this paper, we leverage the concept of conjunctive logical queries to create a complex commonsense reasoning dataset derived from CSKGs. The dataset, COM^2, comprises a human-annotated evaluation set and a distantly supervised training set without further annotations. Our experiments demonstrate the difficulty of answering complex logical queries on CSKGs, even for advanced language models like GPT4. Additionally, we train question answering models and generative commonsense reasoning models using the COM^2 training set. The results show significant improvements across eight downstream commonsense reasoning tasks, encompassing various aspects. This highlights the potential of leveraging CSKGs to acquire complex reasoning signals inexpensively, without relying on extra human efforts.

[4]We leave more quantative analysis in Appx. §C

Limitations

Data Construction The construction of Com^2 reply on sampling complex logical queries from existing CSKGs. However, there are sparsity issue, quality issue, non-close-world-asssumption issue that needs to be tackled. Even we have conducted normalization and filtering, there may still be missing links within ATOMIC and mislabeled or ambiguous triples, which limits the quality of our sampled queries. Future works can focus on deriving complex queries from CSKGs with better quality and more diverse semantics, which should also have higher density, such as on ATOMIC-10x, NovATOMIC (West et al., 2023).

Evaluation In the context of generative commonsense reasoning, we employ lexical-overlap based automatic evaluation metrics to assess the performance of the model in a scalable manner. However, since each query typically has 1 to 3 gold references on average, this type of evaluation may not accurately capture the true plausibility of commonsense reasoning, which is inherently open-ended. To address this limitation, we have supplemented the automatic evaluation with human annotation on a subset of sampled queries. Nevertheless, this approach is still not scalable by nature.

Future research can focus on the development of automatic complex reasoning protocols based on large language models. Such protocols can delve into more fine-grained aspects such as typicality and the degree of correctness, even if it's only partially correct.

Ethical Considerations

We sample the data from ATOMIC$^{20}_{20}$, which is an open-source commonsense knowledge graph that may contain certain bias regarding gender, occupation, and nationality (Mehrabi et al., 2021). The dataset does not contain specific individuals or organizations. Instead, it employs generic placeholders such as PersonX, PersonY, and randomly replaced first names to represent subjects and objects. However, this paper primarily focuses on complex reasoning based on knowledge, which is in contrast to works that solely rely on one-hop biased knowledge exploitation.

We collected 1.3k inferences through crowdsourcing. The participants were compensated with an hourly wage of 16 USD, which is comparable to the minimum wages in the US. The qualification was purely based on the workers' performance on the evaluation set, and we did not collect any personal information about the participants from MTurk.

References

Jiaxin Bai, Xin Liu, Weiqi Wang, Chen Luo, and Yangqiu Song. 2023a. Complex query answering on eventuality knowledge graph with implicit logical constraints. *CoRR*, abs/2305.19068.

Jiaxin Bai, Zihao Wang, Hongming Zhang, and Yangqiu Song. 2022. Query2particles: Knowledge graph reasoning with particle embeddings. In *Findings of the Association for Computational Linguistics: NAACL 2022, Seattle, WA, United States, July 10-15, 2022*, pages 2703–2714. Association for Computational Linguistics.

Yushi Bai, Xin Lv, Juanzi Li, and Lei Hou. 2023b. Answering complex logical queries on knowledge graphs via query computation tree optimization. In *International Conference on Machine Learning, ICML 2023, 23-29 July 2023, Honolulu, Hawaii, USA*, volume 202 of *Proceedings of Machine Learning Research*, pages 1472–1491. PMLR.

Pratyay Banerjee and Chitta Baral. 2020. Self-supervised knowledge triplet learning for zero-shot question answering. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing, EMNLP 2020, Online, November 16-20, 2020*. Association for Computational Linguistics.

Chandra Bhagavatula, Ronan Le Bras, Chaitanya Malaviya, Keisuke Sakaguchi, Ari Holtzman, Hannah Rashkin, Doug Downey, Wen-tau Yih, and Yejin Choi. 2020. Abductive commonsense reasoning. In *8th International Conference on Learning Representations, ICLR 2020, Addis Ababa, Ethiopia, April 26-30, 2020*. OpenReview.net.

Yonatan Bisk, Rowan Zellers, Ronan Le Bras, Jianfeng Gao, and Yejin Choi. 2020. PIQA: reasoning about physical commonsense in natural language. In *The Thirty-Fourth AAAI Conference on Artificial Intelligence, AAAI 2020, The Thirty-Second Innovative Applications of Artificial Intelligence Conference, IAAI 2020, The Tenth AAAI Symposium on Educational Advances in Artificial Intelligence, EAAI 2020, New York, NY, USA, February 7-12, 2020*. AAAI Press.

Antoine Bordes, Nicolas Usunier, Alberto García-Durán, Jason Weston, and Oksana Yakhnenko. 2013. Translating embeddings for modeling multi-relational data. In *Advances in Neural Information Processing Systems 26: 27th Annual Conference on Neural Information Processing Systems 2013. Proceedings of a meeting held December 5-8, 2013, Lake Tahoe, Nevada, United States*, pages 2787–2795.

Antoine Bosselut, Ronan Le Bras, and Yejin Choi. 2021. Dynamic neuro-symbolic knowledge graph construction for zero-shot commonsense question answering. In *Thirty-Fifth AAAI Conference on Artificial Intelligence, AAAI 2021, Thirty-Third Conference on Innovative Applications of Artificial Intelligence, IAAI 2021, The Eleventh Symposium on Educational Advances in Artificial Intelligence, EAAI 2021, Virtual Event, February 2-9, 2021*. AAAI Press.

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 Conference of the Association for Computational Linguistics, ACL 2019, Florence, Italy, July 28- August 2, 2019, Volume 1: Long Papers*. Association for Computational Linguistics.

Nathanael Chambers and Daniel Jurafsky. 2008. Unsupervised learning of narrative event chains. In *ACL 2008, Proceedings of the 46th Annual Meeting of the Association for Computational Linguistics, June 15-20, 2008, Columbus, Ohio, USA*, pages 789–797. The Association for Computer Linguistics.

Zeming Chen, Alejandro Hernández-Cano, Angelika Romanou, Antoine Bonnet, Kyle Matoba, Francesco Salvi, Matteo Pagliardini, Simin Fan, Andreas Köpf, Amirkeivan Mohtashami, Alexandre Sallinen, Alireza Sakhaeirad, Vinitra Swamy, Igor Krawczuk, Deniz Bayazit, Axel Marmet, Syrielle Montariol, Mary-Anne Hartley, Martin Jaggi, and Antoine Bosselut. 2023. MEDITRON-70B: scaling medical pretraining for large language models. *CoRR*, abs/2311.16079.

Hyung Won Chung, Le Hou, Shayne Longpre, Barret Zoph, Yi Tay, William Fedus, Eric Li, Xuezhi Wang, Mostafa Dehghani, Siddhartha Brahma, Albert Webson, Shixiang Shane Gu, Zhuyun Dai, Mirac Suzgun, Xinyun Chen, Aakanksha Chowdhery, Sharan Narang, Gaurav Mishra, Adams Yu, Vincent Y. Zhao, Yanping Huang, Andrew M. Dai, Hongkun Yu, Slav Petrov, Ed H. Chi, Jeff Dean, Jacob Devlin, Adam Roberts, Denny Zhou, Quoc V. Le, and Jason Wei. 2022. Scaling instruction-finetuned language models. *CoRR*, abs/2210.11416.

Biaoyan Fang, Timothy Baldwin, and Karin Verspoor. 2022. What does it take to bake a cake? the recipe-ref corpus and anaphora resolution in procedural text. In *Findings of the Association for Computational Linguistics: ACL 2022, Dublin, Ireland, May 22-27, 2022*, pages 3481–3495. Association for Computational Linguistics.

Saadia Gabriel, Chandra Bhagavatula, Vered Shwartz, Ronan Le Bras, Maxwell Forbes, and Yejin Choi. 2021. Paragraph-level commonsense transformers with recurrent memory. In *Thirty-Fifth AAAI Conference on Artificial Intelligence, AAAI 2021, Thirty-Third Conference on Innovative Applications of Artificial Intelligence, IAAI 2021, The Eleventh Symposium on Educational Advances in Artificial Intelligence, EAAI 2021, Virtual Event, February 2-9, 2021*, pages 12857–12865. AAAI Press.

Silin Gao, Jena D. Hwang, Saya Kanno, Hiromi Wakaki, Yuki Mitsufuji, and Antoine Bosselut. 2022. Comfact: A benchmark for linking contextual commonsense knowledge. In *Findings of the Association for Computational Linguistics: EMNLP 2022, Abu Dhabi, United Arab Emirates, December 7-11, 2022*. Association for Computational Linguistics.

William L. Hamilton, Payal Bajaj, Marinka Zitnik, Dan Jurafsky, and Jure Leskovec. 2018. Embedding logical queries on knowledge graphs. In *Advances in Neural Information Processing Systems 31: Annual Conference on Neural Information Processing Systems 2018, NeurIPS 2018, December 3-8, 2018, Montréal, Canada*, pages 2030–2041.

Pengcheng He, Jianfeng Gao, and Weizhu Chen. 2023. DeBERTav3: Improving deBERTa using ELECTRA-style pre-training with gradient-disentangled embedding sharing. In *The Eleventh International Conference on Learning Representations*.

Jena D. Hwang, Chandra Bhagavatula, Ronan Le Bras, Jeff Da, Keisuke Sakaguchi, Antoine Bosselut, and Yejin Choi. 2021. (comet-) atomic 2020: On symbolic and neural commonsense knowledge graphs. In *Thirty-Fifth AAAI Conference on Artificial Intelligence, AAAI 2021, Thirty-Third Conference on Innovative Applications of Artificial Intelligence, IAAI 2021, The Eleventh Symposium on Educational Advances in Artificial Intelligence, EAAI 2021, Virtual Event, February 2-9, 2021*. AAAI Press.

Daniel Khashabi, Yeganeh Kordi, and Hannaneh Hajishirzi. 2022. Unifiedqa-v2: Stronger generalization via broader cross-format training. *CoRR*, abs/2202.12359.

Hyunwoo Kim, Jack Hessel, Liwei Jiang, Peter West, Ximing Lu, Youngjae Yu, Pei Zhou, Ronan Le Bras, Malihe Alikhani, Gunhee Kim, Maarten Sap, and Yejin Choi. 2023. SODA: million-scale dialogue distillation with social commonsense contextualization. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing, EMNLP 2023, Singapore, December 6-10, 2023*, pages 12930–12949. Association for Computational Linguistics.

Yu Jin Kim, Beong-woo Kwak, Youngwook Kim, Reinald Kim Amplayo, Seung-won Hwang, and Jinyoung Yeo. 2022. Modularized transfer learning with multiple knowledge graphs for zero-shot commonsense reasoning. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL 2022, Seattle, WA, United States, July 10-15, 2022*. Association for Computational Linguistics.

Qika Lin, Rui Mao, Jun Liu, Fangzhi Xu, and Erik Cambria. 2023. Fusing topology contexts and logical rules in language models for knowledge graph completion. *Inf. Fusion*, 90:253–264.

Jiacheng Liu, Wenya Wang, Dianzhuo Wang, Noah A. Smith, Yejin Choi, and Hannaneh Hajishirzi. 2023. Vera: A general-purpose plausibility estimation model for commonsense statements. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing, EMNLP 2023, Singapore, December 6-10, 2023*, pages 1264–1287. Association for Computational Linguistics.

Kaixin Ma, Filip Ilievski, Jonathan Francis, Yonatan Bisk, Eric Nyberg, and Alessandro Oltramari. 2021. Knowledge-driven data construction for zero-shot evaluation in commonsense question answering. In *Thirty-Fifth AAAI Conference on Artificial Intelligence, AAAI 2021, Thirty-Third Conference on Innovative Applications of Artificial Intelligence, IAAI 2021, The Eleventh Symposium on Educational Advances in Artificial Intelligence, EAAI 2021, Virtual Event, February 2-9, 2021*. AAAI Press.

Chaitanya Malaviya, Chandra Bhagavatula, Antoine Bosselut, and Yejin Choi. 2020. Commonsense knowledge base completion with structural and semantic context. In *The Thirty-Fourth AAAI Conference on Artificial Intelligence, AAAI 2020, The Thirty-Second Innovative Applications of Artificial Intelligence Conference, IAAI 2020, The Tenth AAAI Symposium on Educational Advances in Artificial Intelligence, EAAI 2020, New York, NY, USA, February 7-12, 2020*. AAAI Press.

Ninareh Mehrabi, Pei Zhou, Fred Morstatter, Jay Pujara, Xiang Ren, and Aram Galstyan. 2021. Lawyers are dishonest? quantifying representational harms in commonsense knowledge resources. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing, EMNLP 2021, Virtual Event / Punta Cana, Dominican Republic, 7-11 November, 2021*, pages 5016–5033. Association for Computational Linguistics.

Nasrin Mostafazadeh, Aditya Kalyanpur, Lori Moon, David W. Buchanan, Lauren Berkowitz, Or Biran, and Jennifer Chu-Carroll. 2020. GLUCOSE: generalized and contextualized story explanations. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing, EMNLP 2020, Online, November 16-20, 2020*. Association for Computational Linguistics.

Karl Pichotta and Raymond J. Mooney. 2014. Statistical script learning with multi-argument events. In *Proceedings of the 14th Conference of the European Chapter of the Association for Computational Linguistics, EACL 2014, April 26-30, 2014, Gothenburg, Sweden*, pages 220–229. The Association for Computer Linguistics.

Sahithya Ravi, Raymond Ng, and Vered Shwartz. 2023. COMET-M: reasoning about multiple events in complex sentences. In *Findings of the Association for Computational Linguistics: EMNLP 2023, Singapore, December 6-10, 2023*, pages 12921–12937. Association for Computational Linguistics.

Hongyu Ren, Weihua Hu, and Jure Leskovec. 2020. Query2box: Reasoning over knowledge graphs in vector space using box embeddings. In *8th International Conference on Learning Representations, ICLR 2020, Addis Ababa, Ethiopia, April 26-30, 2020*. OpenReview.net.

Hongyu Ren and Jure Leskovec. 2020. Beta embeddings for multi-hop logical reasoning in knowledge graphs. In *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual*.

Joshua Robinson, Christopher Michael Rytting, and David Wingate. 2022. Leveraging large language models for multiple choice question answering. *CoRR*, abs/2210.12353.

Rachel Rudinger, Vered Shwartz, Jena D. Hwang, Chandra Bhagavatula, Maxwell Forbes, Ronan Le Bras, Noah A. Smith, and Yejin Choi. 2020. Thinking like a skeptic: Defeasible inference in natural language. In *Findings of the Association for Computational Linguistics: EMNLP 2020, Online Event, 16-20 November 2020*, volume EMNLP 2020 of *Findings of ACL*, pages 4661–4675. Association for Computational Linguistics.

Keisuke Sakaguchi, Ronan Le Bras, Chandra Bhagavatula, and Yejin Choi. 2021. Winogrande: an adversarial winograd schema challenge at scale. *Commun. ACM*, 64(9).

Maarten Sap, Ronan Le Bras, Emily Allaway, Chandra Bhagavatula, Nicholas Lourie, Hannah Rashkin, Brendan Roof, Noah A. Smith, and Yejin Choi. 2019a. ATOMIC: an atlas of machine commonsense for if-then reasoning. In *The Thirty-Third AAAI Conference on Artificial Intelligence, AAAI 2019, The Thirty-First Innovative Applications of Artificial Intelligence Conference, IAAI 2019, The Ninth AAAI Symposium on Educational Advances in Artificial Intelligence, EAAI 2019, Honolulu, Hawaii, USA, January 27 - February 1, 2019*. AAAI Press.

Maarten Sap, Hannah Rashkin, Derek Chen, Ronan Le Bras, and Yejin Choi. 2019b. Social iqa: Commonsense reasoning about social interactions. 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 2019, Hong Kong, China, November 3-7, 2019*. Association for Computational Linguistics.

Roger C. Schank and Robert P. Abelson. 1975. Scripts, plans and knowledge. In *Advance Papers of the Fourth International Joint Conference on Artificial Intelligence, Tbilisi, Georgia, USSR, September 3-8, 1975*, pages 151–157.

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 2020, Online, November 16-20, 2020. Association for Computational Linguistics.

Robyn Speer, Joshua Chin, and Catherine Havasi. 2017. Conceptnet 5.5: An open multilingual graph of general knowledge. In *Proceedings of the Thirty-First AAAI Conference on Artificial Intelligence, February 4-9, 2017, San Francisco, California, USA*. AAAI Press.

Ying Su, Zihao Wang, Tianqing Fang, Hongming Zhang, Yangqiu Song, and Tong Zhang. 2022. MICO: A multi-alternative contrastive learning framework for commonsense knowledge representation. In *Findings of the Association for Computational Linguistics: EMNLP 2022, Abu Dhabi, United Arab Emirates, December 7-11, 2022*. Association for Computational Linguistics.

Zhiqing Sun, Zhi-Hong Deng, Jian-Yun Nie, and Jian Tang. 2019. Rotate: Knowledge graph embedding by relational rotation in complex space. In *7th International Conference on Learning Representations, ICLR 2019, New Orleans, LA, USA, May 6-9, 2019*. OpenReview.net.

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, NAACL-HLT 2019, Minneapolis, MN, USA, June 2-7, 2019, Volume 1 (Long and Short Papers)*. Association for Computational Linguistics.

Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, Dan Bikel, Lukas Blecher, Cristian Canton-Ferrer, Moya Chen, Guillem Cucurull, David Esiobu, Jude Fernandes, Jeremy Fu, Wenyin Fu, Brian Fuller, Cynthia Gao, Vedanuj Goswami, Naman Goyal, Anthony Hartshorn, Saghar Hosseini, Rui Hou, Hakan Inan, Marcin Kardas, Viktor Kerkez, Madian Khabsa, Isabel Kloumann, Artem Korenev, Punit Singh Koura, Marie-Anne Lachaux, Thibaut Lavril, Jenya Lee, Diana Liskovich, Yinghai Lu, Yuning Mao, Xavier Martinet, Todor Mihaylov, Pushkar Mishra, Igor Molybog, Yixin Nie, Andrew Poulton, Jeremy Reizenstein, Rashi Rungta, Kalyan Saladi, Alan Schelten, Ruan Silva, Eric Michael Smith, Ranjan Subramanian, Xiaoqing Ellen Tan, Binh Tang, Ross Taylor, Adina Williams, Jian Xiang Kuan, Puxin Xu, Zheng Yan, Iliyan Zarov, Yuchen Zhang, Angela Fan, Melanie Kambadur, Sharan Narang, Aurélien Rodriguez, Robert Stojnic, Sergey Edunov, and Thomas Scialom. 2023. Llama 2: Open foundation and fine-tuned chat models. *CoRR*, abs/2307.09288.

Weiqi Wang, Tianqing Fang, Wenxuan Ding, Baixuan Xu, Xin Liu, Yangqiu Song, and Antoine Bosselut. 2023a. CAR: Conceptualization-augmented reasoner for zero-shot commonsense question answering. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 13520–13545, Singapore. Association for Computational Linguistics.

Zihao Wang, Yangqiu Song, Ginny Y. Wong, and Simon See. 2023b. Logical message passing networks with one-hop inference on atomic formulas. In *The Eleventh International Conference on Learning Representations, ICLR 2023, Kigali, Rwanda, May 1-5, 2023*. OpenReview.net.

Zihao Wang, Hang Yin, and Yangqiu Song. 2021. Benchmarking the combinatorial generalizability of complex query answering on knowledge graphs. In *Proceedings of the Neural Information Processing Systems Track on Datasets and Benchmarks 1, NeurIPS Datasets and Benchmarks 2021, December 2021, virtual*.

Peter West, Ronan Bras, Taylor Sorensen, Bill Lin, Liwei Jiang, Ximing Lu, Khyathi Chandu, Jack Hessel, Ashutosh Baheti, Chandra Bhagavatula, and Yejin Choi. 2023. NovaCOMET: Open commonsense foundation models with symbolic knowledge distillation. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 1127–1149, Singapore. Association for Computational Linguistics.

Yiben Yang, Chaitanya Malaviya, Jared Fernandez, Swabha Swayamdipta, Ronan Le Bras, Ji-Ping Wang, Chandra Bhagavatula, Yejin Choi, and Doug Downey. 2020. G-daug: Generative data augmentation for commonsense reasoning. In *Findings of the Association for Computational Linguistics: EMNLP 2020, Online Event, 16-20 November 2020*, volume EMNLP 2020 of *Findings of ACL*, pages 1008–1025. Association for Computational Linguistics.

Siyu Yuan, Jiangjie Chen, Ziquan Fu, Xuyang Ge, Soham Shah, Charles Robert Jankowski, Yanghua Xiao, and Deqing Yang. 2023. Distilling script knowledge from large language models for constrained language planning. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), ACL 2023, Toronto, Canada, July 9-14, 2023*, pages 4303–4325. Association for Computational Linguistics.

Wenting Zhao, Mor Geva, Bill Yuchen Lin, Michihiro Yasunaga, Aman Madaan, and Tao Yu. 2023. Complex reasoning in natural languag. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics: Tutorial Abstracts, ACL 2023, Toronto, Canada, July 9-14, 2023*, pages 11–20. Association for Computational Linguistics.

A Additional Details on Data Construction

In this section, we provide additional details to node normalization, plausibility filter, verbalization, and human annotations. The overview of our construction framework is presented in Fig. 4.

A.1 Nodes Normalization

We present the normalization rules in Tab. 5. For example, a tail of "to go" under the relation xWant will be transformed to "PersonX go". A tail of "satisfied" under the relation xAttr will be transformed to "PersonX is satisfied".

Relations	Mapping rules
xWant/oWant/ xIntent/xNeed	Add PersonX/Y in front of the tail and remove the initial "to"
xEffect/oEffect	Add PersonX/Y in front of the tail
xReact/oReact	Add PersonX/Y and "is" in front of the tail
xAttr	Add a PersonX/Y and "is" in front of the tail

Table 5: Normalization rules for ATOMIC tails.

A.2 Data Filtering

Plausibility Filter We verbalize a (h, r, t) triple from ATOMIC using the default template as provided in Hwang et al. (2021). For example, (PersonX repels PersonY's attack, xAttr, brave) would be transformed to a declarative statement "If PersonX repels PersonY's attack, then PersonX is seen as brave". To obtain a plausibility score, we input the statement into the Vera-5B model. 0.5 is used as the threshold to draw a boundary between plausible and implausible statements. We perform a manual inspection on the triples scored by Vera and randomly select 40 samples for three plausibility score intervals. Among these, we find that 4/40 triples are plausible when the Vera scores range from 0 to 0.1. 13/40 triples are considered plausible within the score range of 0.2 to 0.25. Furthermore, we identify 20/40 triples as plausible when their plausibility scores hover around 0.5, when most of the triples are quite ambiguous. By setting the filter threshold as 0.5, we filter out around 14% triples that are of a relatively lower quality.

Diversity Filter To prevent overfitting to common tails, we conduct a diversity-based filter to acquire diverse queries for training. We take inspirations from G-DAUG (Yang et al., 2020), to use a simple greedy algorithm to iteratively select

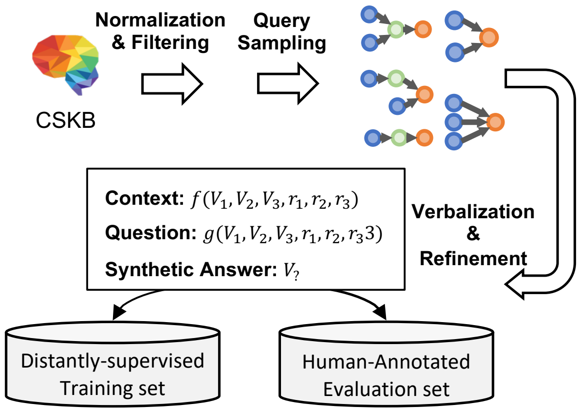

Figure 4: Overview of the construction process.

training data, which has been proven useful for selecting augmented data. To be more specific, for each unique answer, we adopt an iterative approach to select the verbalized query that contributes the highest number of unique 1-gram terms to an ongoing vocabulary constructed for each answer. We select top-20 queries for each unique answer entity.

A.3 Verbalization

Query Verbalization We employ two methods to verbalize complex queries: a rule-based method and a ChatGPT-based method.

In the case of 2i and 3i queries, the rule-based method typically involves inserting an "and" between the anchor entities. However, if the query suggests a specific chronological order between the two events, we use "then" to connect the events. For instance, in 2i queries where one triple is (V_1, xEffect, $V_?$) and the other is (V_2, xIntent, $V_?$), it implies that $V_?$ serves as the effect of V_1 and the intermediate hidden cause of V_2. In this scenario, V_1 should occur before V_2. Therefore, the verbalization would be "V_1 then V_2".

For ChatGPT verbalization, we present the system instructions for verbalizing different kinds of queries in Tab. 6. Then, we generate the verbalized contexts with six exemplars that are manually annotated. In the system instruction, we also ask ChatGPT to output "NA" if the given anchor entities are totally irrelevant or too ambiguous. We filter out those queries where the output is "NA".

For example, to better interpret the query in Fig. 1, we need to take into consideration both the relations of interest and the anchor entities. The query asks about the effect of the first event and what causes (intention) of the second event, which is inherently represents *abductive reasoning*. This

Query	Prompt
2i, ip, pi	Given two events, come up with concise and necessary context to make the a coherent and understandable narrative. No more than 2 additional piece of context should be added. If the one of the given events itself is ambiguous and hardly make sense even with extra context, return NA. If the two events are totally irrelevant even with additional context, then simply return NA. If the given two events can be directly composed to a narrative with simple a discourse connective without additional context, then there's not need to add additional context.\nMark the location of both events with <E1></E1> for event 1 and <E2></E2> for event 2 in the generated narrative.
2i-neg	Given two events, create a cohesive narrative by incorporating event 1 (E1) and negated event 2 (E2) to make the a coherent and understandable narrative. No more than 2 additional piece of context should be added. If the one of the given events itself is ambiguous and hardly make sense even with extra context, return NA. If the two events are totally irrelevant even with additional context, then simply return NA. If the given two events can be directly composed to a narrative with simple a discourse connective without additional context, then there's not need to add additional context.\nMark the location of both events with <E1></E1> for event 1 and <E2></E2> for event 2 in the generated narrative.\nDon't explain the reasons why E2 didn't happen!!\nRemember that negating an event means stating that it did not occur. For instance, if event 2 is "PersonX goes shopping," the negated form would be "PersonX didn't go shopping".

Table 6: System instructions for verbalizing complex queries given different query types.

requires the second event to happen before the first event, to derive reasonable abduction. In this sense, a natural rule of verbalizing the query would be adding a discourse connective "after" to convert the query to "After PersonX gets tired of it, PersonX goes skydiving". However, the verbalized query may still be ambiguous without additional context. To make the verbalized context more informative and human-understandable, we take advantage of Large Language Models (i.e., ChatGPT) to add additional context to compose the query to a narrative.

Relation Verbalization We use conversion rules and pre-defined templates to compose questions based on the relations in the queries. Based on the definition of each commonsense relation (Hwang et al., 2021), we use the templates in Tab. 7 to verbalize each relation. In terms of complex queries, we use the conversion rules in Tab. 8 to convert the query to a question.

Person Names To make the context more natural, we replace PersonX, PersonY, PersonZ in the context to names randomly sampled from the 2021 public US social security application name registry[5].

A.4 Human Annotation

We introduce the details of the annotation process in this subsection.

[5]https://catalog.data.gov/dataset/baby-names-from-social-security-card-applications-national-data

Query Type	Question Template
2i	What event or state is both Prompt(r1) [V1] and also prompt(r2) [V2]?
3i	What event or state is both Prompt(r1) [V1], Prompt(r2) [V2], and also Prompt(r2) [V2]?
2p	What event or state is Prompt(r1) {Prompt(r2) [V1]}?
ip	What event or state is prompt(r3) {both prompt(r1) [V1], and also prompt(r2) [V2] }?
pi	What event or state is both prompt(r1) {prompt(r3) [V3]}, and also prompt(r2) [V2]?

Table 7: Templates for verbalizing one-hop relations.

Worker Selection We have a qualification test to select eligible workers for the main task. We prepare six pre-selected 2i queries of different types, including (negated) common effect, (negated) common cause, common attribute, and abduction. We compare the pair-wise annotation accuracy between each annotator and the gold answer annotated by the authors of the paper, and select those who have at least 85% agreement as qualified workers. After selection, we pick 53 worker out of 120 participants in the qualification round.

Annotation Interface A snapshot of the annotation interface is presented at Fig. 5. In addition, we have provided comprehensive instructions along with detailed examples to guide the annotators throughout the annotation process. To ensure their understanding, we require annotators to confirm that they have thoroughly read the instructions by checking a checkbox before the annotation task. We also manually checked the performance of the annotators along with the annotation process and

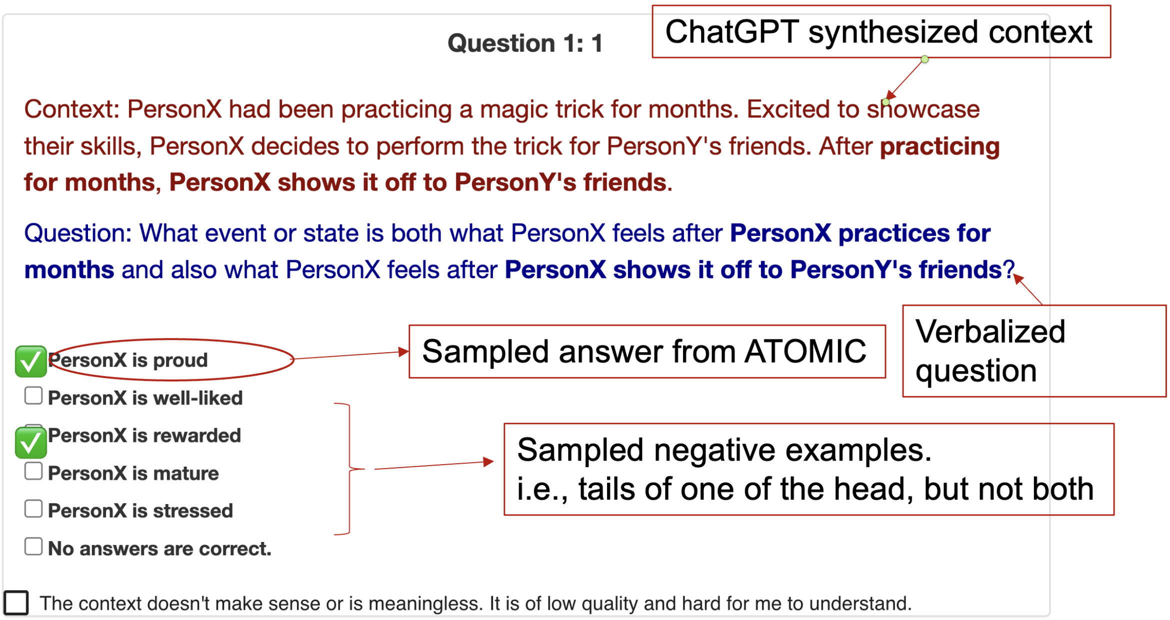

Figure 5: Annotation interface.

Relation	Prompt Template
xIntent	the intention of PersonX before
xNeed	what PersonX needed to do before
xWant	what PersonX wants to do after
xEffect	the effect on PersonX after
xReact	what PersonX feels after
xAttr	what PersonX is seen as given
oEffect	the effect on PersonY after
oReact	what PersonY feels after
oWant	what PersonY wants to do after
HinderedBy	what hindered
isAfter	what happens before
isBefore	what happens after

Table 8: Templates for verbalizing relations in complex queries.

gave feedbacks based on common errors. For example, typical errors include mistakenly regard the one-hop answer as correct instead of fully considering the multi-hop context.

Post-processing To aggregate the annotation result, we randomly sample one option that is labeled as plausible by majority voting as the final positive answer, and sample three negative options and distractors. If there are no options labeled as plausible, then the correct answer is "None of the answers are correct". If there are less than three options labeled as negative, we manually add one or two negative examples to match the number. To improve the quality, after crowdsourcing, the authors of this paper manually checked the QA pairs with an IAA lower than 0.6, and resolve the disagreements manually.

Tab. 9 presents the statistics of the training and evaluation set.

	Training	Evaluation
#Instances	782,140	1,317

Table 9: Statistics of COM^2.

B Additional Details of Experiments

B.1 Implementation Details of the Question Answering Models

We follow the pipeline in HyKAS (Ma et al., 2021) and CAR (Wang et al., 2023a) Let C represent the original context, which is the head entity for 1p triple and the verbalized context for complex queries, Q represent the question verbalized from the anchor relations, and $(A_1, A_2, \ldots)$ be the list of options. We first concatenate C, Q, and an answer option A_i together via natural language prompts following the order of "C Q A_i" to generate input sequences $(T_1, T_2, \ldots)$. We then repeatedly mask out one token at a time to calculate the masked language modeling loss.

$$\mathcal{S}(T) = -\frac{1}{n}\sum_{i=1}^{n} \log P(t_i|\ldots, t_{i-1}, t_{i+1}, \ldots) \quad (2)$$

We then compute the marginal ranking loss based on Equation 3, where η represents the margin

Model	Prompt
Llama2, Flan-T5 ChatGPT, GPT-4	Answer this commonsense reasoning question, where you are supposed to handle a multiple-chioce question answering task to select the correct answer. Select one correct answer from A to E.\n Context: [Context] Question: [Question] A: [Option A]. B: [Option B]. C: [Option C]. D: [Option D]. E: [Option E]. \n Answer:
UnifiedQA	[Question] \n (a): [Option A] (b) [Option B] (c) [Option C] (d) [Option D] (e) [Option E] \n [Context]
Vera	[Context] [Question] [Option]
HyKAS, CAR	[Context] [Question] [Option]

Table 10: Prompt templates for multiple-choice question answering.

<table>
<tr><th>Model</th><th>Prompt</th></tr>
<tr><td>Llama2 (zero-shot)</td><td>[System_Message] = As an expert in commonsense reasoning, your task is to provide a concise response to a question based on the given context. The question focuses on studying the causes, effects, or attributes of personas related to the given context. Answer shortly with no more than 5 words.

<s>[INST] <<SYS>>\n[System_Message] \n<</SYS>>\n\n[Context] [Question] [/INST]</td></tr>
<tr><td>Llama2 (fine-tuned)</td><td><|im_start|>question\n[Context] [Question] <|im_end|>\n<|im_start|>answer\n[Answer]</td></tr>
<tr><td>GPT-2</td><td>2i: [V1] [V2] [r1] [r2] [GEN] [Answer]
3i: [V1] [V2] [V3] [r1] [r2] [r3] [GEN] [Answer]
2p: [V1] [r1] [r2] [GEN] [Answer]</td></tr>
</table>

Table 11: Prompts for fine-tuning generative commonsense inference models.

and y is the index of the correct answer.

$$\mathcal{L} = \frac{1}{m}\sum_{i=1, i\neq y}^{m} \max(0, \eta - S_y + S_i) \qquad (3)$$

We train the DeBERTa QA model for 1 epoch with a learning rate of 5e-6 and a linear learning rate decay. The checkpoint that yields the best performance on the synthetic validation set in CAR (Wang et al., 2023a) or HyKAS (Ma et al., 2021) is selected as the final model. During evaluating, we select the option that yields the lowest score as the final prediction.

We provide the prompt templates for each model in Tab. 10.

B.2 Implementation Details of Generative Commonsense Inference Models

The training and evaluation of GPT2-based model is based on the paradigm defined in COMET (Bosselut et al., 2019). The input of one-hop ATOMIC triples is serialized to "h r" and the expected output is t, where (h, r, t) forms a triple in the CSKG. The input of 2p queries, (h, r_1, V) and (V, r_2, $V_?$), are serialized as "h r_1 r_2" and the expected output is $V_?$. The input of 2i queries, which includes (h_1, r_1, $V_?$) and (h_2, r_2, $V_?$), is serialized as "h_1 h_2 r_1 r_2" with the expected output as $V_?$. All models are fine-tuned for 3 epochs with a batch size of 32, a learning rate of 1e-5, a linear learning rate decay. The last checkpoint is taken as the final model.

For Llama2, we follow the standard instruction tuning procedure and use the pipeline provided by Chen et al. (2023). We train the model with a batch size of 32, learning rate of 1e-5, and linear learning rate decay. We take the final checkpoint as our model to make prediction.

The whole list of prompt templates that we use is presented in Tab. 11.

C Additional Analysis

Differences from ParaCOMET and COMET-M In ParaCOMET, the task involves providing a narrative as input, requiring the model to determine the commonsense causes or effects of a specific sentence within the context. To generate training data, a single-hop COMET model fine-tuned on ATOMIC is employed to create synthetic inferences. These inferences are generated solely based on the target sentence and the desired relation, without accessing the whole context. The resulting one-hop synthetic inferences are then utilized as distant supervision signals during the fine-tuning process for ParaCOMET.

COMET-M utilizes a context consisting of a sentence containing multiple events. Unlike from a sentence level, COMET-M focuses on generating commonsense inferences based on a specific event within the sentence. T his fine-grained approach enables more precise and detailed commonsense reasoning.

In contrast, our complex commonsense reasoning benchmark introduces additional complexities compared to ParaCOMET and COMET-M. Besides the complex structures in the context that involves multiple events, the desired relation or question involves multi-hop reasoning as well. For instance, rather than focusing on the cause of a single sentence or event, COM^2 explores questions related to common causes, effects, attributions of multiple events, and two-hop inferences. This distinctive formulation sets our work apart and poses a greater challenge for LLMs to effectively reason and provide accurate responses.

Discussions on different query types According to the main experiments on the MCQA version of COM^2 in Tab. 1, there are some variance regarding the performance on different query types. A notable distinction is the performance of pi queries, which exhibits a significantly higher success rate compared to other query types, particularly ip queries, as both pi and ip involve a single free variable and both intersection and projection operations. We present two perspectives to explain this phenomenon. First, the limited availability of sampled pi queries restricts the diversity of the data. Out of all the queries sampled from the dev set of ATOMIC$^{20}_{20}$, only 4k are pi queries, while there are 12k ip queries and 598k 2i queries. This paucity of pi queries contributes to a lack of variety. Moreover, within these 4k pi queries, the number of unique answers is limited to 459, indicating a limited range of possible responses. As a result, models fine-tuned on ATOMIC can generate answers to pi queries with relative ease, given that most of them consist of nodes with high degrees. Second, the chances of the sampled answer is actually the correct answer to pi queries (67.8%) is significantly higher than other query types (e.g., 47.2% for ip). This is also a result of the first reason, as the answers to the sampled queries are limited to nodes with high degrees.

In all, despite that the query structure itself is more complicated, the reasoning difficulty is not that hard compared to other query types due to the above two reasons.

Results of the Ablations We present the results of the ablation study in Tab. 12.

Discussions on Further Applications of Complex Queries Intuitively, 2i queries can represent various scenarios such as common attribution, common effect, common cause, and abduction (when one relation pertains to effects and the other relates to cause), depending on the types of relations involved in the query. Besides, complex logical queries, particularly those involving intersection operations, are relevant to defeasible reasoning (Rudinger et al., 2020), where inferences can be weakened given new evidence. In the one-hop setting, tails are annotated in a context-free manner, considering only the most general cases. However, in intersection-based queries like 2i and 3i, additional anchor entities and relations act as specific constraints, narrowing down the inferences to a particular scope while disregarding other commonsense inferences in the context-free scenario. For instance, in the example from Fig. 1, other potential tails for (PersonX goes skydiving, xIntent) could include overcoming fear, seeking enjoyment, or achieving a personal milestone. Nevertheless, when constrained by another query (PersonX gets tired of it, xWant), the intentions related to fear, enjoyment, and fulfillment are weakened, and only the correct inference of “finding new things to do” remains.

D Error Analysis

We present some error cases in Tab. 4. In general, a common error in both projection and intersection queries is that the generated answer can be only the one-hop answer instead of the correct answer that is multi-hop. For example, in the 2p case, “get a new job” is a direct intention of someone who updates his or her resume. However, the 2p query asks about the intention of the intention, which requires inducing the intention behind “get a new job”. In this sense, “to be financially independent” is more plausible inference. In the case of 2i queries, the error lies in the absence of inferential gaps between the context, where the generated an-

Model	Com^2 R-L	Com^2 CIDEr	Com^2 BERT
Filter			
Com^2-COMET	14.7	33.0	46.3
- w/o plau. filter	13.0	31.2	42.3
- w/o div. filter	14.4	32.5	45.8
- w/o both filter	12.5	30.3	40.1
Query Types			
COMET (1p)	10.0	20.7	44.3
+ 2i	13.6	26.1	39.8
+ 2p	9.8	19.9	43.4
+ 2i, 3i, 2p	14.7	33.0	46.3
Verbalization			
Com^2-COMET	13.6	26.1	39.8
Com^2-COMET (V)	14.3	27.1	43.4
Com^2-Llama	35.7	107.2	61.3
Com^2-Llama (V)	36.2	105.4	61.4

Model	ParaCOMET R-L	ParaCOMET CIDEr	ParaCOMET BERT
Verbalization			
Com^2-COMET	13.8	22.1	53.7
Com^2-COMET (V)	14.0	23.2	54.0
Com^2-Llama	14.6	22.1	55.3
Com^2-Llama (V)	14.8	23.6	55.5

Table 12: Ablation studies on filters, type of queries, and using ChatGPT for verbalizing queries (denoted as V).

swers become paraphrases of the events rather than being the result by any anchor entity. In the case of ip, a common error for one-hop COMET is the generation of "None" for complex cases, indicating a deficiency in multi-hop reasoning capabilities.

Type	Context	Question	COMET	COM^2-COMET
2p	Ezra updates Ezra's resume (V1)	What event or state is the intention of Ezra before the intention of Ezra before V1?	get a new job ✗ (one-hop correct)	be financially independent ✓
2i-neg	Every day, Benjamin goes to work diligently (V1), never missing a day. They are dedicated and committed to their job. In particular, Benjamin doesn't work hard on it (V2) and instead takes a more relaxed approach, focusing on maintaining a healthy work-life balance.	What event or state is both the effect on Benjamin after Benjamin go to work every day (V1) and also what hindered Benjamin work hard on it (V2)?	Benjamin is sick ? (Not perfect as Benjamin is trying to keep a work-life balance instead of having a sick leave)	Benjamin gets tired from working hard ✓
2i	Chloe is known for being hardworking (V1) and dedicated. As a result, Chloe leads a good life (V2).	What event or state is both the effect on Chloe after Chloe is hardworking (V1) and also what Chloe wants to do after Chloe leads a good life (V2)?	to have a good life ? (No inferential gap)	to have success in life ? (No inferential gap)
ip	After looking for a new car (V1), Lydia is driving to school (V2).	What event or state is what Lydia needed to do before the event that is both what Lydia wants to do after Lydia is looking for a new car (V1), and also what Lydia needed to do before Lydia is driving to school (V2)?	None ✗	take a car for test drive ✓

Table 13: Error analysis of generated inferences on the evaluation set of COM^2. We present the generations of COMET-Llama-7b and COM^2-Llama-7b fine-tuned on all queries.