RE-EX : REVISING AFTER EXPLANATION REDUCES THE FACTUAL ERRORS IN LLM RESPONSES

Juyeon Kim[1]   **Jeongeun Lee**[1*]   **Yoonho Chang**[1*]
Chanyeol Choi[2]   **Junseong Kim**[2†]   **Jy-yonh Sohn**[1†]

[1]Yonsei University   [2]Linq

{culy1125, ljeadec31, yoonho990625, jysohn1108}@yonsei.ac.kr
{jacob.choi, junseong.kim}@getlinq.com

ABSTRACT

Mitigating hallucination issues is a key challenge that must be overcome to reliably deploy large language models (LLMs) in real-world scenarios. Recently, various methods have been proposed to detect and revise factual errors in LLM-generated texts, in order to reduce hallucination. In this paper, we propose RE-EX, a method for post-editing LLM-generated responses. RE-EX introduces a novel reasoning step dubbed as the factual error *explanation step*. RE-EX revises the initial response of LLMs using 3-steps : first, external tools are used to retrieve the evidences of the factual errors in the initial LLM response; next, LLM is instructed to *explain* the problematic parts of the response based on the gathered evidence; finally, LLM revises the initial response using the *explanations* provided in the previous step. In addition to the *explanation step*, RE-EX also incorporates new prompting techniques to reduce the token count and inference time required for the response revision process. Compared with existing methods including FacTool, CoVE, and RARR, RE-EX provides better detection and revision performance with less inference time and fewer tokens in multiple benchmarks. The code for RE-EX is available at: https://github.com/juyeonnn/ReEx.

1 INTRODUCTION

Large Language Models (LLMs) exhibit remarkable text generation abilities and deep language understanding. However, a significant challenge persists: the reliability of LLM-generated text is often compromised due to frequent instances of hallucinations (Ji et al., 2023). Even widely used language models, including ChatGPT, suffer from this issue. This unreliability severely hinders adoption in real-world scenarios where accurate, up-to-date information is essential. Therefore, detecting and correcting factual errors in LLM-generated text is crucial. To tackle this issue, there have been some approaches which instructed LLMs to self-refine their responses (Tonmoy et al., 2024). For example, the authors of (Manakul et al., 2023; Mündler et al., 2023) focus on detecting *self-contradiction* hallucination, while other methods extended the reasoning steps to reduce hallucination(Dhuliawala et al., 2023; Zhao et al., 2023). However, as discussed in (Mallen et al., 2023), relying solely on their internal knowledge base still has a probability of generating texts that contain factual errors.

While Retrieval-Augmented Generation (RAG) (Lewis et al., 2020) based methods mitigate the aforementioned challenges by leveraging information retrieved from external sources, a significant challenge persists: effectively retrieving and processing the necessary information as evidence. Determining how to handle or segment content, given the length and complexity of both the original response and gathered evidence, becomes crucial. Gao et al. (2023) proposes a method that iterates through the evidence one by one to check whether the evidence aligns with the response, and revises the response accordingly. Several fact-checking methods such as (Chern et al., 2023) and (Li et al., 2023) focus on factual error detection by first splitting the response into claims and then generating queries to verify each claim. However, both iterative evidence checking and claim-by-claim verification can be time-consuming and costly, limiting their use in real-world scenarios.

---

* Equal contribution.   † Corresponding authors.

Prompt P: Which country or city has the maximum number of nuclear power plants?
Initial Response R_{initial} : <u>The United States</u> has the highest number of nuclear power plants in the world, with <u>94 operating reactors</u>. Other countries with a significant number of nuclear power plants ...

Sub-Question Q_1 : <u>How many operating reactors</u> does the United States have?
Sub-Answer A_1 : ... the United States had <u>93 operating commercial reactors</u> at 54 nuclear power plants ...

Explanation of Factual Error E_1 : The initial response states that the United states has **94 operating reactors**, but the correct number is **93 operating reactors** .

Revised Response R_{revised} : The United States has the highest number of nuclear power plants in the world, with ~~94 operating reactors~~ **93 operating reactors** . Other countries with a significant number of ...

Figure 1: An example of revising the response with RE-EX. The initial response R_{initial} states the US has 94 operating reactors, proven inaccurate by search result A_1. RE-EX *explains* the factual error in E_1 and revises the response accordingly.

In this paper, we propose RE-EX, which aims to post-edit factual errors in LLM-generated responses using evidence gathered from external sources. Building on research demonstrating significant LLM reasoning improvement through *intermediate reasoning steps* (Wei et al., 2023; Deng et al., 2023; Chowdhery et al., 2022), *e.g.,* chain-of-thought prompting, we introduce an *explanation step* to reduce hallucination errors. RE-EX first gathers evidences of factual errors in the response, and then *explains* the factual errors in the response, and finally revise the response accordingly (as shown in Figure 1). Our contributions are significant in three primary aspects:

1. Instead of making LLMs *directly* revise the response, RE-EX employs the process of *revising after explanation*: we let LLMs first *explain* the factual errors (based on the evidences), and then let LLMs revise their response accordingly. This intermediate *explanation step* leads to better revision performance, as confirmed by our ablation study.

2. Compared with baselines including FacTool (Chern et al., 2023), CoVe (Dhuliawala et al., 2023) and RARR (Gao et al., 2023), our method RE-EX enjoys higher revision performances in multiple benchmarks (FactPrompt (Chern et al., 2023), WiCE (Kamoi et al., 2023)) and in the annotated dataset given in (Min et al., 2023).

3. RE-EX enhances efficiency, achieving up to 10x faster processing and 6x fewer tokens, by integrating certain steps with better prompting. Instead of generating search queries after splitting the response into claims, RE-EX employs a single-step process directly prompting LLM to decompose text and generate sub-questions simultaneously (inspired by the success of question decomposition method (Radhakrishnan et al., 2023)). Additionally, rather than iterative evidence checking, RE-EX presents all evidence at once, instructing LLMs to *explain* factual errors.

2 METHOD

Suppose a LLM generates its initial response R_{initial} given a prompt P. RE-EX revises this response in three steps. In step 1, the LLM generates N sub-questions $\{Q_i\}_{i=1}^N$ related to the prompt and the initial response. External search tools are then used to obtain answers for each question, denoted by $\{A_i\}_{i=1}^N$, thus forming N evidence pairs $\{(Q_i, A_i)\}_{i=1}^N$. In step 2, the LLM is instructed to *explain* each factual error in the response using the evidence pairs $\{(Q_i, A_i)\}_{i=1}^N$. The M *explanations* of factual errors produced in this stage are denoted by $\{E_j\}_{j=1}^M$. Finally, in step 3, RE-EX produces the final revised response R_{revised} using $\{E_j\}_{j=1}^M$, yielding a more factually accurate and refined response with corrected factual errors. The overview of RE-EX is in Fig.2, and the prompt used in our method is in Appendix E. Below we provide a detailed explanation on each step:

Step 1. Evidence Retrieval RE-EX first decomposes the initial response R_{initial}, along with the given prompt P, into a series of self-contained, concise sub-questions. The prompt used for generating sub-question is given in Table 11 in Appendix D. Using the sub-question Q_i as search queries, relevant snippets such as as answer boxes, Knowledge Graph, and organic search results are retrieved using Google Search[1]. These search results are denoted as sub-answer A_i for each sub-question Q_i.
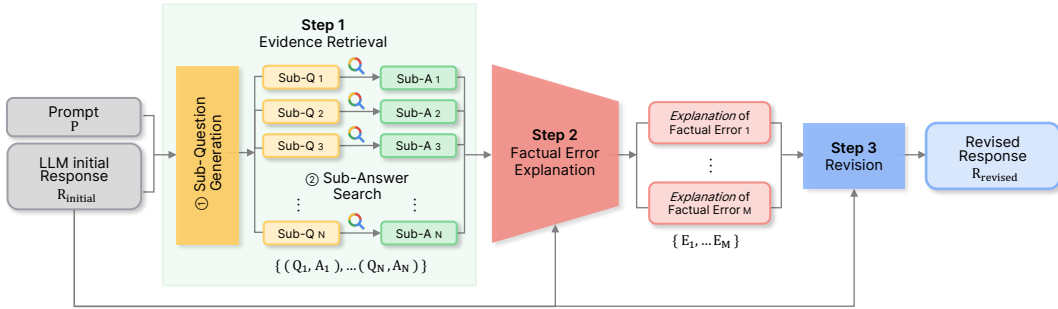
---

[1] https://serper.dev

Figure 2: Overview of RE-EX, which revises the initial response R_{initial} of LLMs. First, RE-EX retrieves the evidences for factual errors in R_{initial} by using external tools; it first generates the sub-questions $\{Q_i\}_{i=1}^N$ useful for checking the factual errors and then gets the answers (or evidences) $\{A_i\}_{i=1}^N$ from external sources. Second, RE-EX lets LLMs explain the factual errors in R_{initial} based on the evidences, thus getting the explanations $\{E_i\}_{i=1}^M$. Finally, LLM revises its response based on the explanations, outputting the revised response R_{revised}.

Step 2. Factual Error Explanation Using the pairs of sub-question and sub-answer, denoted as $\{(Q_i, A_i)\}_{i=1}^N$, RE-EX employs *explanation step* to pinpoint and explain each factual error. The *explanation* E_i of each factual error includes both the factual error in the initial response R_{initial} and the factually accurate information, which corrects the error, retrieved from $\{(Q_i, A_i)\}_{i=1}^N$. The prompt used for factual error explanation is given in Table 13.

Step 3. Revision The revised response R_{revised} is generated using the explanation $\{E_i\}_{i=1}^M$ of the factual error to revise the initial response R_{initial}. Two prompting options are evaluated: *one-step*, where both the revision step and the explanation step are conducted within a single prompt, and *two-step*, where both steps are conducted with two separate prompts. The prompt used for the first option is in Table 12 and the prompts for the second option is in Tables 13 and 14.

3 EXPERIMENTS

3.1 EXPERIMENTAL SETUP

Baselines. We compare RE-EX with three baselines: CoVE (Dhuliawala et al., 2023) refines responses through self-verification, tested in factored (CoVE *(F)*) and factor+revise (CoVE *(F+R)*) variants; RARR (Gao et al., 2023) which *revises* responses via an external search method; and FacTool (Chern et al., 2023) which employs external searches for detailed factuality *detection*. Notably, FacTool is limited to factual error detection, while CoVE *(F)* focuses solely on revision without providing response-level consistency labels. The comparison of baseline methods is provided in Table 4 and more details are available in Appendix A.1.

Detection Task. For RE-EX and baselines, we evaluate the ability to identify (or detect) factual errors at the response level using two distinct benchmarks. First, we utilize the FactPrompt dataset (Chern et al., 2023), which includes 50 real-world prompts and corresponding responses from GPT-3.5. Additionally, we employ the WiCE dataset (Kamoi et al., 2023), containing 358 claims with multiple pieces of information. The balanced accuracy and F1 score are chosen as evaluation metrics. To accommodate varying labeling criteria across baselines and datasets, we transformed the data into binary labels. Further details are available in Appendix A.2.

Revision Task. For RE-EX and baselines, we evaluate the ability to revise the response if it has factual errors. We test on 157 responses from GPT-3.5, each containing an average of 31.1 labeled fact units. These fact units consist of concise sentences, each conveying a single piece of information, and are annotated by humans in a paper proposing FActScore (Min et al., 2023). Let a response R_{initial} has n fact units, while n_f of them are false. Let n_{ft} be the number of fact units that are incorrectly stated in R_{initial}, while correctly written in R_{revised}. Similarly, let n_{tt} be the number of fact units that are correctly written in both R_{initial} and R_{revised}. We use two metrics, (1) *correction accuracy* $= \frac{n_{ft}}{n_f}$ and (2) *revision accuracy* $= \frac{n_{ft}+n_{tt}}{n}$; the correction accuracy measures the ratio of *corrected* fact units that are incorrectly stated in the initial response, and the revision accuracy measures the ratio of the correct fact units in the *revised* response. Here, n_{ft} and n_{tt} are measured by using NLI model introduced in (Liu et al., 2023a), to determine whether they *entail*, are *neutral*, or *contradict* the revised response. Further details are provided in Appendix B.

Dataset	Methods	BAcc. (↑)		F1 (↑)		Avg.Time (sec, ↓)		Avg.Token (K, ↓)	
		GPT-3.5	GPT-4	GPT-3.5	GPT-4	GPT-3.5	GPT-4	GPT-3.5	GPT-4
Fact Prompt	FacTool	66.5	68.0	52.6	57.1	20.0	26.2	3.1K	4.1K
	CoVE *(F+R)*	62.0	54.2	59.6	53.1	5.7	23.3	6.0K	5.9K
	RARR	63.0	71.0	57.0	68.0	59.9	119.7	3.2K	3.1K
	RE-EX *(one-step)*	**75.9**	84.2	73.9	83.3	4.8	**19.0**	**1.0K**	**1.0K**
	RE-EX *(two-step)*	74.7	**87.0**	**75.5**	**86.8**	**4.8**	19.1	1.0K	1.1K
WiCE	FacTool	50.8	55.3	22.2	35.8	6.5	25.3	2.7K	2.9K
	CoVE *(F+R)*	51.8	49.3	22.5	3.3	7.3	22.4	5.2K	5.0K
	RARR	**56.0**	53.0	**46.0**	43.0	59.1	112.6	2.8K	2.8K
	RE-EX *(one-step)*	48.5	50.7	33.1	41.8	**5.4**	17.8	**0.8K**	**0.8K**
	RE-EX *(two-step)*	52.8	55.8	39.2	47.6	5.6	**15.4**	0.9K	0.8K

Table 1: Experimental results on the task of detecting factual errors. Compared with other baselines, RE-EX has higher balanced accuracy (BAcc) and F1 score, by using less time and tokens.

Methods	Correction Acc. (↑)		Revision Acc. (↑)		Avg.Time (sec, ↓)		Avg.Token (K, ↓)	
	GPT-3.5	GPT-4	GPT-3.5	GPT-4	GPT-3.5	GPT-4	GPT-3.5	GPT-4
RARR	36.9	39.1	63.8	68.7	106.9	257.9	3.7	3.9
CoVE *(F)*	36.3	33.6	64.9	69.9	**8.7**	**36.1**	3.0	3.0
CoVE *(F+R)*	35.9	19.4	62.0	67.0	16.2	49.8	13.0	13.1
RE-EX *(one-step)*	42.8	46.6	70.3	**75.4**	11.4	48.6	**2.1**	**2.4**
RE-EX *(two-step)*	**52.2**	**50.2**	**71.5**	74.9	12.5	48.9	2.7	2.8

Table 2: Experimental result on the task of revising factual errors, tested on the dataset given in (Min et al., 2023). RE-EX enjoys the best accuracies with reduced time/tokens, compared with baselines.

3.2 EXPERIMENTAL RESULTS

RE-EX performs well with low cost. As in Table 1, RE-EX achieves the best scores (balanced accuracy and F1 score) on the detection of factual errors. In particular, the performance improvement for GPT-4 was more pronounced, with an F1 score of 86.8 compared to 53.1 for CoVE *(F+R)* in the FactPrompt dataset. This is a much larger margin than that observed in GPT-3.5, where RE-EX achieved an F1 of 75.5 against 52.6 for FacTool. As in Table 2, RE-EX far outperforms existing baselines in the revision task, for both GPT-3.5 and GPT-4 models. For example, the correction accuracy increases by more than 15% for GPT-3.5, and by more than 10% for GPT-4. Similarly, the revision accuracy increases by more than 5% for both GPT-3.5 and GPT-4. As in Tables 1 and 2, RE-EX requires much less time and tokens than baselines to achieve such high performance, *e.g.,* RE-EX uses 6x fewer tokens than CoVE *(F+R)*. Table 6 and 7 in Appendix compares the revised responses for different methods including RE-EX, CoVE and RARR. One can confirm that RE-EX successfully corrects the factual errors in the initial response, while CoVE and RARR cannot.

When and why does RE-EX perform well? Recall that our method has two versions: RE-EX *(two-step)* which uses separate prompts for explanation and revision, and RE-EX *(one-step)* which uses a single prompt for both. As shown in Tables 1 and 2, RE-EX *(two-step)* outperforms RE-EX *(one-step)* in most cases. The correction accuracy gap is around 10%p, suggesting that splitting the task into smaller steps is helpful, similar to the observation reported in Bubeck et al. (2023). Furthermore, RE-EX demonstrates more consistent and reliable performance compared to baseline methods. As shown in Table 4 in Appendix, baselines can exhibit significant variation and vulnerability based on few-shot examples or labeled data, as discussed in Nguyen & Wong (2023); Lu et al. (2022). In contrast, RE-EX enables a zero-shot setting with improved prompting techniques.

Ablation Study. Recall that RE-EX consist of 3 steps as shown in Fig. 2. As in Table 3, ablating each step significantly reduces the correction accuracy for GPT-3.5. This shows that both the evidence retrieval step and the factual error explanation step are necessary for successful revision. In addition, the results in Table 8 shows that utilizing external evidence outperformed relying solely on internal response. This result coincides with our qualitative comparison in Table 8 in Appendix, showing that relying solely on the internal knowledge of LLMs can lead to insufficient evidence since they often provide outdated information or the document that does not contain the data we want to retrieve.

Step 1	Step 2	Step 3	Correction Acc. (↑)		Revision Acc. (↑)	
Evidence	Explanation	Revision	GPT-3.5	GPT-4	GPT-3.5	GPT-4
-	-	✓	8.4	24.5	62.2	67.7
-	✓	✓	28.7 +20.3%p	26.7 +2.2%p	65.4 +2.2%p	70.9 +3.2%p
✓ · Internal	-	✓	<u>40.3</u> +31.9%p	31.8 +7.3%p	67.7 +5.5%p	66.7 -1.0%p
✓ · Internal	✓	✓	39.9 +31.5%p	35.5 +11.0%p	<u>68.8</u> +6.6%p	67.5 -0.2%p
✓ · External	-	✓	25.4 +17.0%p	<u>45.2</u> +20.7%p	67.5 +7.0%p	<u>74.7</u> +7.0%p
✓ · External	✓	✓	**52.2** +43.8%p	**50.2** +25.7%p	**71.5** +9.3%p	**74.9** +7.2%p

Table 3: Ablation Study of each stage of RE-EX, tested on the FactScore dataset used in Table 2. We tested the revision task on GPT-3.5 and GPT-4, where steps 1, 2, and 3 are illustrated in Fig. 2. Excluding step 1 or step 2 results in over a 20% degradation in correction accuracy for GPT-3.5. We also tested the effect of using external sources in step 1, by comparing with a variant of our method using internal source (i.e., retrieve from the response of LLMs to the query), similar to CoVE (Dhuliawala et al., 2023) shown in Table 4 in Appendix.

4 RELATED WORKS

Fact Verification. The task of fact verification, which aims to assess the factual accuracy of claims, has seen significant research interest due to its wide applications. Existing methodologies for fact verification have often focused on short-form datasets, primarily FEVER (Thorne et al., 2018) or VitaminC (Schuster et al., 2021), which often struggle with the length and complexity of LLM-generated texts. Consequently, researchers have begun exploring ways to detect factual errors in longer textual outputs (Fan et al., 2020; Lattimer et al., 2023; Soleimani et al., 2023). The research also explored post-editing methods that could provide revised sentences along with the output of factual error detection, in order to achieve more practical aims. Drawing inspiration from the recent success of LLMs by using the Chain-of-Thought (CoT) method (Wei et al., 2023), some methods have tried to revise factual error via an additional reasoning step (Dhuliawala et al., 2023; Zhao et al., 2023). Following this line of research, our work introduces an explanation step as an intermediate reasoning process, which does not depend on label or few-shot examples (see Table 4 in Appendix).

Tool Use. Language models inherently possess a limited knowledge base derived from their pre-trained data. This limitation has motivated the recent incorporation of various tools to augment these models, significantly enhancing their capabilities (Schick et al., 2023; Zhuang et al., 2023; Shen et al., 2023; Liang et al., 2023). While some methods utilize established knowledge repositories like Wikipedia (Semnani et al., 2023; Li et al., 2023; Yu et al., 2023), others leverage search APIs to access more current and broader information (Gou et al., 2023; Chern et al., 2023; Gao et al., 2023). Our method also employs the Google Search API to address the limitations of pre-existing knowledge databases and enhance its suitability for more practical scenarios.

Automated Evaluation. Various metrics like BLEU (Papineni et al., 2002) and ROUGE (Lin, 2004) have been used to assess machine-generated text quality. Recent methods have leveraged LLMs for automated text evaluation (Fu et al., 2023; Liu et al., 2023b), building on their impressive reasoning capabilities. Despite efforts to revise factual errors in the text, a standardized evaluation metric has been lacking. Consequently, alternative methods have been proposed to assess revised text quality, aiming to replace costly human evaluation. Approaches like FActScore (Min et al., 2023) and SAFE (Wei et al., 2024) focused on evaluating LLM-generated text factual accuracy, but still relied on external sources which is potentially inaccurate. In contrast, our work used human-annotated data from FActScore to directly measure correction accuracy and revision accuracy.

5 CONCLUSION

In this paper, we propose RE-EX which detects and revises the factual errors in LLM-generated texts. The key idea of RE-EX is revising after explaining the factual error, based on the evidences gathered by external sources. Experimental results on GPT-3.5 and GPT-4 for multiple datasets show that RE-EX has a higher accuracy in detection and revision of factual errors, at lower cost. The demonstrated efficacy of RE-EX underscores its potential to foster trust and credibility in LLM-generated content across various domains, paving the way for further advancements in the field of artificial intelligence and natural language processing.

LIMITATION

Limitations of Evaluation Metrics. While our evaluation metrics for revision task, *correction accuracy* and *revision accuracy*, are designed to evaluate the performance of RE-EX and baseline models in revising responses with factual errors, they have certain limitations. The *correction accuracy* measures the proportion of false fact units in the initial response (R_{initial}) that are correctly revised in the revised response (R_{revised}). However, this metric does not account for the quality of the corrections made. For instance, a response could be deemed 'corrected' if the false information is merely deleted or negated without being replaced with accurate information. Such revisions, while technically reducing the number of false fact units, is not enough for being the revised response informative. The *revision accuracy* evaluates the ratio of correct fact units in the revised response. However, due to the dataset provided in (Min et al., 2023) comprising a higher proportion of true fact units (approximately twice as many as false ones), this metric might inadvertently favor models that simply retain the true facts while inadequately addressing the false ones. These limitations suggest that while our metrics provide a quantitative measure of a model's revision capabilities, they might not fully capture the qualitative aspects of proper fact correction. Future research should consider developing more comprehensive metrics or methodologies that evaluate not only the factual accuracy but also the informativeness and the contextual completeness of model-generated revisions.

Limitations of Our Method. While our method shows promising results, particularly with long-form datasets such as FactPrompt (Min et al., 2023), we could not achieve satisfactory results in the context of short-form text verification (*e.g.,* FEVER (Thorne et al., 2018)). It turns out that decomposing short text and retrieving relevant information is a challenging task for our method. In these cases, including evidences during the explanation and revision phases frequently resulted in information overload, which caused the model to lose focus on correcting factual inaccuracies. For instance, the revised output might become verbose or contain contradictory statements. These observations underscore the importance of evidence retrieval and interpretation, and thus suggest directions for future work on filtering the evidences.

REFERENCES

Sébastien Bubeck, Varun Chandrasekaran, Ronen Eldan, Johannes Gehrke, Eric Horvitz, Ece Kamar, Peter Lee, Yin Tat Lee, Yuanzhi Li, Scott Lundberg, et al. Sparks of artificial general intelligence: Early experiments with gpt-4. arXiv preprint arXiv:2303.12712, 2023.

I-Chun Chern, Steffi Chern, Shiqi Chen, Weizhe Yuan, Kehua Feng, Chunting Zhou, Junxian He, Graham Neubig, and Pengfei Liu. Factool: Factuality detection in generative ai – a tool augmented framework for multi-task and multi-domain scenarios, 2023.

Aakanksha Chowdhery, Sharan Narang, Jacob Devlin, Maarten Bosma, Gaurav Mishra, Adam Roberts, Paul Barham, Hyung Won Chung, Charles Sutton, Sebastian Gehrmann, Parker Schuh, Kensen Shi, Sasha Tsvyashchenko, Joshua Maynez, Abhishek Rao, Parker Barnes, Yi Tay, Noam Shazeer, Vinodkumar Prabhakaran, Emily Reif, Nan Du, Ben Hutchinson, Reiner Pope, James Bradbury, Jacob Austin, Michael Isard, Guy Gur-Ari, Pengcheng Yin, Toju Duke, Anselm Levskaya, Sanjay Ghemawat, Sunipa Dev, Henryk Michalewski, Xavier Garcia, Vedant Misra, Kevin Robinson, Liam Fedus, Denny Zhou, Daphne Ippolito, David Luan, Hyeontaek Lim, Barret Zoph, Alexander Spiridonov, Ryan Sepassi, David Dohan, Shivani Agrawal, Mark Omernick, Andrew M. Dai, Thanumalayan Sankaranarayana Pillai, Marie Pellat, Aitor Lewkowycz, Erica Moreira, Rewon Child, Oleksandr Polozov, Katherine Lee, Zongwei Zhou, Xuezhi Wang, Brennan Saeta, Mark Diaz, Orhan Firat, Michele Catasta, Jason Wei, Kathy Meier-Hellstern, Douglas Eck, Jeff Dean, Slav Petrov, and Noah Fiedel. Palm: Scaling language modeling with pathways, 2022.

Yihe Deng, Weitong Zhang, Zixiang Chen, and Quanquan Gu. Rephrase and respond: Let large language models ask better questions for themselves, 2023.

Shehzaad Dhuliawala, Mojtaba Komeili, Jing Xu, Roberta Raileanu, Xian Li, Asli Celikyilmaz, and Jason Weston. Chain-of-verification reduces hallucination in large language models, 2023.

Angela Fan, Aleksandra Piktus, Fabio Petroni, Guillaume Wenzek, Marzieh Saeidi, Andreas Vlachos, Antoine Bordes, and Sebastian Riedel. Generating fact checking briefs. arXiv preprint arXiv:2011.05448, 2020.

Jinlan Fu, See-Kiong Ng, Zhengbao Jiang, and Pengfei Liu. Gptscore: Evaluate as you desire. arXiv preprint arXiv:2302.04166, 2023.

Luyu Gao, Zhuyun Dai, Panupong Pasupat, Anthony Chen, Arun Tejasvi Chaganty, Yicheng Fan, Vincent Y. Zhao, Ni Lao, Hongrae Lee, Da-Cheng Juan, and Kelvin Guu. Rarr: Researching and revising what language models say, using language models, 2023.

Zhibin Gou, Zhihong Shao, Yeyun Gong, Yelong Shen, Yujiu Yang, Nan Duan, and Weizhu Chen. Critic: Large language models can self-correct with tool-interactive critiquing. arXiv preprint arXiv:2305.11738, 2023.

Ziwei Ji, Nayeon Lee, Rita Frieske, Tiezheng Yu, Dan Su, Yan Xu, Etsuko Ishii, Ye Jin Bang, Andrea Madotto, and Pascale Fung. Survey of hallucination in natural language generation. ACM Computing Surveys, 55(12):1–38, 2023.

Ryo Kamoi, Tanya Goyal, Juan Diego Rodriguez, and Greg Durrett. Wice: Real-world entailment for claims in wikipedia, 2023.

Barrett Martin Lattimer, Patrick Chen, Xinyuan Zhang, and Yi Yang. Fast and accurate factual inconsistency detection over long documents. arXiv preprint arXiv:2310.13189, 2023.

Patrick Lewis, Ethan Perez, Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich Küttler, Mike Lewis, Wen-tau Yih, Tim Rocktäschel, Sebastian Riedel, and Douwe Kiela. Retrieval-augmented generation for knowledge-intensive nlp tasks. In H. Larochelle, M. Ranzato, R. Hadsell, M.F. Balcan, and H. Lin (eds.), Advances in Neural Information Processing Systems, volume 33, pp. 9459–9474. Curran Associates, Inc., 2020. URL https://proceedings.neurips.cc/paper_files/paper/2020/file/6b493230205f780e1bc26945df7481e5-Paper.pdf.

Miaoran Li, Baolin Peng, and Zhu Zhang. Self-checker: Plug-and-play modules for fact-checking with large language models, 2023.

Yaobo Liang, Chenfei Wu, Ting Song, Wenshan Wu, Yan Xia, Yu Liu, Yang Ou, Shuai Lu, Lei Ji, Shaoguang Mao, et al. Taskmatrix. ai: Completing tasks by connecting foundation models with millions of apis. arXiv preprint arXiv:2303.16434, 2023.

Chin-Yew Lin. Rouge: A package for automatic evaluation of summaries. In Text summarization branches out, pp. 74–81, 2004.

Hanmeng Liu, Ruoxi Ning, Zhiyang Teng, Jian Liu, Qiji Zhou, and Yue Zhang. Evaluating the logical reasoning ability of chatgpt and gpt-4, 2023a.

Yang Liu, Dan Iter, Yichong Xu, Shuohang Wang, Ruochen Xu, and Chenguang Zhu. Gpteval: Nlg evaluation using gpt-4 with better human alignment. arXiv preprint arXiv:2303.16634, 2023b.

Yao Lu, Max Bartolo, Alastair Moore, Sebastian Riedel, and Pontus Stenetorp. Fantastically ordered prompts and where to find them: Overcoming few-shot prompt order sensitivity, 2022.

Alex Mallen, Akari Asai, Victor Zhong, Rajarshi Das, Daniel Khashabi, and Hannaneh Hajishirzi. When not to trust language models: Investigating effectiveness of parametric and non-parametric memories, 2023.

Potsawee Manakul, Adian Liusie, and Mark J. F. Gales. Selfcheckgpt: Zero-resource black-box hallucination detection for generative large language models, 2023.

Sewon Min, Kalpesh Krishna, Xinxi Lyu, Mike Lewis, Wen tau Yih, Pang Wei Koh, Mohit Iyyer, Luke Zettlemoyer, and Hannaneh Hajishirzi. Factscore: Fine-grained atomic evaluation of factual precision in long form text generation, 2023.

Niels Mündler, Jingxuan He, Slobodan Jenko, and Martin Vechev. Self-contradictory hallucinations of large language models: Evaluation, detection and mitigation, 2023.

Tai Nguyen and Eric Wong. In-context example selection with influences, 2023.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. Bleu: a method for automatic evaluation of machine translation. In Proceedings of the 40th annual meeting of the Association for Computational Linguistics, pp. 311–318, 2002.

Ansh Radhakrishnan, Karina Nguyen, Anna Chen, Carol Chen, Carson Denison, Danny Hernandez, Esin Durmus, Evan Hubinger, Jackson Kernion, Kamilė Lukošiūtė, Newton Cheng, Nicholas Joseph, Nicholas Schiefer, Oliver Rausch, Sam McCandlish, Sheer El Showk, Tamera Lanham, Tim Maxwell, Venkatesa Chandrasekaran, Zac Hatfield-Dodds, Jared Kaplan, Jan Brauner, Samuel R. Bowman, and Ethan Perez. Question decomposition improves the faithfulness of model-generated reasoning, 2023.

Timo Schick, Jane Dwivedi-Yu, Roberto Dessì, Roberta Raileanu, Maria Lomeli, Luke Zettlemoyer, Nicola Cancedda, and Thomas Scialom. Toolformer: Language models can teach themselves to use tools. arXiv preprint arXiv:2302.04761, 2023.

Tal Schuster, Adam Fisch, and Regina Barzilay. Get your vitamin c! robust fact verification with contrastive evidence, 2021.

Sina Semnani, Violet Yao, Heidi Zhang, and Monica Lam. Wikichat: Stopping the hallucination of large language model chatbots by few-shot grounding on wikipedia. In Findings of the Association for Computational Linguistics: EMNLP 2023, pp. 2387–2413, 2023.

Yongliang Shen, Kaitao Song, Xu Tan, Dongsheng Li, Weiming Lu, and Yueting Zhuang. Hugginggpt: Solving ai tasks with chatgpt and its friends in huggingface. arXiv preprint arXiv:2303.17580, 2023.

Amir Soleimani, Christof Monz, and Marcel Worring. Nonfacts: Nonfactual summary generation for factuality evaluation in document summarization. In Findings of the Association for Computational Linguistics: ACL 2023, pp. 6405–6419, 2023.

James Thorne, Andreas Vlachos, Christos Christodoulopoulos, and Arpit Mittal. Fever: a large-scale dataset for fact extraction and verification. arXiv preprint arXiv:1803.05355, 2018.

S. M Towhidul Islam Tonmoy, S M Mehedi Zaman, Vinija Jain, Anku Rani, Vipula Rawte, Aman Chadha, and Amitava Das. A comprehensive survey of hallucination mitigation techniques in large language models, 2024.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Brian Ichter, Fei Xia, Ed Chi, Quoc Le, and Denny Zhou. Chain-of-thought prompting elicits reasoning in large language models, 2023.

Jerry Wei, Chengrun Yang, Xinying Song, Yifeng Lu, Nathan Hu, Jie Huang, Dustin Tran, Daiyi Peng, Ruibo Liu, Da Huang, Cosmo Du, and Quoc V. Le. Long-form factuality in large language models, 2024.

Wenhao Yu, Hongming Zhang, Xiaoman Pan, Kaixin Ma, Hongwei Wang, and Dong Yu. Chain-of-note: Enhancing robustness in retrieval-augmented language models. arXiv preprint arXiv:2311.09210, 2023.

Ruochen Zhao, Xingxuan Li, Shafiq Joty, Chengwei Qin, and Lidong Bing. Verify-and-edit: A knowledge-enhanced chain-of-thought framework, 2023.

Yuchen Zhuang, Yue Yu, Kuan Wang, Haotian Sun, and Chao Zhang. Toolqa: A dataset for llm question answering with external tools. arXiv preprint arXiv:2306.13304, 2023.

APPENDIX

A DETAILS

A.1 DETAILS ON THE BASELINE

Methods	Evidence	Query Generation		Detection		Revision	
	source	level	n-shot	level	n-shot	level	n-shot
FacTool	external	claim	3-shot	claim	3-shot	-	-
RARR	external	response	6-shot	query	6-shot	response	6-shot
CoVE *(F)*	internal	response	3-shot	-	-	response	3-shot
CoVE *(F+R)*	internal	response	3-shot	query	3-shot	response	3-shot
RE-EX	external	response	0-shot	response	0-shot	response	0-shot

Table 4: Comparison of baseline methods with our method RE-EX for evidence source, query generation, detection, and revision. The table shows the source of evidence (internal knowledge of language models or external search tools), the input/output level (claim, query, raw response, or response), and whether each step employs few-shot or zero-shot approaches across different methods.

As shown in Table 4, FacTool and RARR both rely on external search tools: Serper API [2] and Bing Search API [3], respectively, for evidence retrieval. In contrast, CoVE revises responses internally within the LLM without external searches. Official code implementations exist for FacTool and RARR. We implemented CoVE following the paper's description and the prompt provided in the appendix, due to the lack of official code. For standardization, all experiments were conducted using both GPT-3.5 ('gpt-3.5-turbo') and GPT-4 ('gpt-4') with a temperature setting of 0.

The details of the baselines and our evaluation methods for these baselines are as follows:

- CoVE (Factored) (Dhuliawala et al., 2023) first generates the verification questions and answers one by one, uses these verification question and answer pairs as evidence to revise the response. Since this method only focuses on revising the response, it was evaluated solely on the revision task.

- CoVE (Factor + Revise) (Dhuliawala et al., 2023), which is similar to CoVE(Factored) but adds a step before delivering the final revised response. This step involves iterating through each verification question and its answer, and cross-checking the consistency between the verification question(query) and the answer pair and the initial response, categorizing them into three labels: *'consistent'*, *'partially consistent'*, and *'inconsistent'*. If the consistency for even one of the verification questions and answers is labeled as *'partially consistent'* or *'inconsistent'*, the initial response is considered false; otherwise, it is considered true for the evaluation of the detection task.

- RARR (Gao et al., 2023) utilizes two models, the *agreement model* and the *edit model*, to check the agreement between the evidence and the initial response, then revises it only if disagreement is detected. Similar to CoVE(Factor + Revise), the output from the *agreement model* is used for the evaluation of the detection task. If the output for even one piece of evidence is *'disagree'*, the initial response is considered false; otherwise, it is considered true.

- FacTool (Chern et al., 2023), which categorizes the initial response as either true or false. Therefore, the response-level factuality label from the model is used directly. Since this method focuses only on detecting factual errors in the response, it was evaluated solely on the detection task.

---

[2] https://serper.dev [3] https://www.microsoft.com/en-us/bing/apis

A.2 Details on the Dataset

We pre-processed each dataset to meet the specific requirements of our experiments, which used response-level binary labels { true, false }. Table 5 provides descriptions of each dataset after pre-processing.

Dataset	level	# True	# False	# Total
Annotated Dataset from FActScore	fact-unit	3194	1692	4886
Annotated Dataset from FActScore	response	-	157	157
FactPrompt	response	23	27	50
WiCE (test)	content	111	247	358

Table 5: Dataset Descriptions

- Annotated dataset from FActScore (Min et al., 2023), contains three labels { *S*, *NS*, *IR* } for each fact unit, which is a short sentence conveying one piece of information. We excluded the *'IR (Irrelevant)'* responses to simplify the dataset to two primary labels. Consequently, we categorized *'S (Supported)'* as true, and *'NS (Not Supported)'* as false. Additionally, if even a single fact unit within a response was false, the corresponding response was considered false. Otherwise, it was considered true.

- WiCE (Kamoi et al., 2023) dataset, it contains three categories { *S*, *PS*, *NS* }. For our purposes, *'S (Supported)'* is considered true, while both *'PS (Partially Supported)'* and *'NS (Not Supported)'* are treated as False in our evaluation metrics.

- FactPrompt (Chern et al., 2023) dataset provides the response-level labels in two categories { True, False }, which were used directly in our experiment without any further modification.

B Revision Score & Correction Score

Here we explain how n_{ft} and n_{tt} given in Sec. 3.1 are measured by using NLI model introduced in (Liu et al., 2023a). Let p be a fact unit, and let R_{revised} be the revised response. Then, we check the output of NLI model, denoted by

$$q = \text{NLI}(p, R_{\text{revised}}) \in \{\text{entails}, \text{neutral}, \text{contradicts}\}.$$

Then, n_{ft} is the number of fact units p that satisfies two conditions: (1) the fact unit is incorrectly stated in the initial response R_{initial} and (2) the output of NLI model satisfies $q \in \{\text{neutral}, \text{contradicts}\}$. Similarly, n_{tt} is the number of fact units p that satisfies two conditions: (1) the fact unit is correctly stated in the initial response R_{initial} and (2) the output of NLI model satisfies $q \in \{\text{entails}\}$.

C   OUTPUT EXAMPLES

C.1   RESPONSE REVISION EXAMPLES

As shown in Table 6 and 7, RE-EX successfully revised the factual error by replacing it with correct information, whereas other methods either failed to catch the error or simply negated it.

	Initial Response
GPT-3.5	Lanny Flaherty is an American actor born on **December 18, 1949, in Pensacola, Florida**... Some of his notable film credits include **"King of New York"**, ...
	Revised Response
RARR	Lanny Flaherty is an American actor born on July 27, 1942, in Pontotoc, Mississippi ... Some of his notable film credits include "King of New York," ...
CoVE *(F)*	Lanny Flaherty is an American actor. He was not born on December 18, 1949. ... He has appeared in films such as "King of New York," ...
CoVE *(F+R)*	Lanny Flaherty is an American actor born on December 18, 1949, in Pensacola, Florida. ... He has appeared in films such as "King of New York," ...
RE-EX *(one-step)*	Lanny Flaherty is an American actor born on July 27, 1942, in Pontotoc, Mississippi. ... Some of his notable film credits include "Signs" (2002), ...
RE-EX *(two-step)*	Lanny Flaherty is an American actor born on July 27, 1942, in Pontotoc, Mississippi. ... Some of his notable film credits include "The Ballad of Little Jo" (1993), ...

Table 6: Example Output of Revised Response Methods powered by GPT-3.5. The **Bold** Text contains factual errors in the initial response. Blue text contains correctly revised information, while Red text contains incorrectly revised information or not revised.

	Initial Response
GPT-3.5	The most valuable NFT ever sold is a digital artwork called **'Everydays: The First 5000 Days'** by the artist Beeple, which sold for **$69 million** at a Christie's auction in March 2021
	Revised Response
RARR	The most valuable NFT ever sold is a digital artwork called 'Everydays: The First 5000 Days' by the artist Beeple, which sold for $69 million at a Christie's auction in March 2021.
CoVE *(F)*	The most valuable Non-fungible Token (NFT) is 'Everydays: The First 5000 Days' by Beeple, which was sold for $69 million at a Christie's auction in March 2021.
CoVE *(F+R)*	The most valuable Non-fungible Token (NFT) as of 2021 is CryptoPunk #7804, which was sold for 4200 Ether, equivalent to approximately $7.6 million at the time of sale. However, the value of NFTs can fluctuate and new sales may have occurred.
RE-EX *(one-step)*	The most valuable NFT ever sold is actually a digital artwork called 'The Merge' by the artist Pak, which was sold for $91.8 million on the NFT platform Nifty Gateway in December 2021.
RE-EX *(two-step)*	The most valuable NFT ever sold is 'The Merge' by the artist Pak. However, 'Everydays: The First 5000 Days' by the artist Beeple is one of the most valuable, having sold for $69.3 million at a Christie's auction in March 2021.

Table 7: Example Output of Revised Response Methods powered by GPT-4. The **Bold** Text contains factual errors in the initial response. Blue text contains correctly revised information, while Red text contains incorrectly revised information or not revised.

C.2 RETRIEVED EVIDENCE EXAMPLES

Table 8 shows the evidences retrieved in Step 1 of RE-EX, either using the external search or the internal retrieval method. One can confirm that external search API enriches the method by leveraging up-to-date and accurate information. Compared to the approaches solely reliant on the internal knowledge of LLMs, which may be outdated or inaccurate, integrating externally sourced data can help LLMs revise the initial response with rich and accurate information.

Sub-Question (Query)	Does Thomas Houghton currently play as a hooker for Hull FC in the Super League?
Sub-Answer (Result)	
Internal · GPT-3.5	I'm sorry, but I am unable to provide real-time information on specific players' current teams. I recommend checking the latest news or the official website of the rugby league team to find out which team Thomas Houghton currently plays for.
Internal · GPT4	I'm sorry, I cannot provide real-time or current information. It is recommended to check the official Hull FC website or the Super League website for the most up-to-date information on player rosters and positions.
External · Google Search	Danny Houghton (born 25 September 1988) is an English rugby league footballer who plays as a hooker for Hull F.C. in the Super League. He has played his entire career for his hometown club Hull FC. He is an England Knights international ... *(Source: Wikipedia)* A member of the Super League Dream Team in 2016, Houghton finished as Super League's top tackler in 2011, 2013, 2015, 2016, 2017, 2019 and 2022 whilst featuring in the top three in 2012, 2014 and 2018... In the 2022 season, Houghton surpassed Brian Hancock to become Hull FC's leading living appearance maker ... *(Source: FC Official Website)*

Table 8: Example of Sub-answer which is the result of a search query, comparing 1) instructing LLMs to respond to the query using their internal knowledge, which is denoted as **Internal**, and 2) using organic results, which means top search results from the Search API, denoted as **External**. GPT-3.5 and GPT-4 cannot provide the information correctly; however, the external search gets the top 2 organic results: one is from Wikipedia, and the other is from an FC website.

D   COMPARING QUERY GENERATION STEPS

In this section, we compare the query generation step for baselines (FacTool (Chern et al., 2023), CoVe (Dhuliawala et al., 2023) and RARR (Gao et al., 2023)) and RE-EX. FacTool uses a two-step process for query generation, beginning with the claim extraction and then generating individual queries for each extracted claim. CoVE uses a plan verification step that directly generates verification questions. The RARR method employs a comprehensive question generation (CQGen) process, which first samples three questions from the CQGen model and then combines these results to form a comprehensive set of queries. Lastly, RE-EX, utilizes a question-decomposition based approach for the sub-question generation, breaking down the text into sub-questions.

Methods	Time (sec, ↓)		Token (K, ↓)		# Queries	
	GPT-3.5	GPT-4	GPT-3.5	GPT-4	GPT-3.5	GPT-4
FacTool	2.5	11.0	1.9	2.0	8.3	8.9
CoVE	**1.0**	**2.6**	<u>0.6</u>	<u>0.6</u>	2.6	2.4
RARR	3.9	11.5	1.8	1.5	5.5	5.6
RE-EX	<u>1.6</u>	<u>8.9</u>	**0.2**	**0.2**	4.9	5.1

Table 9: Experimental results for the query generation step using the FactPrompt Dataset, comparing the average number of tokens, time, and the average number of generated queries for each method. Full step results are provided in Table 1.

As shown in Table 9, RE-EX requires the fewest tokens compared to the other baselines. In terms of the number of queries and time, CoVE requires and generates the least, followed by RE-EX. As illustrated in Table 10, CoVE indeed generates fewer queries, which leads to poorer performance, as shown in Table 1. In the case of FacTool and RARR, as seen in Table 1 and Table 9, RE-EX uses fewer tokens and time while achieving better performance than both. Since FacTool and RARR do not generate queries in a single step, duplicated queries, as shown in Table 10, are often observed, leading to inefficiency.

	Prompt
User	Who is the CEO of Twitter?

	Initial Response
GPT-3.5	The CEO of Twitter at the time of writing this answer is Jack Dorsey. He co-founded Twitter in 2006 and served as the CEO until 2008, and returned as CEO in 2015. Dorsey is also the CEO and co-founder of Square, a financial services and mobile payment company. He has been recognized as one of Time magazine's 100 most influential people in the world and has also been awarded the Innovator of the Year Award by Wall Street Journal.

	Generated Queries
FacTool	1. Who is the current CEO of Twitter? 2. Jack Dorsey 3. Who co-founded Twitter in 2006? 4. Jack Dorsey 5. When did Jack Dorsey serve as the CEO of Twitter? 6. Jack Dorsey CEO Twitter 7. When did Jack Dorsey return as CEO of Twitter? 8. Jack Dorsey CEO Twitter 2015 9. Who is the CEO of Square? 10. Who is the co-founder of Square? 11. What is Square? 12. Square financial services 13. Has Jack Dorsey been recognized as one of Time magazine's 100 most influential people in the world? 14. Time magazine 100 most influential people Jack Dorsey 15. Has Jack Dorsey been awarded the Innovator of the Year Award by Wall Street Journal? 16. Jack Dorsey Innovator of the Year Award Wall Street Journal
RARR	1. Who is the CEO of Twitter? 2. When was Twitter founded? 3. When did Jack Dorsey co-found Twitter? 4. What other company is Jack Dorsey the CEO of? 5. What other company is Jack Dorsey the CEO and co-founder of? 6. What other company does Jack Dorsey co-founded? 7. Has Jack Dorsey been recognized by Time magazine? 8. Has Jack Dorsey been recognized by Time magazine and Wall Street Journal? 9. Has Jack Dorsey been awarded the Innovator of the Year Award by Wall Street Journal?
CoVE	1. When did Jack Dorsey co-found Twitter? 2. When did Jack Dorsey first become CEO of Twitter?
RE-EX	1. Who is the CEO of Twitter? 2. When did Jack Dorsey co-found Twitter? 3. How long did Jack Dorsey serve as the CEO of Twitter in his first term? 4. When did Jack Dorsey return as CEO of Twitter? 5. What other company is Jack Dorsey the CEO and co-founder of? 6. What type of company is Square? 7. Was Jack Dorsey been recognized by Time magazine? If so, for what? 8. What award has Jack Dorsey received from the Wall Street Journal?

Table 10: Example of query generation output powered by GPT-3.5. Here, underlined text contains duplicated queries.

E   PROMPTS

Your task is to decompose the text into simple sub-questions for checking factual accuracy of the text. Make sure to clear up any references.

Topic: `[prompt]`
Text: `[response]`

Sub-Questions:

Table 11: Prompt template used for Sub-Question Generation.

You will receive an initial response along with a prompt. Your goal is to refine and enhance this response, ensuring its factual accuracy. Check for any factually inaccurate information in the initial response.

Use the provided sub-questions and corresponding answers as key resources in this process. Sub-questions and Answers :
```
[ sub-question 1 & sub-answer 1 ]
[ sub-question 2 & sub-answer 2 ]
[ sub-question 3 & sub-answer 3 ]
 ...
```

Prompt: `[prompt]`
Initial Response: `[response]`

Please explain the factual errors in the initial response, and revise it accordingly.
If there are no factual errors, respond with "None".
If there are factual errors, explain each factual error.

Factual Errors:
Revised Response:

Table 12: Prompt template used for RE-EX *(one-step)*: Step 2 & 3. Factual Error Explanation & Revision.

You will receive an initial response along with a prompt. Your goal is to refine and enhance this response, ensuring its factual accuracy. Check for any factually inaccurate information in the initial response.

Use the provided sub-questions and corresponding answers as key resources in this process. Sub-questions and Answers :
```
[ sub-question 1 & sub-answer 1 ]
[ sub-question 2 & sub-answer 2 ]
[ sub-question 3 & sub-answer 3 ]
 ...
```

Prompt: `[prompt]`
Initial Response: `[response]`

Please explain the factual errors in the initial response.
If there are no factual errors, respond with "None".
If there are factual errors, explain each factual error.

Factual Errors:

Table 13: Prompt template used for RE-EX *(two-step)*: Step 2. Factual Error Explanation.

You will receive an initial response along with a prompt. Your goal is to refine and enhance this response, ensuring its factual accuracy. Check for any factually inaccurate information in the initial response. You will receive a list of factual errors in the initial response from the previous step. Use this explanation of each factual error as a key resource in this process :

Factual Errors:
```
[ explanation-of-factual-error 1 ]
[ explanation-of-factual-error 2 ]
[ explanation-of-factual-error 3 ]
...
```

Prompt: `[prompt]`
Initial Response: `[response]`
Revised Response:

Table 14: Prompt template used for RE-EX *(two-step)* Step 3. Revision.