GenDec: A robust generative Question-decomposition method for Multi-hop reasoning

Anonymous ACL submission

Abstract

Multi-hop QA (MHQA) involves step-by-step reasoning to answer complex questions and find multiple relevant supporting facts. However, Existing large language models'(LLMs) reasoning ability in multi-hop question answering remains exploration, which is inadequate in answering multi-hop questions. Moreover, it is unclear whether LLMs follow a desired reasoning chain to reach the right final answer. In this paper, we propose a **gen**erative question **dec**omposition method (GenDec) from the perspective of explainable QA by generating independent and complete sub-questions based on incorporating additional extracted evidence for enhancing LLMs' reasoning ability in RAG. To demonstrate the impact, generalization, and robustness of Gendec, we conduct two experiments, the first is combining GenDec with small QA systems on paragraph retrieval and QA tasks. We secondly examine the reasoning capabilities of various state-of-the-art LLMs including GPT-4 and GPT-3.5 combined with GenDec. We experiment on the HotpotQA, 2WikihopMultiHopQA, MuSiQue, and PokeMQA datasets.

1 Introduction

In the field of natural language processing, Multi-hop Question Answering (MHQA) tasks entail iterative reasoning across diverse informational sources, such as text paragraphs. Recent advancements have demonstrated that Large Language Models (LLMs) can achieve performance comparable to that of models fine-tuned for this specific task.

Retrieval-augmented generation (RAG) represents a significant enhancement to LLMs by incorporating relevant knowledge retrieval, thereby showing considerable promise in reducing LLM-generated hallucinations and improving the overall quality of responses. This, in turn, promotes the broader application of LLMs in practical scenarios (Tang and Yang, 2024). Nonetheless, the core aspect of LLMs' reasoning capability—achieving correct answers through accurate reasoning chains—remains under investigation, particularly regarding its potential to further augment LLMs' performance.

Moreover, Tang et al. (2020) introduced a dataset of human-verified sub-questions derived from HotpotQA (Yang et al., 2018a) and conducted experiments focused on sub-question reasoning. Their findings reveal that models such as DFGN(Qiu et al., 2019), DecompRC(Min et al., 2019a), and CogQA(Ding et al., 2019), while capable of correctly answering the overarching multi-hop questions, exhibit significant deficiencies in addressing sub-questions. This underscores a prevalent issue wherein models may circumvent the necessary reasoning process, thus failing to deduce intermediate answers to sub-questions, highlighting a critical area for further research and development in MHQA systems.

Thus, understanding and potentially decomposing multi-hop questions into finer-grained sub-questions is a key desired step in QA. To accurately answer a multi-hop question, traditionally QD + QA methods start by decomposing the given multi-hop question into simpler sub-questions, attempting to answer them in a specific order, and then finally aggregating the information obtained from all sub-questions.

The elucidation and subsequent decomposition of multi-hop questions into more granular sub-questions represent a pivotal step in the domain of Question Answering (QA). Conventional methodologies, namely Question Decomposition (QD) plus QA, commence by segmenting the multi-hop question into simpler, constituent sub-questions. This process involves answering these sub-questions in a predetermined sequence, culminating in the synthesis of information derived from all sub-questions to formulate the final answer.

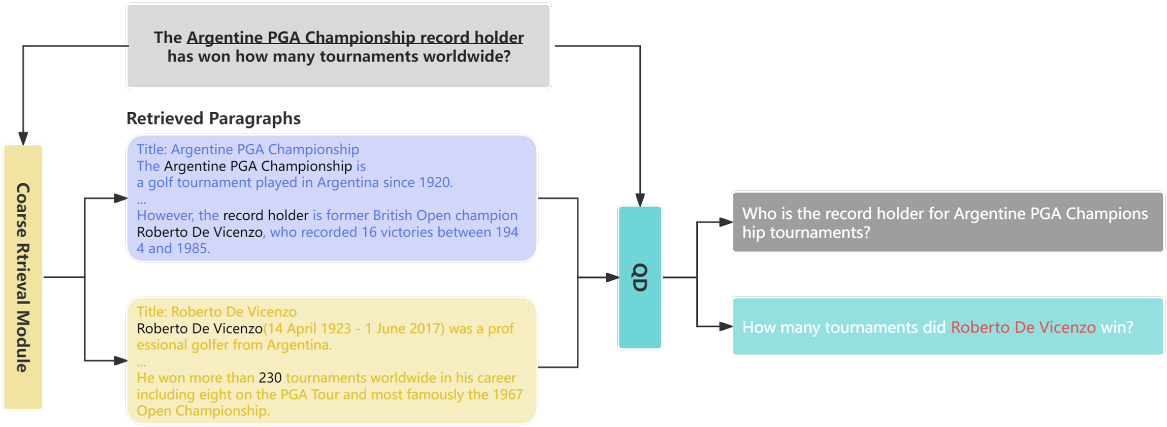

The Argentine PGA Championship record holder has won how many tournaments worldwide?
Retrieved Paragraphs
Coarse Rtrieval Module
Title: Argentine PGA Championship
The Argentine PGA Championship is a golf tournament played in Argentina since 1920.
...
However, the record holder is former British Open champion Roberto De Vicenzo, who recorded 16 victories between 194 4 and 1985.
Title: Roberto De Vicenzo
Roberto De Vicenzo(14 April 1923 - 1 June 2017) was a prof essional golfer from Argentina.
...
He won more than 230 tournaments worldwide in his career including eight on the PGA Tour and most famously the 1967 Open Championship.
QD
Who is the record holder for Argentine PGA Champions hip tournaments?
How many tournaments did Roberto De Vicenzo win?

Figure 1: Example of multi-hop and decomposed sub-questions from the HotpotQA dataset. The original question is shown in light grey and the decomposed ones are in deep gray and cyan. *"Roberto de Vincenzo"* in the retrieved paragraph is the answer to sub-question Q1 and also part of sub-question Q2. The literal *"230"* is the answer to sub-question Q2. Since the paragraphs are too long, we here only list the sentences that contain supporting facts.

Our initial exploration reveals that QD constitutes a significant impediment within Multi-hop Question Answering (MHQA). Prior approaches to QD, as documented by Min et al. (2019b); Perez et al. (2020), typically bifurcate multi-hop questions into **dependent** sub-questions. For instance, as illustrated in figure 1, a multi-hop question is divided into *"Who is the record holder for Argentine PGA Championship tournaments?"* followed by *"How many tournaments did [Answer of Sub Q1] win?"*. This necessitates that QA models accurately resolve the first sub-question and utilize its answer to address the subsequent one, aiming to ascertain the ultimate response. This QD+QA paradigm is prone to error propagation, where inaccuracies in resolving any sub-question can misguide the final answer determination. Recent work on solving complex or multihop questions is to leverage LLMs with Chain-of-Thought (CoT) (Fu et al., 2022b; Wei et al., 2022) and In-context-learning (ICT) (Liu et al., 2021) to iteratively decompose and answer complex questions in step-wise, which suffer the error propagation as well. Venktesh et al. (2023) hypothesizes that existing specialized QA datasets with rationales or decompositions might already contain instances that can be reused as demonstrations for question decompositions but fail for generalization on other complex QA scenarios.

Our proposed GenDec model alleviates these issues by ensuring that the decomposed sub-questions are independent and self-contained, eliminating the need for sequential answering inherent in previous models. GenDec integrates these sub-questions into the QA model to facilitate an appropriate reasoning pathway.

We introduce **GenDec**, a generative approach to QD that leverages retrieved paragraphs containing evidential support for segmenting multi-hop questions into independent sub-questions, which do not necessitate ordered answering. Post-QD, GenDec amalgamates the attributes of these sub-questions with mechanisms for relevant paragraph retrieval, supporting facts prediction, and the QA process. As depicted in figure 1, GenDec's decomposition of a question from the HotpotQA demonstrates this methodology. The original multi-hop inquiry *"The Argentine PGA Championship record holder has won how many tournaments worldwide?"* is segmented into independent sub-questions: *"Who is the record holder for Argentine PGA Championship tournaments?"* and *"How many tournaments did Roberto De Vicenzo win?"*.

GenDec distinguishes itself in the realm of sub-question answering within MHQA tasks by its reliance solely on retrieved paragraphs for decomposing questions, thereby obviating the need to account for the sequence or relational hierarchy of sub-questions. This feature not only simplifies the decomposition process but also enhances the robustness of the model. To underscore the continued significance of Question Decomposition (QD) in the era of Large Language Models (LLMs), we undertake a comprehensive evaluation of GenDec's performance and its generalizability across various MHQA contexts.

Our evaluation framework consists of two pivotal experiments designed to benchmark the performance of the GenDec with a fine-tuned QA model against state-of-the-art QA models. The initial experiment focuses on evaluating various SOTA models in terms of their paragraph retrieval and QA capabilities. Subsequently, the second experiment aims to examine the reasoning and answering prowess of advanced LLMs, including GPT-4 (Achiam et al., 2023), GPT-3.5 (Ouyang et al., 2022), and text-davinci-003, particularly when navigating MHQA tasks with the aid of sub-questions. The findings from these experiments collectively affirm that GenDec not only enhances QA performance but also significantly improves paragraph retrieval outcomes across both fine-tuned models and LLMs.

This comprehensive examination and the resultant insights not only highlight GenDec's superior performance and versatility but also reiterate the indispensable role of question decomposition in refining the reasoning abilities of LLMs. The contributions of our work are manifold and can be summarized as follows: 1) We introduce GenDec, an innovative and robust approach that adeptly generates natural language sub-questions leveraging retrieved paragraphs, thereby concealing the underlying reasoning chains. This methodology facilitates a more intuitive and efficient process for question decomposition. 2) Through rigorous experimentation, we demonstrate that GenDec's integration of generated sub-questions into paragraph retrieval and QA modules not only surpasses the performance of existing QD-based QA models but also establishes new benchmarks when compared to other formidable baselines. 3) Our analysis extends to the synergistic combination of GenDec with LLMs, revealing the pivotal role of QD in augmenting the reasoning capabilities of LLMs.

2 Related Work

2.1 Multi-hop Question-answering

Multi-hop QA requires more than one reasoning step in multiple paragraphs to answer a question. For example, multi-hop QA in DROP (Dua et al., 2019) requires numerical reasoning such as addition and subtraction. Yang et al. (2018b) proposed the HotpotQA dataset that contains 113K multi-hop QA pairs collected from Wikipedia articles by crowd-sourcing. Ho et al. (2020a) presented 2WikiMultiHopQA, which uses structured and unstructured data and introduces the evidence information containing a reasoning path for multi-hop questions.

2.2 Question Decomposition

Several studies conducted QD in complex QA tasks by using different methods. Wolfson et al. (2020a) and Talmor and Berant (2018), inspired by SQL and SPARQL query, proposed rule-based methods. However, they failed to generalize into different types of questions because of the limited rules. Min et al. (2019b) proposed a supervised QD method with human-labeling data to predict the text span of sub-questions. ONUS (Perez et al., 2020) is a one-to-N unsupervised sequence transduction method that uses supervision information of pseudo-decompositions from Common Crawl to map complex questions into simpler questions and recompose intermediate answers of sub-questions for reasoning final answers. These supervised and unsupervised QD methods decompose complex questions into two sub-questions but are not applicable to real scenarios. Deng et al. (2022b) trains an Abstract Meaning Representation (AMR)-to-text generation model on the QDMR (Wolfson et al., 2020b) dataset. The entity description graph (EDG)-based QD method (Hu et al., 2021) represents the structure of complex questions to solve the question-understanding and component-linking problems of knowledge base QA tasks. Zhou et al. (2022) pre-trained Decomp-T5 on human-collected parallel news to improve the ability of semantic understanding for QD. Instead of answering sub-questions one by one, Guo et al. (2022) directly concatenated sub-questions with the original question and context to leverage the reading-comprehension model to predict the answer. Wang et al. (2022) propose a step-by-step sub-question generation that generates sub-questions at each intermediate step. However, such step-wise reasoning and generation methods suffer from error propagation, while ours can directly generate the sub-questions and reasoning at the same time.

2.3 LLM reasoning

LLMs have shown reasoning abilities over several tasks, such as multi-hop QA (Bang et al., 2023), commonsense reasoning (Liu et al., 2022), and table QA (Chen, 2022). Chain-of-thought (CoT) (Wei et al., 2022) leverages a series of intermediate reasoning steps, achieving better reasoning performance on complex tasks. Jin and Lu (2023) proposed a framework called Tabular Chain of

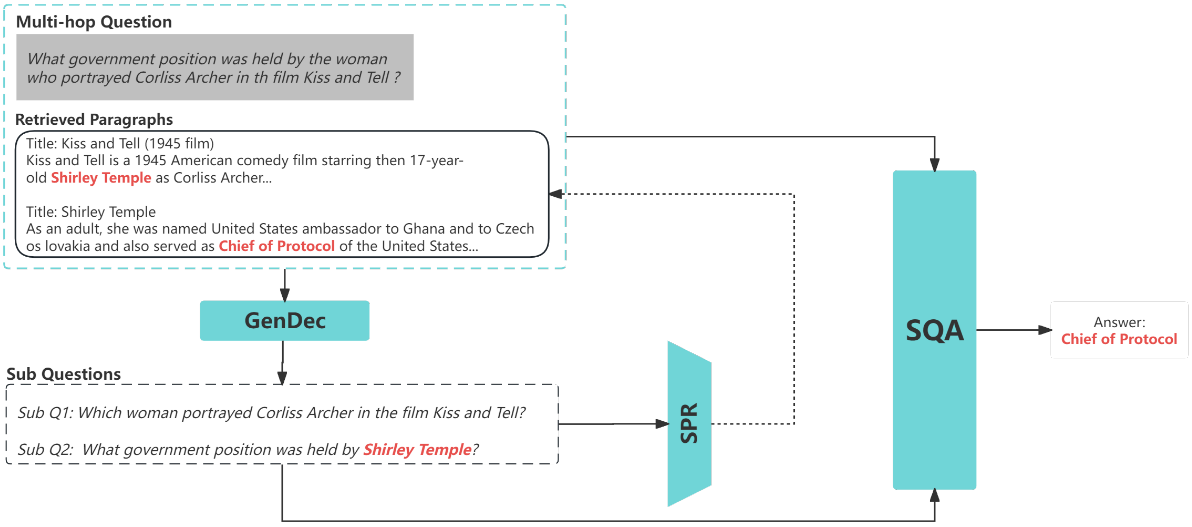

Multi-hop Question
What government position was held by the woman who portrayed Corliss Archer in th film Kiss and Tell ?
Retrieved Paragraphs
Title: Kiss and Tell (1945 film)
Kiss and Tell is a 1945 American comedy film starring then 17-year-old Shirley Temple as Corliss Archer...
Title: Shirley Temple
As an adult, she was named United States ambassador to Ghana and to Czechos lovakia and also served as Chief of Protocol of the United States...
GenDec
Sub Questions
Sub Q1: Which woman portrayed Corliss Archer in the film Kiss and Tell?
Sub Q2: What government position was held by Shirley Temple?
SPR
SQA
Answer:
Chief of Protocol

Figure 2: Pipeline of GenDec. From top to bottom. We first carry out Question Decomposition (QD) to decompose a multi-hop question into its sub-questions and then train a **S**ub-question-enhanced **P**aragraph **R**etrieval module (SPR). We then input multi-hop questions, sub-questions, as well as retrieved paragraphs, into the sub-question-enhanced QA module to extract the final answers.

Thought (Tab-CoT) that can perform step-by-step reasoning on complex tableQA tasks by creating a table without fine-tuning by combining the table header with related column names as a prompt. Khot et al. (2022) proposed an approach called Decomposed Prompting to solve complex tasks by decomposing them into simple sub-tasks that can be delegated to a shared library of prompting-based LLMs dedicated to these sub-tasks.

However, these studies only decomposed questions into sub-questions and the latter sub-questions always rely on previous sub-questions. When the previous sub-questions are incorrectly answered, the latter sub-questions are also prone to be misguided.

3 Framework

As discussed in the preceding section, previous QD-based QA methods fail to mitigate the error-propagation problem during the answer reasoning process as they decompose questions into sub-questions. Our framework consists of three main components: (1) a generative QD module, GenDec, to generate independent sub-questions (2) a sub-question-enhanced paragraph-retrieval module, that serves both the supporting facts prediction and QA tasks; and (3) a sub-question enhanced QA module, which fuses features of sub-questions for QA and supporting-facts prediction. Figure 2 shows the overall framework.

3.1 GenDec

We explore different model architectures for the GenDec, i.e., generative language models (e.g., BART (Lewis et al., 2019), T5(Raffel et al., 2020)) as the QD module.

We first leverage the coarse retrieval module proposed by Yin et al. (2023) to retrieve relevant paragraphs to serve the GenDec module. In the coarse retrieval module, each question Q is typically combined by a set of N paragraphs $P_1, P_2 \dots, P_N$, but only a small number of paragraphs (e.g., two in HotpotQA) are labeled as relevant to the question Q. We model paragraph retrieval as a binary classification task. Specifically, for each question-paragraph pair, we concatenate it as "[CLS], question, [SEP], paragraph, [SEP]" in sequence.

Generative Question Decomposition To ensure the sub-questions are answerable by the QA module, we train a text-to-text generation model on the sub-question dataset from HotpotQA Khot et al. (2021). Moreover, to improve the generalization of the GenDec module for different types of multihop questions, we also utilize the PokeMQA dataset(Gu et al., 2023) as part of the training data for the QD module. PokeMQA is an MHQA dataset with corresponding sub-questions (including 2, 3, 4-hop questions).

We use BART-large and T5-large models as backend models and fine-tune them on the sub-question datasets to generate sub-questions. We use the retrieved paragraphs that contain supporting facts p and question q as input to train a question-generator model $G : (p, q) \Rightarrow sub\_qs$, where sub_qs is the generated sub-question set. Such a generator, G, produces the two sub-questions in the example in Figure 1. The details of finetuning T5-large and BART-large are given in Appendix A.

3.2 Sub-question-enhanced Paragraph Retrieval (SPR)

Multi-hop question answering takes textual context into account and usually, MHQA datasets include multiple paragraphs as question context (e.g., HotpotQA and 2WikiMultiHopQA datasets include 10 paragraphs per question). However, including all such paragraphs is not ideal due to noise and size (length). Therefore, paragraph retrieval plays a vital role in both QA and QD modules, since GenDec utilizes information from sub-questions and can thus focus on the more relevant data.

We propose sub-question-enhanced paragraph retrieval (**SPR**) for refined retrieval, which utilizes an encoder and a classification head to compute scores for each paragraph to help the supporting facts prediction. Given a k-hop question Q, generated k sub-questions $q_1, ...q_k$, and a candidate set with n passages as $\mathcal{P} = \{p_1, p_2, ..., p_n\}$, SPR aims to retrieve a relevant paragraph set $(\hat{p}_1, \hat{p}_2, ..., \hat{p}_k)$ that relates to the k sub-questions and the k-hop question Q. While most previous work formulates it as a one- or two-step sequence labeling task, classifying every passage $p_i \in \mathcal{P}$ as relevant or not.

A passage $p_i \in \mathcal{P}$ corresponds to the question Q and j-th sub-question $q_j \in \mathcal{S}$. Consequently, we also denote the output score of SPR as $S(\hat{p}_i|Q, q_j)$, given the concatenated sequence of question, sub-question, and passages identified so far, $(Q, q_j, \hat{p}_i)$.

We use the DeBERTa model (He et al., 2021) as an encoder to derive embeddings for the concatenated sequence $(Q, q_j, \hat{p}_i)$ and the output $\acute{o}_i \in \mathbb{R}^n$. Subsequently, a fully connected layer is added after DeBERTa to project the final dimension of the "[CLS]" representations of these embeddings into a 2-dimensional space, representing "irrelevant" and "relevant" respectively. The logit in the "relevant" side serves as the score for each paragraph. This scoring process is denoted by a function $S(\hat{p}_i|Q, q_j)$. In SPR, we optimize the classification of each combination of question, sub-question, and paragraph using Cross-Entropy loss.

$$\mathcal{L}_j = -\sum_{q_i \in \mathcal{S}} \sum_{\hat{p}_i \in \mathcal{P}} l_{j,p} log S(\hat{p}_i|Q, q_j) + (1 - l_{j,p}) log(1 - S(\hat{p}_i|Q, q_j)) \quad (1)$$

where $l_{j,p}$ is the label of $\hat{p}_i$ and $S(\hat{p}_i|Q, q_j)$ is the score function predicted by the model.

Thus, we train a paragraph retrieval model based on DeBERTa (He et al., 2021) to execute binary classification and rank the scores of paragraphs containing the gold supporting facts.

3.3 Sub-question-enhanced QA (SQA)

In the QA task, we propose a sub-question-enhanced QA model which utilizes multi-task learning to simultaneously predict supporting facts, and extract answer spans by incorporating sub-questions produced from GenDec. In order to better evaluate the role of sub-question incorporation, we do not include other additional modules in our model. Instead, we focus on the effects of sub-question incorporation on the performance of SQA. Additionally, as both HotpotQA and 2WikiMultiHopQA datasets also contain questions with yes/no answers, a common scenario, we include an answer type task.

The SQA first combines all retrieved paragraphs into context C, which is concatenated with question Q and sub-questions $\{Sub\_Qs\}$ and fed into DeBERTa. We denote the encoded question and sub-question representations as $\mathbf{Q} = \{\mathbf{q}_0, \mathbf{q}_1, \ldots, \mathbf{q}_{Q-1}\} \in \mathbf{R}^{m \times d}$ and the encoded context representation as $\mathbf{C} = \{\mathbf{c}_0, \mathbf{c}_1, ..., \mathbf{c}_{C-1}\} \in \mathbf{R}^{C \times d}$, where Q is the length of the question. Each $\mathbf{q}_i$ and $\mathbf{c}_j \in \mathbf{R}^d$.

$$\begin{aligned} \mathbf{P}^i &= \text{DeBERTa}\left(P^{(i)}\right) \\ \mathbf{sub\_q}^i &= \text{DeBERTa}\left(Sub\_Q^{(i)}\right) \\ \mathbf{q} &= \text{DeBERTa}(\mathbf{Q}), \end{aligned} \quad (2)$$

where $P^{(i)} \in \mathbf{R}^d$, $Sub\_Q^{(i)} \in \mathbf{R}^d$, $\mathbf{Q} \in \mathbf{R}^d$ respectively denote the i-th paragraph, sub-question, and question representations.

To extract answer spans, we use a linear prediction layer on the contextual representation to identify the start and end positions of answers and employ cross-entropy as the loss function. The corresponding loss terms are denoted as $\mathcal{L}_{start}$ and $\mathcal{L}_{end}$, respectively. The classification loss for the supporting facts is denoted as $\mathcal{L}_{sup}$, and we jointly optimize all of these objectives in our model. We also introduce an answer type classification module trained with crossentropy loss function.

$$\mathcal{L}_{type} = \mathbb{E}[-\sum_{i=1}^{3} y_i^{type} log(\hat{y}_i^{type})] \quad (3)$$

where $\hat{y}i^{fine}$ denotes the predicted probability of answer types classified by our model, and yi^{fine} represents the corresponding one-hot encoded ground-truth distribution. y_i^{type} has three values: 0 denotes a negative answer, 1 denotes a positive answer, and 2 denotes the answer is a span.

The multi-task prediction model's total loss is:

$$\mathcal{L}_{reading} = \lambda_1 \mathcal{L}_{type} + \lambda_2(\mathcal{L}_{start} + \mathcal{L}_{end}) + \lambda_3 \mathcal{L}_{sup} \quad (4)$$

Similarly, we set λ_1, λ_2, and λ_3 all to 1, giving equal importance to each module for multitask learning. The implementation details of the Sub-question-enhanced QA module are described in Appendix A.

4 Experiments and Analysis

This section describes the utilized datasets and we first illustrate the performance of SQA in the QA task in section 4.3; then we show the impact of GenDec with LLMs in section 4.4; To describe the GenDec effectiveness on paragraph retrieval, we conduct and analyze experiment results in section 4.5; To further show the effectiveness of SQA on reasoning chain, we list results in section 4.6; We analyze the ablation study of GenDec on SQA in section 4.7. We also illustrate the quality of generated sub-questions to describe the effectiveness, generalization and robustness of GenDec in appendix B.

4.1 Datasets

Question Answering (QA) We evaluate GenDec on the MuSiQue (Trivedi et al., 2022), 2WikiMultiHopQA (Ho et al., 2020b) and HotpotQA (Yang et al., 2018a) datasets, which contain 20K, 160K and 90K training instances. These three multi-hop QA datasets consist of questions, answers, supporting facts, and a collection of 10 paragraphs as context per question.

Question Decomposition (QD) To train and evaluate GenDec, we use the sub-questions and answers data processed from the multi-hop HotpotQA dataset Khot et al. (2021) - here named SQA for clarity. These sub-questions are relatively high quality, in that we are able to use them to train a sub-question generator that achieves high task performance on multi-hop QD. However, most of the questions in the HotpotQA dataset are 2-hop questions, to improve the generalization of GenDec, we here add the PokeMQA (Gu et al., 2023) (including 2, 3, 4-hop questions) as the data augmentation. PokeMQA contains 3000 multi-hop questions (1000 2-hop questions, 1000 3-hop questions, and 1000 4-hop questions) with corresponding sub-questions. We divided the PokeMQA into the train and test sets, and the split ratio is 80% for the train set and 20% for the test set. It is important to note that once trained, the GenDec module can be re-used across scenarios, without depending on dataset-specific decomposed questions of golden annotations.

Sub-question Reasoning To evaluate the reasoning ability of GenDec with LLMs and finetuned QA models, we also utilize a human-verified sub-question test dataset derived from HotpotQA Tang et al. (2020) - here named HVSQA for clarity; which provides a strong benchmark to evaluate QA models in answering complex questions via sub-question reasoning.

4.2 Experiment Results

4.3 SQA against SOTA QA systems

We use Exact Match (EM) and F1 scores as evaluation metrics for answer span prediction and supporting facts prediction on the HotpotQA, 2WikiMultiHopQA, and MuSiQue-Ans datasets to compare the performance of SQA with that of QD-based, GNN-based, and other SOTA QA models.

As shown in Table 1, SQA With the Deberta-large model outperforms most models in both metrics, including the strong baseline, and performs very competitively to the latest SOTA on the HotpotQA dataset, only below the Beam retrieval (Zhang et al., 2023).

The middle and bottom sections of the table also show that SQA significantly outperforms most previous work on the 2WikiMultiHopQA and MuSiQue-Ans (Trivedi et al., 2022) datasets. GenDec's performance is only lower than the contemporary Beam Retrieval Zhang et al. (2023), which takes a retrieval approach that can be complementary to SQA itself.

Model	Ans		Sup		Joint	
	EM	F1	EM	F1	EM	F1
HotpotQA						
QD-based QA Models						
DecompRC (Min et al., 2019b)	55.20	69.63	-	-	-	-
ONUS (Perez et al., 2020)	66.33	79.34	-	-	-	-
GNN-based Models						
DFGN (Xiao et al., 2019)	56.31	69.69	51.50	81.62	33.62	59.82
SAE-large (Tu et al., 2020)	66.92	79.62	61.53	86.86	45.36	71.45
C2F Reader(Shao et al., 2020)	67.98	81.24	60.81	87.63	44.67	72.73
HGN-large (Fang et al., 2019)	69.22	82.19	62.76	88.47	47.11	74.21
BRF-graph (Huang and Yang, 2021)	70.06	82.20	61.33	88.41	45.92	74.13
AMGN+ (Li et al., 2021)	70.53	83.37	63.57	88.83	47.77	75.24
Other STATE-OF-The-ART(SOTA) Models						
FE2H on ALBERT (Li et al., 2022b)	71.89	84.44	64.98	89.14	50.04	76.54
PCL (Deng et al., 2022a)	71.76	84.39	64.61	89.20	49.27	76.56
Smoothing R3 (Yin et al., 2023)	72.07	84.34	65.44	89.55	49.73	76.69
Beam Retrieval (Zhang et al., 2023)	**72.69**	**85.04**	**66.25**	90.09	**50.53**	**77.54**
SQA (ours)	72.39	84.69	65.88	**90.31**	50.34	77.48
2WikiMultihopQA						
CRERC (Fu et al., 2021)	69.58	72.33	82.86	90.68	49.80	58.99
NA-Reviewer (Fu et al., 2022a)	76.73	81.91	89.61	94.31	52.75	65.23
BigBird-base model (Ho et al., 2023)	74.05	79.68	77.14	92.13	39.30	63.24
Beam Retrieval (Zhang et al., 2023)	**88.47**	**90.87**	**95.87**	**98.15**	-	-
SQA (ours)	86.47	88.15	93.28	96.45	56.87	68.38
MuSiQue-Ans						
Beam Retrieval (beam size 2) (Zhang et al., 2023)	-	**69.20**	-	**91.40**	-	-
Beam Retrieval (beam size 1) (Zhang et al., 2023)	-	66.90	-	90.00	-	-
Ex(SA) (Trivedi et al., 2022)	-	49.00	-	78.10	-	-
Ex(EE) (Trivedi et al., 2022)	-	46.40	-	80.60	-	-
SQA (ours)	-	65.40	-	87.90	-	-

Table 1: Performance of different QA models on test distractor settings of HotpotQA, 2WikiMultihopQA and MuSiQue Answerable datasets. GenDec outperforms all QD-based and other GNN-based QA models.

Model	**EM**	**F1**
SAE$_{large}$ (Tu et al., 2020)	91.98	95.76
S2G$_{large}$ (Wu et al., 2021)	95.77	97.82
FE2H$_{large}$ (Li et al., 2022a)	96.32	98.02
C2FM$_{large}$ (Yin et al., 2023)	96.85	98.32
Beam Retrieval$_{large}$ (Zhang et al., 2023)	97.52	98.68
SPR (ours)	97.53	98.78
GenDec + Beam Retrieval	**98.02**	**99.17**

Table 2: Comparison of our SPR with previous baselines on HotpotQA dev set.

q	q_{sub1}	q_{sub2}	DFGN	DecRC	HGN	PCL	BR	SQA
c	c	c	23	31.3	39.5	43.6	52.2	52.9
c	c	w	9.7	7.2	5.1	6.8	7.8	6.4
c	w	c	17.9	19.1	19.6	21.3	21.7	18.8
c	w	w	7.5	5.5	3.8	2.1	6.7	6.2
w	c	c	4.9	3.0	2.8	1.7	1.2	1.1
w	c	w	17	18.6	16.7	16.3	7.3	10.3
w	w	c	3.5	3.4	2.6	1.1	0.7	0.9
w	w	w	16.5	11.9	9.9	7.1	2.4	3.4

Table 3: Categorical EM statistics (%) of sub-question evaluation for six multi-hop QA models over HVSQA (Tang et al., 2020). c/w denotes questions answered correctly/wrongly. For example, the fourth row shows the percentage of multi-hop questions that can be correctly answered while sub-questions cannot.

4.4 GenDec with LLMs

We also evaluate the impact of generated independent sub-questions on LLM reasoning in table 5. We also evaluated the performance of LLMs with and without GenDec on 1000 samples of dev distractor settings. Figure 3 shows the used with QD and without QD prompt settings. We selected the 1-shot setting in which LLMs are given one example from the training set with two prompts, one is reasoning over sub-questions and the other is directly reasoning answers. As shown in Table 5, GPT-4 with additional sub-question information performs better than without sub-questions. GPT-4 with QD prompting achieves higher answer span extraction on the F1 score (82.46) and EM (64.34) respectively.

Moreover, we also evaluate the reasoning chain of LLMs with and without GenDec, we here follow the experiment settings of Tang et al. (2020) and calculate the proportion of reasoning chains which is shown in table 6 in the appendix. For instance, GPT-4 gets 39.7% right reasoning chain (green row) which achieves a correct right final answer based on right sub-answer1 and sub-answer2. However, the wrong reasoning chain is 21.1% (sum of the three red rows) although a right final answer either sub-answer1 or sub-answer2 is wrong. When removing the GenDec module, GPT-4 only gets a 30.5% right reasoning chain, and 24.5% wrong reasoning chain, deeming that GenDec generated parallel sub-questions could help LLMs achieve a higher performance following a higher proportion of reasoning chain.

4.5 SPR on paragraph retrieval

Table 2 shows the SOTA paragraph retrieval performance of GenDec's SPR method against previous strong paragraph retrieval model baselines. SPR reaches very competitive results against Beam Retrieval (slightly higher in F1 vs slightly lower EM). Moreover, combining our GenDec approach with Beam Retrieval further improves performance and showcases the efficacy of leveraging sub-questions. GenDec + Beam Retrieval achieves SOTA performance on paragraph retrieval, showing the high quality of retrieved paragraphs.

4.6 SQA on reasoning chain evaluation

To analyze whether existing multi-hop QA models can demonstrate the right reasoning process, we compare the percentage of correct final answers and intermediate answers obtained by DFGN (Xiao et al., 2019), DecompRC (DecRC) (Min et al., 2019b), HGN (Fang et al., 2019), PCL (Deng et al., 2022a), Beam Retrieval (BR) (Zhang et al., 2023), and our SQA. Table 3 summarizes the percentage of correct answers for intermediary sub-questions and final multi-hop questions on the HVSQA (Tang et al., 2020) dataset. We observe that, although Beam Retrieval (BR) achieves a relatively high percentage of correct final answers (88.3% vs SQA's 84.3%), 36.2% of these are obtained from incorrect intermediate answers and reasoning chains. While SQA's performance is less affected by this case, at 31.4%. Comparing fully correct answer chains, Beam Retrieval and SQA reach 52.2%, and 52.9% respectively. The experiment results reveal that previous SOTA QA systems are not reliable enough and get an inflated performance as they usually bypass the right reasoning chain.

Model	Q_ori		Q_sub1		Q_sub2	
	F1	EM	F1	EM	F1	EM
CogQA	67.82	53.2	69.65	58.6	68.49	54
DFGN	71.96	58.1	68.54	54.6	60.83	49.3
DecompRC	77.61	63.1	75.21	61	70.77	56.8
ONUS	79.25	67.43	77.56	63.89	72.21	57.62
PCL	73.8	87.15	68.4	83.62	68.5	81.07
SQA w/o GenDec	82.81	70.72	87.45	72.65	80.12	70.38
SQA with GenDec	**86.17**	**72.88**	**90.52**	**76.43**	**84.61**	**74.83**

Table 4: Performance comparison between SQA (with and without the GenDec module) and other QA models on HVSQA (Tang et al., 2020), a human-verified sub-question test dataset from HotpotQA.

4.7 Ablation Study

To evaluate the impact of the GenDec module on the SQA model, we conduct an ablation study testing the performance of answering all sub-questions and original questions, with and without the GenDec module. The results, shown in Table 4, indicate that the GenDec module shows consistent and significantly improved results; improving the F1 score and EM by 3.36 and 2.16, respectively, in the original QA. In answering intermediate answers to sub-questions, SQA w/ GenDec also improves over w/o GenDec (improving the F1 score and EM by 3.07 and 3.78, and 4.49 and 4.45 on sub-questions 1 and 2 respectively). The results indicate that the GenDec module plays an important role in QA tasks not only in its QA ability but also in high performance of intermediate answer reasoning to support answering the final question.

5 Conclusion

We proposed GenDec, a generative-based QD method that generates complete and independent sub-questions based on incorporating retrieved paragraphs. We train an SPR and an SQA module that incorporates sub-questions generated by GenDec and show that it significantly improves QA and paragraph retrieval tasks. We also explore the impact of GenDec on LLMs. Lastly, while GenDec outperforms all previous QD-based and GNN-based QA systems in multi-hop QA, it can still face errors due to incorrect supporting fact predictions influencing the model to incorrectly predict both sub-questions and final answers.

6 Limitations

In this paper, we focus on the impact of GenDec in multi-hop QA, where the answers to most questions can be decomposed into several independent subquestions via the fusion of retrieved paragraphs. Although GenDec performs very well on QD and help improve QA, one of its limitations is that it is still sensitive to errors in paragraph retrieving. The QD results would be affected when incorrect paragraphs are selected. For future work, we plan to focus on tackling this problem.

References

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

Yejin Bang, Samuel Cahyawijaya, Nayeon Lee, Wenliang Dai, Dan Su, Bryan Wilie, Holy Lovenia, Ziwei Ji, Tiezheng Yu, Willy Chung, et al. 2023. A multitask, multilingual, multimodal evaluation of chatgpt on reasoning, hallucination, and interactivity. *arXiv preprint arXiv:2302.04023*.

Wenhu Chen. 2022. Large language models are few(1)-shot table reasoners. In *Findings*.

Zhenyun Deng, Yonghua Zhu, Yang Chen, Qianqian Qi, Michael Witbrock, and Patricia Riddle. 2022a. Prompt-based conservation learning for multi-hop question answering. *arXiv preprint arXiv:2209.06923*.

Zhenyun Deng, Yonghua Zhu, Yang Chen, M. Witbrock, and Patricia J. Riddle. 2022b. Interpretable amr-based question decomposition for multi-hop question answering. In *International Joint Conference on Artificial Intelligence*.

Ming Ding, Chang Zhou, Qibin Chen, Hongxia Yang, and Jie Tang. 2019. Cognitive graph for multi-hop reading comprehension at scale. *arXiv preprint arXiv:1905.05460*.

Dheeru Dua, Yizhong Wang, Pradeep Dasigi, Gabriel Stanovsky, Sameer Singh, and Matt Gardner. 2019. Drop: A reading comprehension benchmark requiring discrete reasoning over paragraphs. In *North American Chapter of the Association for Computational Linguistics*.

Yuwei Fang, S. Sun, Zhe Gan, Rohit Radhakrishna Pillai, Shuohang Wang, and Jingjing Liu. 2019. Hierarchical graph network for multi-hop question answering. In *Conference on Empirical Methods in Natural Language Processing*.

Ruiliu Fu, Han Wang, Xuejun Zhang, Jun Zhou, and Yonghong Yan. 2021. Decomposing complex questions makes multi-hop QA easier and more interpretable. In *Findings of the Association for Computational Linguistics: EMNLP 2021*, pages 169–180, Punta Cana, Dominican Republic. Association for Computational Linguistics.

Ruiliu Fu, Han Wang, Jun Zhou, and Xuejun Zhang. 2022a. Na-reviewer: Reviewing the context to improve the error accumulation issue for multi-hop qa. *Electronics Letters*, 58(6):237–239.

Yao Fu, Hao-Chun Peng, Ashish Sabharwal, Peter Clark, and Tushar Khot. 2022b. Complexity-based prompting for multi-step reasoning. *ArXiv*, abs/2210.00720.

Hengrui Gu, Kaixiong Zhou, Xiaotian Han, Ninghao Liu, Ruobing Wang, and Xin Wang. 2023. Pokemqa: Programmable knowledge editing for multi-hop question answering. *ArXiv*, abs/2312.15194.

Xiao-Yu Guo, Yuan-Fang Li, and Gholamreza Haffari. 2022. Complex reading comprehension through question decomposition. *ArXiv*, abs/2211.03277.

Pengcheng He, Xiaodong Liu, Jianfeng Gao, and Weizhu Chen. 2021. DeBERTa: Decoding-enhanced BERT with disentangled attention. In *ICLR 2021*.

Xanh Ho, Anh-Khoa Duong Nguyen, Saku Sugawara, and Akiko Aizawa. 2020a. Constructing a multi-hop QA dataset for comprehensive evaluation of reasoning steps. In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 6609–6625, Barcelona, Spain (Online). International Committee on Computational Linguistics.

Xanh Ho, Anh-Khoa Duong Nguyen, Saku Sugawara, and Akiko Aizawa. 2020b. Constructing a multi-hop QA dataset for comprehensive evaluation of reasoning steps. In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 6609–6625, Barcelona, Spain (Online). International Committee on Computational Linguistics.

Xanh Ho, Anh-Khoa Duong Nguyen, Saku Sugawara, and Akiko Aizawa. 2023. Analyzing the effectiveness of the underlying reasoning tasks in multi-hop question answering. In *Findings of the Association for Computational Linguistics: EACL 2023*, pages 1163–1180, Dubrovnik, Croatia. Association for Computational Linguistics.

Xixin Hu, Yiheng Shu, Xiang Huang, and Yuzhong Qu. 2021. Edg-based question decomposition for complex question answering over knowledge bases. In *International Workshop on the Semantic Web*.

Yongjie Huang and Meng Yang. 2021. Breadth first reasoning graph for multi-hop question answering. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 5810–5821.

Ziqi Jin and Wei Lu. 2023. Tab-cot: Zero-shot tabular chain of thought. *arXiv preprint arXiv:2305.17812*.

Tushar Khot, Daniel Khashabi, Kyle Richardson, Peter Clark, and Ashish Sabharwal. 2021. Text modular networks: Learning to decompose tasks in the language of existing models. *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, page 1264–1279.

Tushar Khot, H. Trivedi, Matthew Finlayson, Yao Fu, Kyle Richardson, Peter Clark, and Ashish Sabharwal. 2022. Decomposed prompting: A modular approach for solving complex tasks. *ArXiv*, abs/2210.02406.

Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdel rahman Mohamed, Omer Levy, Veselin Stoyanov, and Luke Zettlemoyer. 2019. Bart: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. In *Annual Meeting of the Association for Computational Linguistics*.

Ronghan Li, Lifang Wang, Shengli Wang, and Zejun Jiang. 2021. Asynchronous multi-grained graph network for interpretable multi-hop reading comprehension. In *Proceedings of the Thirtieth International Joint Conference on Artificial Intelligence, IJCAI-21*, pages 3857–3863. International Joint Conferences on Artificial Intelligence Organization. Main Track.

Xin-Yi Li, Wei-Jun Lei, and Yu-Bin Yang. 2022a. From easy to hard: Two-stage selector and reader for multi-hop question answering. *ArXiv preprint*, abs/2205.11729.

Xin-Yi Li, Weixian Lei, and Yubin Yang. 2022b. From easy to hard: Two-stage selector and reader for multi-hop question answering. *ArXiv*, abs/2205.11729.

Jiachang Liu, Dinghan Shen, Yizhe Zhang, Bill Dolan, Lawrence Carin, and Weizhu Chen. 2021. What makes good in-context examples for gpt-3? In *Workshop on Knowledge Extraction and Integration for Deep Learning Architectures; Deep Learning Inside Out*.

Jiacheng Liu, Skyler Hallinan, Ximing Lu, Pengfei He, Sean Welleck, Hannaneh Hajishirzi, and Yejin Choi. 2022. Rainier: Reinforced knowledge introspector for commonsense question answering. *ArXiv*, abs/2210.03078.

Sewon Min, Victor Zhong, Luke Zettlemoyer, and Hannaneh Hajishirzi. 2019a. Multi-hop reading comprehension through question decomposition and rescoring. In *ACL*.

Sewon Min, Victor Zhong, Luke Zettlemoyer, and Hannaneh Hajishirzi. 2019b. Multi-hop reading comprehension through question decomposition and rescoring. *ArXiv*, abs/1906.02916.

Long Ouyang, Jeff Wu, Xu Jiang, Diogo Almeida, Carroll L. Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, John Schulman, Jacob Hilton, Fraser Kelton, Luke E. Miller, Maddie Simens, Amanda Askell, Peter Welinder, Paul Francis Christiano, Jan Leike, and Ryan J. Lowe. 2022. Training language models to follow instructions with human feedback. *ArXiv*, abs/2203.02155.

Ethan Perez, Patrick Lewis, Wen tau Yih, Kyunghyun Cho, and Douwe Kiela. 2020. Unsupervised question decomposition for question answering. In *Conference on Empirical Methods in Natural Language Processing*.

Lin Qiu, Yunxuan Xiao, Yanru Qu, Hao Zhou, Lei Li, Weinan Zhang, and Yong Yu. 2019. Dynamically fused graph network for multi-hop reasoning. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 6140–6150, Florence, Italy. Association for Computational Linguistics.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. 2020. Exploring the limits of transfer learning with a unified text-to-text transformer. *Journal of Machine Learning Research*, 21(140):1–67.

Nan Shao, Yiming Cui, Ting Liu, Shijin Wang, and Guoping Hu. 2020. Is graph structure necessary for multi-hop reasoning? *ArXiv*, abs/2004.03096.

A. Talmor and J. Berant. 2018. The web as a knowledge-base for answering complex questions. In *North American Association for Computational Linguistics (NAACL)*.

Yixuan Tang, Hwee Tou Ng, and Anthony KH Tung. 2020. Do multi-hop question answering systems know how to answer the single-hop sub-questions? *arXiv preprint arXiv:2002.09919*.

Yixuan Tang and Yi Yang. 2024. Multihop-rag: Benchmarking retrieval-augmented generation for multi-hop queries. *ArXiv*, abs/2401.15391.

Harsh Trivedi, Niranjan Balasubramanian, Tushar Khot, and Ashish Sabharwal. 2022. MuSiQue: Multi-hop questions via single-hop question composition. *Transactions of the Association for Computational Linguistics*.

Ming Tu, Kevin Huang, Guangtao Wang, Jing Huang, Xiaodong He, and Bowen Zhou. 2020. Select, answer and explain: Interpretable multi-hop reading comprehension over multiple documents. In *Proceedings of the AAAI conference on artificial intelligence*, volume 34, pages 9073–9080.

V Venktesh, Sourangshu Bhattacharya, and Avishek Anand. 2023. In-context ability transfer for question decomposition in complex qa. *ArXiv*, abs/2310.18371.

Siyuan Wang, Zhongyu Wei, Zhihao Fan, Qi Zhang, and Xuanjing Huang. 2022. Locate then ask: Interpretable stepwise reasoning for multi-hop question answering. *arXiv preprint arXiv:2208.10297*.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Ed Huai hsin Chi, F. Xia, Quoc Le, and Denny Zhou. 2022. Chain of thought prompting elicits reasoning in large language models. *ArXiv*, abs/2201.11903.

Tomer Wolfson, Mor Geva, Ankit Gupta, Matt Gardner, Yoav Goldberg, Daniel Deutch, and Jonathan Berant. 2020a. Break it down: A question understanding benchmark. *Transactions of the Association for Computational Linguistics*.

Tomer Wolfson, Mor Geva, Ankit Gupta, Matt Gardner, Yoav Goldberg, Daniel Deutch, and Jonathan Berant. 2020b. Break it down: A question understanding benchmark. *Transactions of the Association for Computational Linguistics*, 8:183–198.

Bohong Wu, Zhuosheng Zhang, and Hai Zhao. 2021. Graph-free multi-hop reading comprehension: A select-to-guide strategy. *ArXiv preprint*, abs/2107.11823.

Yunxuan Xiao, Yanru Qu, Lin Qiu, Hao Zhou, Lei Li, Weinan Zhang, and Yong Yu. 2019. Dynamically fused graph network for multi-hop reasoning. *ArXiv*, abs/1905.06933.

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

Zhilin Yang, Peng Qi, Saizheng Zhang, Yoshua Bengio, William W. Cohen, Ruslan Salakhutdinov, and Christopher D. Manning. 2018b. HotpotQA: A dataset for diverse, explainable multi-hop question answering. In *Conference on Empirical Methods in Natural Language Processing (EMNLP)*.

Zhangyue Yin, Yuxin Wang, Xiannian Hu, Yiguang Wu, Hang Yan, Xinyu Zhang, Zhao Cao, Xuanjing Huang, and Xipeng Qiu. 2023. Rethinking label smoothing on multi-hop question answering. In *China National Conference on Chinese Computational Linguistics*, pages 72–87. Springer.

Jiahao Zhang, Haiyang Zhang, Dongmei Zhang, Yong Liu, and Shen Huang. 2023. Beam retrieval: General end-to-end retrieval for multi-hop question answering. *arXiv preprint arXiv:2308.08973*.

Ben Zhou, Kyle Richardson, Xiaodong Yu, and Dan Roth. 2022. Learning to decompose: Hypothetical question decomposition based on comparable texts. *ArXiv*, abs/2210.16865.

A Implementation Details

Question Decomposition We use the pre-trained T5-large and BART-large models with max_input_length $L = 512$, and max_output_length O = 64. During training, we used the Adam optimizer in the QD modules and set the batch size to 32 and the learning rate to 5e-5. All experiments utilized two TITAN RTX GPUs and 5 hours in total.

Question Answering We choose DeBERTa-v2-large as the backend model and set the number of epochs to 12 and batch size to 4. We use BERTAdam with a learning rate of 5e-6 for the optimization and set max position embeddings to 1024.

B GenDec Quality Analysis

We first quantitatively compare the F measure, ROUGE-1, ROUGE-L, and BLEU scores of the generated sub-question qualities in table 7, and BART-large significantly improves the quality of sub-questions reaching 76.52, 75.89,64.48 and 28.91, respectively. Likely due to the different max input lengths of T5-large (512) and BART-large (1024), BART outperforms it since some inputs contain more sentences (including both the multi-hop questions themselves and retrieved paragraphs).

We further compare the decomposition quality of GenDec and previous QD models in Table 8, which illustrates how GenDec can produce sub-questions with more natural and fluent language with higher quality. Table 10 further shows some examples of our generated sub-questions. The GenDec produced sub-questions are also independent that can be answered in parallel. However, the ModularQA failed to generate fluent sentences and DecompRC can not produce independent sub-questions. To illustrate the generalization of GenDec on different questions, we list 3 multihop questions (including 2, 3, 4-hop questions) and generated sub-questions in table 9, and the results show that our GenDec is robust enough when the complexity of multihop questions improves.

Model	Ans		Sup		Joint	
	EM	F1	EM	F1	EM	F1
GPT-4 w/o GenDec	61.26	78.93	64.25	85.43	48.72	75.31
GPT-4 w GenDec	**64.34**	**82.46**	**67.74**	**88.45**	**52.43**	**78.92**
GPT-3.5 w/o GenDec	57.35	75.11	59.78	76.46	42.37	69.24
GPT-3.5 w GenDec	**60.47**	**77.59**	**63.25**	**80.34**	**47.62**	**72.27**
text-davinci-003 w/o GenDec	40.37	59.46	47.52	62.96	36.62	55.89
text-davinci-003 w GenDec	**44.39**	**65.48**	**50.16**	**67.28**	**40.56**	**61.75**

Table 5: Performance of LLMs (with and without GenDec) on 1000 samples from HotpotQA's dev set distractor setting data.

q	q_{sub1}	q_{sub2}	GPT-4 w GD	GPT-4 w/o GD	GPT-3.5 w GD	GPT-3.5 w/o GD	text w GD	text w/o GD
c	c	c	39.7	30.5	31.1	28.3	29.8	27.4
c	c	w	10.5	9.1	12.3	11.5	9.7	12.3
c	w	c	12.6	9.7	8.6	9.2	8.8	11.8
c	w	w	20.3	21.4	23.8	24.5	20.5	16.2
w	c	c	4.9	8.1	7.8	5.7	9.9	11.5
w	c	w	6.7	7.6	5.7	6.3	6.8	7.6
w	w	c	3.6	6.7	2.6	8.4	7.9	5.8
w	w	w	7.6	5.9	8.1	6.1	6.6	7.4

Table 6: Categorical EM statistics (%) of sub-question evaluation for six multi-hop QA models over HVSQA (Tang et al., 2020). c/w denotes questions answered correctly/wrongly. For example, the fourth row shows the percentage of multi-hop questions that can be correctly answered while sub-questions cannot. We abbreviate GenDec as GD and text-davinci-003 as text.

	Metric			
Models	**F Measure**	**Rouge1**	**Rouge-L**	**BLEU**
BART-LARGE	76.52	75.89	64.48	28.91
T5-LARGE	72.68	74.14	62.22	27.87

Table 7: Generative QD performance of different generative LMs on test instances of HOTPOTQA sub-questions. Results are averaged on 1549 test instances.

GENDEC (OURS)
Sub-question 1: Which South Korean boy group had their debut album in 2014? **Sub-question 2:** WINNER was formed by who?
MODULARQA (Khot et al., 2021)
Sub-question 1: What is the name of the South Korean group that had their debut album in 2014? **Sub-question 2:** What was WINNER formed by?
DECOMPRC (Min et al., 2019b)
Sub-question 1: 2014 S/S is the debut album of which South Korean boy group? **Sub-question 2:** which formed by who ?

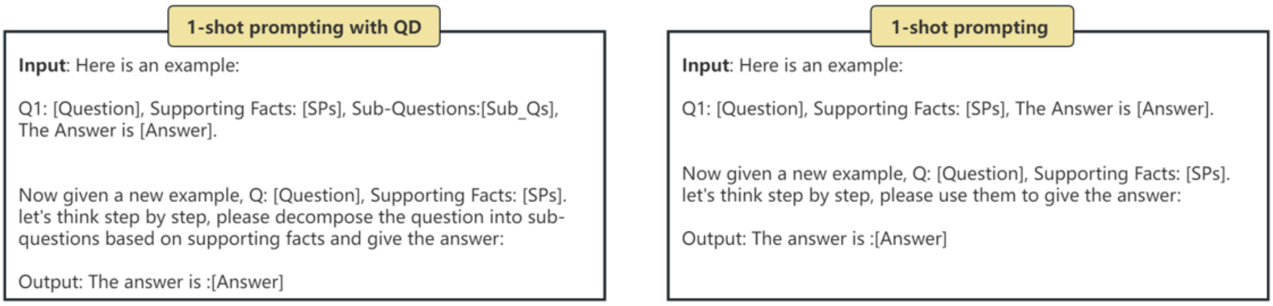
Table 8: QD examples produced by {GENDEC, MODULARQA, DECOMPRC} for question "2014 S/S is the debut album of a South Korean boy group that was formed by who?".

2 HOP QUESTION
Original Question What is the capital city of the country that Michael Feinstein is a citizen of ? **Sub-question 1:** What is the country of citizenship of Michael Feinstein? **Sub-question 2:** What is the capital of the United States of America?
3 HOP QUESTION
Original Question What is the work location of the person who founded the religion of Pius VI? **Sub-question 1:** Which religion is Pius VI associated with? **Sub-question 2:** Who founded Catholic Church? **Sub-question 3:** What is the work location of Jesus Christ?
4 HOP QUESTION
Original Question What is the capital of the country where Mel Daniels' sport originated from? **Sub-question 1:** What position does Mel Daniels play? **Sub-question 2:** Which sport is the center associated with? **Sub-question 3:** Which country was basketball from? **Sub-question 4:** What is the capital of the United States of America?

Table 9: Generalization of {GENDEC}, examples produced by {GENDEC} for different questions (including 2, 3, 4-hop questions).

Original Question	Sub-questions	Intermediate Answers	Answer
Were Scott Derrickson and Ed Wood of the same nationality?	What was Scott Derrickson's nationality? What was Ed Wood's nationality? ✓	American ✓	Yes ✓
What government position was held by the woman who portrayed Corliss Archer in the film Kiss and Tell?	Who portrayed Corliss Archer in Kiss and Tell? What position was held by Shirley Temple? ✓	Shirley Temple ✓	Chief of Protocol ✓
The director of the romantic comedy B̈ig Stone Gapïs based in what New York City neighborhood?	Who is the director of the romantic comedy Big Stone Gap? In what New York City neighborhood is Adriana Trigiani based? ✓	Adriana Trigiani ✓	Greenwich Village ✓
Are Random House Tower and 888 7th Avenue both used for real estate?	The Random House Tower used as real estate? What is 888 7th Avenue used also for? ✗	Used ✗	No ✗
What is the name of the executive producer of the film that has a score composed by Jerry Goldsmith?	What is the name of the film of which Jerry Goldsmith composed the score? Which co-writer of Alien was also an executive producer? ✓	Alien ✓	Francis Ford Coppola ✗
Alvaro Mexia had a diplomatic mission with which tribe of indigenous people?	Who was given a diplomatic mission to the native populations living south of St. Augustine and in the Cape Canaveral area? What is the name of the indigenous tribe of Florida? ✗	Alvaro Mexia ✗	Indigenous peoples of Florida ✗

Table 10: Examples of 3 correct samples and 3 incorrect samples from dev set of HotpotQA



Input: Here is an example:

Q1: [Question], Supporting Facts: [SPs], Sub-Questions:[Sub_Qs], The Answer is [Answer].

Now given a new example, Q: [Question], Supporting Facts: [SPs]. let's think step by step, please decompose the question into sub-questions based on supporting facts and give the answer:

Output: The answer is :[Answer]

Input: Here is an example:

Q1: [Question], Supporting Facts: [SPs], The Answer is [Answer].

Now given a new example, Q: [Question], Supporting Facts: [SPs]. let's think step by step, please use them to give the answer:

Output: The answer is :[Answer]

Figure 3: Prompting examples of different settings.