FSM: A Finite State Machine Based Zero-Shot Prompting Paradigm for Multi-Hop Question Answering

Anonymous ACL submission

Abstract

Large Language Models (LLMs) with chain-of-thought (COT) prompting have demonstrated impressive abilities on simple nature language inference tasks. However, they tend to perform poorly on Multi-hop Question Answering (MHQA) tasks due to several challenges, including hallucination, error propagation and limited context length. We propose a prompting method, Finite State Machine (FSM) to enhance the reasoning capabilities of LLM for complex tasks in addition to improved effectiveness and trustworthiness. Different from COT methods, FSM addresses MHQA by iteratively decomposing a question into multi-turn sub-questions, and self-correcting in time, improving the accuracy of answers in each step. Specifically, FSM addresses one sub-question at a time and decides on the next step based on its current result and state, in an automaton-like format. Experiments on benchmarks show the effectiveness of our method. Although our method performs on par with the baseline on relatively simpler datasets, it excels on challenging datasets like Musique. Moreover, this approach mitigates the hallucination phenomenon, wherein the correct final answer can be recovered despite errors in intermediate reasoning. Furthermore, our method improves LLMs' ability to follow specified output format requirements, significantly reducing the difficulty of answer interpretation and the need for reformatting.

1 Introduction

Multi-hop Question Answering has intrigued researchers for its complexity and practical implications. Researchers employ two primary strategies to address MHQA using Large Language Models. One effective method is In-Context Learning (ICL) (Wang et al., 2023; Zhou et al., 2022), where models are guided to solve problems based on detailed instructions, often through examples of problem decomposition. However, few-shot methods with

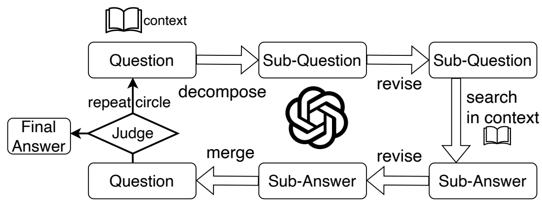

Figure 1: The abstract flow chart of FSM

manual demonstrations are expansive and time-consuming. Another approach involves fine-tuning LLMs with domain-specific data, a complex process (Cao et al., 2023) requiring substantial high-quality data and computational resources. This approach is unable to generalize to unseen datasets and domains without training. Despite advancements in single-hop question answering, MHQA remains challenging due to the need to extract information from lengthy texts and conduct multi-step reasoning without supervision, which poses difficulties for LLMs. LLMs struggle with reading long texts and multi-step reasoning tasks.

Why do LLMs underperform in current MHQA methods? By analyzing errors in existing approaches, we identified four common error types, which will be detailed in Section 4. Specific incorrect examples from common methods are illustrated in Figure 5. We found that LLMs struggle particularly in intermediate reasoning stages, where errors in initial steps can propagate, leading to incorrect conclusions. Additionally, few-shot techniques like REACT (Yao et al., 2022) and SP-COT (Wang et al., 2023) need a minimum of 4-shot displays with long context, surpassing context boundaries.

According to the analysis above, we propose a zero-shot method named **F**inite **S**tate **M**achine prompting (FSM), simplifying the MHQA task into four sub-tasks: decomposing questions, searching for answers in candidate paragraphs, revising the format, judging whether to continue or summa-

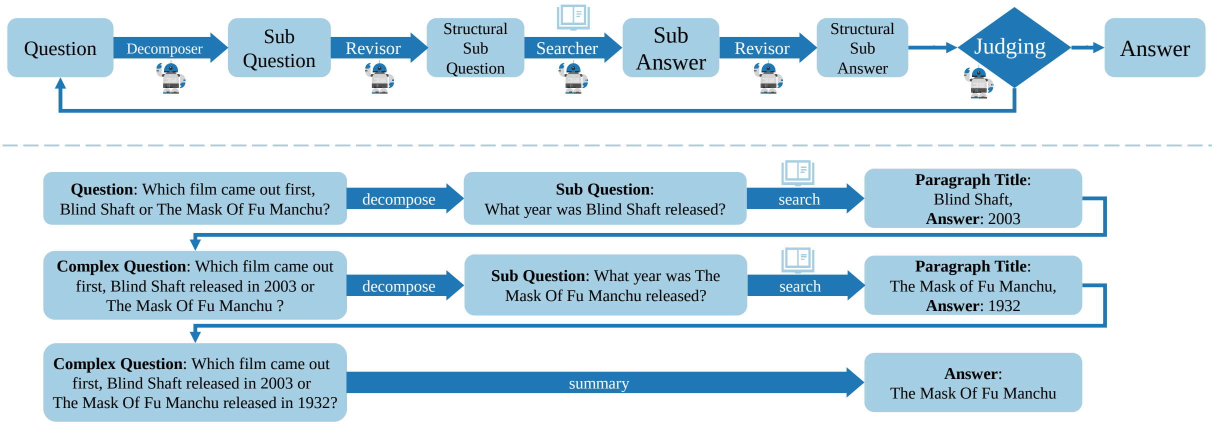

Figure 2: The flow chart of proposed FSM and a simple case in detail. The book icon indicates candidate paragraphs in the search step. The robot denotes LLMs.

rizing with all key information. Figure 2 depicts the process of the FSM. LLMs address one sub-question per round, deciding the next step based on the current state, following an automaton-like process. Clear and explicit sub-tasks, along with timely revisions, make the reasoning process more manageable and accurate.

Extensive experiments on MHQA benchmarks (Yang et al., 2018; Trivedi et al., 2022; Ho et al., 2020) demonstrate that our approach outperforms GPT and 72B LLM baselines, nearly doubling the F1 score on Musique (Trivedi et al., 2022). Furthermore, unlike our framework, baselines have a high frequency of producing outputs in unexpected formats and type errors that require additional processing to extract correct answers.

Our contributions are as follows:

• To address reasoning challenges in LLMs for MHQA tasks, we introduce FSM, a zero-shot prompting paradigm based on finite state machines to decompose complex questions iteratively. This approach aims to strengthen control over intermediate reasoning and improve overall accuracy.

• We investigate the reason for errors in MHQA and conduct various experiments on the insights. e.g. hallucination exists in direct answer predictions, and the contextual length is a bottleneck for reasoning.

• Extensive experiments on MHQA benchmarks in different settings validate FSM's effectiveness, especially on challenging datasets. The method can be adapted to other similar complex tasks.

2 Methodology

2.1 Strategies

The average score of the large model is reported as only 0.3, which is significantly lower than the current sota method (Zhang et al., 2023). Through manual observation of error examples 5, we deduce from the results of baseline methods that the model struggles with completing complex instructions in a single step. LLMs tend to forget previous instructions during reasoning.

To address these issues, we propose the following strategies:

a) **Iterative Decomposition**: Unlike few-shot reasoning approaches, FSM adopts a multi-turn process. Each iteration focuses on addressing a single sub-task, enabling LLMs to understand instructions clearly and execute them accurately.

b) **Error Checking and Backtracking**: For each reasoning step, FSM conducts a verification check to ensure the correctness of response. If an irregular or incorrect output is identified, the model is allowed to self-revise the answer or backtrack.

c) **Final Review Step**: To minimize distractions from lengthy contexts, we utilize sub-questions, corresponding supporting factual paragraphs, evidence, and answers to further verify the consistency of answers and reasoning, named FSM2.

2.2 Framework

We present our proposed Finite State Machine (FSM) in two distinct stages as illustrated in Figure 2. Initially, we instruct LLMs to address sub-questions iteratively during the first phase. Subsequently, in stage 2, LLMs are tasked with summarizing the responses incorporating key information from each sub-question. The FSM framework is depicted in Figure 2.

To elaborate, our approach commences by assisting the model in breaking down the primary question into smaller components. Following this, we compare the original question with the sub-questions to ensure semantic equivalence; any disparities prompt the model to further decompose the elements. In the third phase, the model scans the context for related paragraphs, retrieving relevant information and answers. The fourth step entails revising the complex question with the response to the sub-question and identifying the relationship with updated complex question and sub-question, composition or comparison. Additionally, we conduct checks to ascertain whether the answer constitutes a simple or compound sentence, then promptly breaks down compound sentences. This iterative process continues until the revised question reaches a point where no further decomposition is feasible. By meticulously following each step, our methodology enables a more accurate evaluation of a model's true capabilities, distinguishing it from other approaches that tend to overlook crucial intermediate stages, which may yield seemingly correct outcomes despite flawed reasoning processes. We have included prompts for the whole process in the Appendix.

3 Experiments

3.1 Benchmark and Evaluation

We evaluate our model on three high-quality MHQA datasets: HotpotQA (Yang et al., 2018), 2WikiMultiHopQA (Ho et al., 2020) and Musique (Trivedi et al., 2022). Learning from the shortcut phenomenon (Min et al., 2019) of single hop questions in HotpotQA, Musique strictly controls the composition of the question, ensuring that it must undergo multiple inferences to find the answer. Both HotpotQA and 2Wiki have ten candidate paragraphs for each question and originally have supporting facts. While Musique has twenty candidate with longer text and no supporting facts. Therefore, Musique is the most standard and difficult MHQA datasets. Following traditions (Wang et al., 2023), We adopt the exact match (EM) and F1 scores as evaluation metrics and conduct experiments on subsets of the datasets by randomly selecting 1000 samples from the test sets. Despite having similar basic instructions and a clearly defined output format for all methods, the model's consistency in following instructions may vary across different methods. This variation can result difficulty for answer extraction during evaluation. To address this issue, we introduce a new metric, format, measuring the accuracy of the output format.

3.2 Baselines

Baseline methods in the experiment:

• The **Normal** is the basic form, involving only task descriptions and output requirements, without explicit instructions for reasoning.

• The **COT** (Wei et al., 2022) is widely used in LLMs for inference due to its simplicity and effectiveness. It prompts LLMs to create intermediate step-by-step rationales, aiding in the reasoning process for obtaining answers.

• The **SP-COT** (Wang et al., 2023) introduces a pipeline for generating high-quality Open-Domain Multi-step Reasoning (ODMR) datasets. It utilizes an adaptive sampler for case selection and self-prompted inference via ICL. This technique organizes reasoning chains into six categories, inspired by the construction of the Musique (Trivedi et al., 2022) dataset.

3.3 Setting

Our study explores two settings: (1) generating answers directly from the context and question, and (2) building a complete reasoning chain that includes the answer, supporting evidence, and facts to assess the coherence of the reasoning process. Due to the lack of gold evidence for Setting 2 in Musique, our evaluation can not evaluate on it.

3.4 Models

For MHQA task, we require models with the ability for processing lengthy text. FSM operates in multiple rounds, demanding models capable of handling conversational contexts. We selected GPT-3.5-turbo-32k and Qwen72B-chat (Bai et al., 2023) for our study. Additionally, we employed vllm (Kwon et al., 2023) to accelerate the inference process.

3.5 Results

The results of setting 1(sole answer) are detailed in Table 2, while the outcomes for setting 2(answer paired with supporting fact) are displayed in Table 1. Our approach demonstrates superior results in setting 2, particularly on the most difficult dataset. This is attributed to the increased complexity of instructions in Setting 2, making it harder for models

		Musique			HotpotQA							2wiki						
		ans			ans		sup		joint			ans		sup		joint		
		EM	F1	Format	EM	F1	EM	F1	EM	F1	Format	EM	F1	EM	F1	EM	F1	Format
Qwen	**Normal**	18.2	30.9	84.0	31.6	42.8	**2.6**	26.4	**1.3**	13.4	90.7	6.7	8.0	1.6	5.5	1.0	**2.6**	89.8
	COT	1.0	6.6	7.0	3.1	9.7	0.1	0.7	0.1	0.4	4.4	0.6	1.9	0	0.1	0.0	0.0	4.2
	FSM1	**26.2**	**41.2**	100.0	22.5	33.3	0.7	9.9	0.4	3.6	100.0	27.6	37.9	4.7	25.8	1.9	9.1	100.0
	FSM2	21.9	37.7	100.0	**33.1**	**46.0**	1.8	**28.8**	1.0	**15.7**	100.0	**36.1**	**49.3**	**7.7**	**38.4**	**5.1**	**19.4**	100.0
GPT	**Normal**	16.7	27.8	94.0	34.0	45.9	0.7	15.0	3.0	8.0	94.3	**37.3**	**46.6**	1.0	14.1	**9.0**	7.2	95.8
	COT	4.5	13.6	14.7	12.3	26.0	0.4	4.5	2.0	17.8	16.2	8.2	19.3	0.2	1.3	1.0	4.6	7.0
	FSM1	**26.0**	**38.4**	100.0	23.4	32.0	**2.4**	**29.3**	2.0	9.8	100.0	30.1	40.0	**14.2**	**47.0**	2.0	8.5	100.0
	FSM2	18.6	27.4	100.0	**28.4**	**36.7**	2.2	21.4	**4.0**	**26.7**	100.0	30.6	37.2	6.9	29.6	7.0	**19.8**	100.0

Table 1: Results on the MHQA benchmark by the gpt-3.5-turbo-1106 and Qwen-72B with zero-shot in setting 2. Ans means answer. Sup means supporting paragraph index and tile. Joint means evidence triples including relationship with sub-answers. FSM2 means LLMs summary with results of FSM1 again

		Musique		HotpotQA		2Wiki	
		EM	F1	EM	F1	EM	F1
GPT	**Normal**	19.2	33.3	31.9	43.7	36.0	46.6
	COT	20.6	35.6	32.1	45.5	38.1	**53.0**
	SP-COT	14.4	28.4	24.8	37.4	23.2	36.0
	FSM1	23.1	40.3	24.5	39.3	27.1	40.6
	FSM2	**26.7**	**40.5**	**33.3**	**45.7**	**39.2**	50.1
Qwen	**Normal**	12.9	19.9	31.0	41.6	31.9	39.1
	COT	14.1	24.0	30.6	**42.7**	39.9	49.8
	SP-COT	6.0	14.7	14.6	28.6	18.5	31.8
	FSM1	**33.2**	**48.5**	28.0	37.4	39.1	47.9
	FSM2	**33.2**	**48.5**	**32.2**	41.3	**40.2**	**50.3**

Table 2: Results on the MHQA benchmark by the gpt-3.5-turbo-1106 and Qwen-72B in setting 1.

to follow them accurately. Furthermore, the presence of straightforward single-hop questions in the HotpotQA and 2Wiki datasets (Min et al., 2019) can confuse the LLMs with multi-hop reasoning. While our method's performance in Setting 1 on less complex datasets like HotpotQA and 2Wiki is moderately satisfactory, it excels in precision with fewer instances of hallucination.

The performance of COT is notably inadequate, falling considerably below the standard few-shot settings. This discrepancy is mainly due to its failure to provide answers in the required format, detailed in Figure 4, a flaw we attribute to its bad instruction following ability. Conversely, the normal method struggles with supporting facts but achieves substantially higher scores on answers. This phenomenon indicates that although LLMs may misinterpret intermediate reasoning steps, they still yield correct answers, hinting at underlying data leakage and speculating. While some errors may stem from misinterpreting instructions, it is evident that there are significant concerns surrounding the authenticity and logical coherence of the models' reasoning chains. Additionally, the prospect of dataset leakage during evaluation cannot be disregarded. In conclusion, we posit that our method maintains a competitive edge in this context.

4 Discussion

Figure. 5 provides error examples in experiments. We conclude four types of errors. **a)Reasoning Lost Issue**: providing an answer just with the last sub-question, instead of options for the original sentence. **b)Formatting Error**: The output can not be parsed to get answer, which added difficulty to the evaluation. Examples are presented in Figure 4. **c)Sub-question Decomposition Error**: Incorrectly decomposed the sub-questions. **d)Sub-answer Error**: Identified the wrong paragraph but provided a correct answer. **e)Hallucination Response**: Provided an correct answer without locating the relevant paragraph.

5 Conclusion

We have identified issues in traditional methods where LLMs may produce errors in the intermediate reasoning process but still arrive at the correct answer. Additionally, these methods often require few-shot demonstrations, which may surpass the maximum context length of LLMs. Therefore, we propose an easy zero-shot prompt paradigm called the FSM to address MHQA tasks systematically in an automated format. Our framework approaches problem-solving by focusing on one sub-task at a time iteratively, revising each step to ensure precision. By guiding LLMs through problems incrementally, FSM achieves superior results and aids in enhancing the LLMs' capabilities without resorting to shortcuts.

Limitations

This multi-turn dialogue process, inherent to our framework, mandates repeated handling of improperly formatted outputs, due to the output before will be the next input, which can be challenging for models with smaller parameter sizes and weaker follow-instruction capabilities. Therefore, models with limited capacity to follow instructions might not benefit from our method as any error in the intermediate steps could lead to an abrupt termination of the process.

References

Jinze Bai, Shuai Bai, Yunfei Chu, Zeyu Cui, Kai Dang, Xiaodong Deng, Yang Fan, Wenbin Ge, Yu Han, Fei Huang, Binyuan Hui, Luo Ji, Mei Li, Junyang Lin, Runji Lin, Dayiheng Liu, Gao Liu, Chengqiang Lu, Keming Lu, Jianxin Ma, Rui Men, Xingzhang Ren, Xuancheng Ren, Chuanqi Tan, Sinan Tan, Jianhong Tu, Peng Wang, Shijie Wang, Wei Wang, Shengguang Wu, Benfeng Xu, Jin Xu, An Yang, Hao Yang, Jian Yang, Shusheng Yang, Yang Yao, Bowen Yu, Hongyi Yuan, Zheng Yuan, Jianwei Zhang, Xingxuan Zhang, Yichang Zhang, Zhenru Zhang, Chang Zhou, Jingren Zhou, Xiaohuan Zhou, and Tianhang Zhu. 2023. Qwen technical report.

Maciej Besta, Nils Blach, Ales Kubicek, Robert Gerstenberger, Lukas Gianinazzi, Joanna Gajda, Tomasz Lehmann, Michal Podstawski, Hubert Niewiadomski, Piotr Nyczyk, et al. 2023. Graph of thoughts: Solving elaborate problems with large language models. *arXiv preprint arXiv:2308.09687*.

Hejing Cao, Zhenwei An, Jiazhan Feng, Kun Xu, Liwei Chen, and Dongyan Zhao. 2023. A step closer to comprehensive answers: Constrained multi-stage question decomposition with large language models. *arXiv preprint arXiv:2311.07491*.

Yuwei Fang, Siqi Sun, Zhe Gan, Rohit Pillai, Shuohang Wang, and Jingjing Liu. 2020. Hierarchical graph network for multi-hop question answering. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 8823–8838, Online. Association for Computational Linguistics.

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*, Punta Cana, Dominican Republic. Association for Computational Linguistics.

Xanh Ho, Anh-Khoa Duong Nguyen, Saku Sugawara, and Akiko Aizawa. 2020. Constructing a multi-hop qa dataset for comprehensive evaluation of reasoning steps. In *Proceedings of the 28th International Conference on Computational Linguistics*.

Takeshi Kojima, Shane Shixiang, Gu Gu, Machel Reid, Yutaka Matsuo, and Yusuke Iwasawa. Large language models are zero-shot reasoners.

Woosuk Kwon, Zhuohan Li, Siyuan Zhuang, Ying Sheng, Lianmin Zheng, Cody Hao Yu, Joseph E. Gonzalez, Hao Zhang, and Ion Stoica. 2023. Efficient memory management for large language model serving with pagedattention. In *Proceedings of the ACM SIGOPS 29th Symposium on Operating Systems Principles*.

Sewon Min, Eric Wallace, Sameer Singh, Matt Gardner, Hannaneh Hajishirzi, and Luke Zettlemoyer. 2019. Compositional questions do not necessitate multi-hop reasoning. *arXiv: Computation and Language,arXiv: Computation and Language*.

Ethan Perez, Patrick Lewis, Wen-tau Yih, Kyunghyun Cho, and Douwe Kiela. 2020a. Unsupervised question decomposition for question answering. *arXiv preprint arXiv:2002.09758*.

Ethan Perez, Patrick Lewis, Wen-tau Yih, Kyunghyun Cho, and Douwe Kiela. 2020b. Unsupervised question decomposition for question answering. *arXiv: Computation and Language,arXiv: Computation and Language*.

Peng Qi, Xiaowen Lin, Leo Mehr, Zijian Wang, and ChristopherD. Manning. 2019. Answering complex open-domain questions through iterative query generation. *Cornell University - arXiv,Cornell University - arXiv*.

Mokanarangan Thayaparan, Marco Valentino, Viktor Schlegel, and André Freitas. 2019. Identifying supporting facts for multi-hop question answering with document graph networks. In *Proceedings of the Thirteenth Workshop on Graph-Based Methods for Natural Language Processing (TextGraphs-13)*, pages 42–51, Hong Kong. Association for Computational Linguistics.

Harsh Trivedi, Niranjan Balasubramanian, Tushar Khot, and Ashish Sabharwal. 2022. musique: Multihop questions via single-hop question composition. *Transactions of the Association for Computational Linguistics*, page 539–554.

Ming Tu, Guangtao Wang, Jing Huang, Yun Tang, Xiaodong He, and Bowen Zhou. 2019. Multi-hop reading comprehension across multiple documents by reasoning over heterogeneous graphs. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 2704–2713, Florence, Italy. Association for Computational Linguistics.

Jinyuan Wang, Junlong Li, and Hai Zhao. 2023. Self-prompted chain-of-thought on large language models for open-domain multi-hop reasoning. In *Findings of the Association for Computational Linguis-*

tics: EMNLP 2023, pages 2717–2731, Singapore. Association for Computational Linguistics.

Xuezhi Wang, Jason Wei, Dale Schuurmans, Quoc Le, Ed Chi, Sharan Narang, Aakanksha Chowdhery, and Denny Zhou. 2022. Self-consistency improves chain of thought reasoning in language models. *arXiv preprint arXiv:2203.11171*.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Fei Xia, Ed Chi, Quoc V Le, Denny Zhou, et al. 2022. Chain-of-thought prompting elicits reasoning in large language models. *Advances in Neural Information Processing Systems*, 35:24824–24837.

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

Shunyu Yao, Jeffrey Zhao, Dian Yu, Nan Du, Izhak Shafran, Karthik Narasimhan, and Yuan Cao. 2022. React: Synergizing reasoning and acting in language models. *arXiv preprint arXiv:2210.03629*.

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*.

Denny Zhou, Nathanael Schärli, Le Hou, Jason Wei, Nathan Scales, Xuezhi Wang, Dale Schuurmans, Olivier Bousquet, Quoc Le, and Ed Chi. 2022. Least-to-most prompting enables complex reasoning in large language models.

Appendix

A Related Work

Multi-hop Question Answering Existing approaches to solving the multi-hop QA task can be mainly categorized into question decomposition (Perez et al., 2020a; Fu et al., 2021; Perez et al., 2020b), graph-based method (Tu et al., 2019; Thayaparan et al., 2019; Fang et al., 2020), iterative method (Qi et al., 2019) and LLMs (Wang et al., 2023) prompts. These models grapple with computational complexity and extensibility, and they lack an interpretable reasoning chain, which deviates from human cognitive processes.

Language model for reasoning. CoT(Wei et al., 2022) reveals the ability of large language models to formulate their reasoning procedure for problem-solving. Several follow-up works have since been performed, including the least-to-most prompting technique (Zhou et al., 2022) for solving complicated tasks, zero-shot CoT (Kojima et al.), graph-of-thought (GoT) (Besta et al., 2023), and reasoning with self-consistency (Wang et al., 2022). ReAct (Yao et al., 2022) interleaves the generation of reasoning traces with task-specific actions, promoting greater synergy.

Task decomposition. (Perez et al., 2020a) decomposes a multi-hop question into a number of independent single-hop sub-questions, which are answered by an off-the-shelf question-answering (QA) model. These answers are then aggregated to form the final answer. Both question decomposition and answer aggregation require training models. After the emergence of Large Language Models (LLMs), traditional training methods (Cao et al., 2023) are rarely used due to their expensive nature. Most current research focuses on the few-shot approach. (Zhou et al., 2022) chains the processes of problem decomposition and sub-problem solving. The original problem and its sub-problems are inherently interrelated, and forcibly breaking them down into unrelated problems would unnecessarily increase the difficulty.

B Prompt

B.1 FSM1

Decomposer = Please determine whether the question is simple sentence or compound sentence. If it is a simple sentence, return "simple":true,"subquestion’:null.Otherwise, simple: false, decompose the question and generate the first answerable simple sentence. reply in the form of "simple":false,"subquestion":xxx. Do not reply any other words and provide answers in JSON format!

Searcher = ”’Given the paragraph below, please find out the paragraph that contains the answer of "" Please take a moment to thoroughly understand the content before proceeding to the questions, then carefully read the relevant paragraphs based on the question and provide the most likely answer. Return the title of the paragraph and the answer no more than 5 words in the form of "question":xxx, "paragraph title":xxx, "answer":xxx. Do not reply any other words and provide answers in JSON format!”’

Judge-if-continue=”’Please compare the complex question and subquestion, answer whether they are semanically identical in the form of "identical":true or false. Do not reply any other words and provide answers in JSON format!”’

B.2 FSM2

FSM2-post-summary-again="'Documents: paragraphs:paragraphs found in FSM1 subquestion and answers:subquestion and answers given in FSM1 Question:origin question Answer the question reasoning step-by-step based on the Doucments. If it is a general question, please respond with 'Yes' or 'No'. Finally, you must return the title of the context, the sentence index (start from 0) of the paragraph and the concise answer no more than 10 words and explaination in the form of "supporting-facts": [[title, sentence id], ...], "evidences": [[subject entity, relation, object entity],...], "answer":"xxx","explain":"xxxx". Do not reply any other words."'

B.3 Baseline

SP-COT(Wang et al., 2023)="'This is a two-hop to four-hop reasoning question-answering task that requires decomposing the questions into simple, answerable single-hop questions. The decomposition process involves four types of questions: comparison, inference, compositional, and bridge-comparison. There are six specific decomposition steps in total, denoted by Q* representing the decomposed subproblems. The steps are as follows: First, Q1 -> Q2 Second, Q1 -> Q2 -> Q3 Third, Q1 -> Q2 -> Q3 Fourth, (Q1&Q2) -> Q3 Fifth, (Q1&Q2) -> Q3; Q3 -> Q4 Sixth, Q1 -> Q2; (Q2&Q3) -> Q4 The process involves first determining the type of question and then identifying the decomposition process type. It's important to note that the decomposition of questions cannot be provided all at once; it must be done step by step. Each subproblem needs to be decomposed and answered before moving on to the next one, as there is interdependence between the subproblems .Finally, you must return the title of the context, the sentence index (start from 0) of the paragraph and the concise answer and explaination in the form of "explain":"xxxx","supporting-facts": [[title, sentence id], ...], "evidences": [[subject entity, relation, object entity],...],"answer":"no sentence and no more than 10 words ". Do not reply any other words."'

COT-setting1-w/o-evidence = "'Answer the question according to the context,Let's think step by step, and explain your reasoning process. You must return in the form of "explain":"xxxx","answer":answer. Do not reply any other words."'

normal-setting1-w/o-evidence = "'Answer the question according to the context. You must return in the form of "explain":"xxxx","answer":answer. Do not reply any other words."'

normal-setting2-w-evidence = "'Answer the question according to the context. Find the paragraph that contains the answer of question, and summarize a triple that contains [subject entity, relation, object entity]. Finally, you must return the title of the context, the sentence index (start from 0) of the paragraph and the concise answer no more than 10 words in the form of "supporting-facts": [[title, sentence id], ...], "evidences": [[subject entity, relation, object entity],...], "answer":answer. Do not reply any other words."'

prompt-step = "'Answer the question according to the context,Let's think step by step, and explain your reasoning process. Find the paragraph that contains the answer of question, and summarize a triple that contains [subject entity, relation, object entity]. Finally, you must return the title of the context, the sentence index (start from 0) of the paragraph and the concise answer no more than 10 words in the form of "supporting-facts": [[title, sentence id], ...], "evidences": [[subject entity, relation, object entity],...], "answer":answer. Do not reply any other words."'

React-setting2-w-evidence = """Solve a question answering task with interleaving Thought, Action, Observation steps. Thought can reason about the current situation, and Action can be three types: (1) Search[entity], which searches the exact entity on given context and returns the first paragraph if it exists. If not, it will return some similar entities to search. (2) Lookup[keyword], which returns the next sentence containing keyword in the current passage. (3) Finish[results], which returns the answer and finishes the task. You should plan and reason in the 'Thought', then perform your 'Action', lastly, observe the result of action. Loop this process until the problem was finished. At last, you must additional output the title of the paragraphs, the sentence index (start from 0) of the paragraph and the concise answer no more than 10 words and explaination in the form of Thought: reasoning Action: Search[entity] or Lookup[keyword] or Finish[results] Observation: observe the results of action end with Finish["supporting-facts": [[title, sentence id], ...], "evidences": [[subject entity, relation, object entity],...], "answer":answer] """

B.4 Format Error

Gold-answer

{"question": "What government position was held by the woman who portrayed Corliss Archer in the film Kiss and Tell?", "answer": "Chief of Protocol", "type": "bridge", "supporting_facts": [["Kiss and Tell (1945 film)", 0], ["Shirley Temple", 0], ["Shirley Temple", 1]], "level": "hard"}

{"question": "What science fantasy young adult series, told in first person, has a set of companion books narrating the stories of enslaved worlds and alien species?", "answer": "Animorphs", "type": "bridge", "supporting_facts": [["The Hork-Bajir Chronicles", 0], ["The Hork-Bajir Chronicles", 1], ["The Hork-Bajir Chronicles", 2], ["Animorphs", 0], ["Animorphs", 1]], "level": "hard"}

{"question": "Are the Laleli Mosque and Esma Sultan Mansion located in the same neighborhood?", "answer": "no", "type": "comparison", "supporting_facts": [["Laleli Mosque", 0], ["Esma Sultan Mansion", 0]], "level": "hard"}

{"question": "The director of the romantic comedy \"Big Stone Gap\" is based in what New York city?", "answer": "Greenwich Village, New York City", "type": "bridge", "supporting_facts": [["Big Stone Gap (film)", 0], ["Adriana Trigiani", 0]], "level": "hard"}

FSM

{"answer": "Ambassador to Ghana and Czechoslovakia, Chief of Protocol of the United States.", "supportin_facts": [["Kiss and Tell (1945 film)", 0], ["Shirley Temple", "1"], ["Shirley Temple", "1"]], "evidences": [["Kiss and Tell (1945 film)", "portrayed", "Shirley Temple"], ["Shirley Temple Black", "was named", "United States ambassador to Ghana and to Czechoslovakia and also served as Chief of Protocol of the United States"], ["Shirley Temple Black", "had", "notable achievements outside of her acting career"]]}

{"answer": "The Hork-Bajir Chronicles", "supportin_facts": [["The Hork-Bajir Chronicles", "2"], ["The Hork-Bajir Chronicles", "0"]], "evidences": [["Animorphs series", "involves", "enslaved worlds and alien species"], ["The Hork-Bajir Chronicles", "involving", "enslaved worlds and alien species"]]}

{"answer": "Ortaköy neighborhood of Istanbul, Turkey", "supportin_facts": [["Esma Sultan Mansion", "0"], ["Esma Sultan Mansion", "0"]], "evidences": [["Esma Sultan Mansion", "location", "Ortaköy neighborhood of Istanbul, Turkey"]]}

{"answer": "Greenwich Village", "supportin_facts": [["Big Stone Gap (film)", "0"], ["Adriana Trigiani", "0"]], "evidences": [["Big Stone Gap", "director", "Adriana Trigiani"], ["Adriana Trigiani", "based in", "Greenwich Village, New York City"]]}

Figure 3: The outputs of FSM are standard json format.

COT


```
{"answer": "To answer the question and provide a reasoning process, first, I will search for information related to the woman who portrayed Corliss Archer in the film Kiss and Tell. Then, I will search for the film Kiss and Tell in the given context to find the relevant paragraph. After finding the relevant paragraph, I will summarize a triple that contains [subject entity, relation, object entity] and return the title of the context, the sentence index (start from 0) of the paragraph, and the concise answer.\n\nUpon analyzing the context, I found the relevant paragraph in the section \"Kiss and Tell (1945 film)\". The paragraph contains the information related to the woman who portrayed Corliss Archer, which is Shirley Temple.\n\nTriple: [\"Shirley Temple\", \"portrayed\", \"Corliss Archer\"]\n\nTitle: Meet Corliss Archer\n\nSentence Index: 6\n\nConcise Answer: Shirley Temple portrayed Corliss Archer.\n\nTherefore, the final output is:\n```json\n{\n  \"supporting_facts\": [[\"Meet Corliss Archer\", 6]],\n  \"evidences\": [[\"Shirley Temple\", \"portrayed\", \"Corliss Archer\"]],\n  \"answer\": \"Shirley Temple portrayed Corliss Archer.\"\n}\n```", "question": "\"What government position was held by the woman who portrayed Corliss Archer in the film Kiss and Tell?\""}
```

```
{ "answer": "To find the answer, let's first identify the science fantasy young adult series told in first person. The paragraph that contains the relevant information is from the \"Animorphs\" section. The triple containing the solution is [\"Animorphs\", \"is a science fantasy series of young adult books written by\", \"Katherine Applegate and her husband Michael Grant\"]. The title of the context, sentence index, and the concise answer are as follows:\n\nTitle: \"Animorphs\"\nSentence Index: 0\nAnswer: \"Animorphs\"", "question": "\"What science fantasy young adult series, told in first person, has a set of companion books narrating the stories of enslaved worlds and alien species?\""}
```

```
{"answer": "To find out if the Laleli Mosque and Esma Sultan Mansion are located in the same neighborhood, first, I need to locate the paragraph that contains the relevant information about the locations of both entities. Then, checking the information from the paragraph, the relation \"located in\" will be identified between \"Laleli Mosque\" and \"Laleli, Fatih\" or \"Esma Sultan Mansion\" and \"Ortaköy neighborhood\", to conclude if they are in the same neighborhood or not.\n\nThe relevant paragraph is:\n\n[\"Laleli Mosque\", [\"The Laleli Mosque (Turkish: \\\"Laleli Camii, or Tulip Mosque\\\" ) is an 18th-century Ottoman imperial mosque located in Laleli, Fatih, Istanbul, Turkey.\"]],\n[\"Esma Sultan Mansion\", [\"The Esma Sultan Mansion (Turkish: \\\"Esma Sultan Yal\\u0131s\\u0131\\\" ), a historical yal\\u0131 (English: waterside mansion ) located at Bosphorus in Ortak\\u00f6y neighborhood of Istanbul, Turkey and named after its original owner Esma Sultan, is used today as a cultural center after being redeveloped.\"]],\n\nThe triple that contains the necessary information is: \n[\"Laleli Mosque\", \"located in\", \"Laleli, Fatih\"]\n[\"Esma Sultan Mansion\", \"located at\", \"Bosphorus in Ortaköy neighborhood\"]\n\nBased on this information, the concise answer is: \"No, they are not located in the same neighborhood.\"\n\nNow, I will use this evidence to provide a summary triple and supporting facts.\nThe summary triple is:\n{\"supporting_facts\": [[\"Esma Sultan Mansion\", 6], [\"Laleli Mosque\", 5]], \"evidences\": [[\"Laleli Mosque\", \"located in\", \"Laleli, Fatih\"], [\"Esma Sultan Mansion\", \"located at\", \"Bosphorus in Ortaköy neighborhood\"]], \"answer\":\"No, they are not located in the same neighborhood.\"}", "question": "\"Are the Laleli Mosque and Esma Sultan Mansion located in the same neighborhood?\""}\
```

```
{"answer": "To find the answer, let's start by identifying the paragraph containing information about the director of the romantic comedy \"Big Stone Gap.\" Upon examining the paragraphs, we can see that the paragraph about \"Big Stone Gap (film)\" contains the relevant information.\n\nTriple: [\"Adriana Trigiani\", \"is the director of\", \"Big Stone Gap\"]\n\nThen, the concise answer is: \"Adriana Trigiani\"\n\nBased on this, the requested details are as follows:\n- Title of the context: \"The director of the romantic comedy \"Big Stone Gap\" is based in what New York city?\"\n- Sentence index: 8\n- Concise answer: \"Adriana Trigiani\"\n- Supporting facts: [[\"Big Stone Gap (film)\", 0]]\n- Evidence: [[\"Adriana Trigiani\", \"is the director of\", \"Big Stone Gap\"]]", "question": "\"The director of the romantic comedy \\\"Big Stone Gap\\\" is based in what New York city?\""}
```


Figure 4: There are some error format examples for COT.

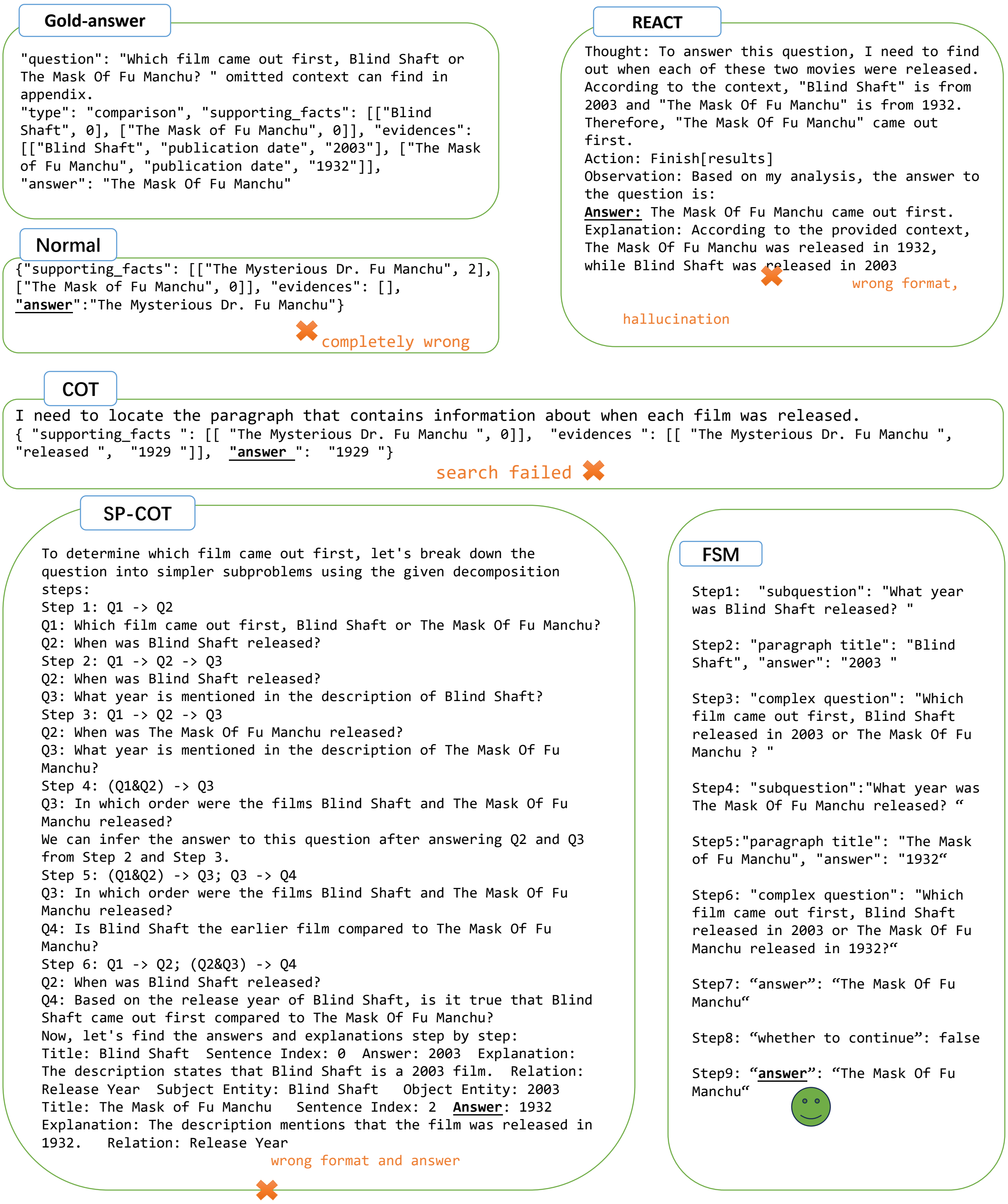

Figure 5: Contrast between baseline and FSM. There are some error examples for baseline.