Extractive Summarization via ChatGPT for Faithful Summary Generation

Haopeng Zhang **Xiao Liu** **Jiawei Zhang**
IFM Lab, University of California, Davis

Abstract

Extractive summarization is a crucial task in natural language processing that aims to condense long documents into shorter versions by directly extracting sentences. The recent introduction of ChatGPT has attracted significant interest in the NLP community due to its remarkable performance on a wide range of downstream tasks. However, concerns regarding factuality and faithfulness have hindered its practical applications for summarization systems. This paper first presents a thorough evaluation of ChatGPT's performance on extractive summarization and compares it with traditional fine-tuning methods on various benchmark datasets. Our experimental analysis reveals that ChatGPT's extractive summarization performance is still inferior to existing supervised systems in terms of ROUGE scores. In addition, we explore the effectiveness of in-context learning and chain-of-thought reasoning for enhancing its performance. Furthermore, we find that applying an extract-then-generate pipeline with ChatGPT yields significant performance improvements over abstractive baselines in terms of summary faithfulness. These observations highlight potential directions for enhancing ChatGPT's capabilities for faithful text summarization tasks using two-stage approaches.

1 Introduction

Document summarization aims to compress text material while retaining its most salient information. With the increasing amount of publicly available text data, automatic summarization approaches have become increasingly important. These approaches can be broadly classified into two categories: abstractive and extractive summarization. While abstractive methods (Nallapati et al., 2016; Gupta and Gupta, 2019; Bae et al., 2019; Li et al., 2020) have the advantage of producing flexible and less redundant summaries, they often struggle with generating ungrammatical or even nonfactual contents (Kryściński et al., 2019; Zhang et al., 2022b). Unlike abstractive methods, extractive summarization directly selects sentences from the source document to form the summary, resulting in summaries that are grammatically correct and faithful to the original text.

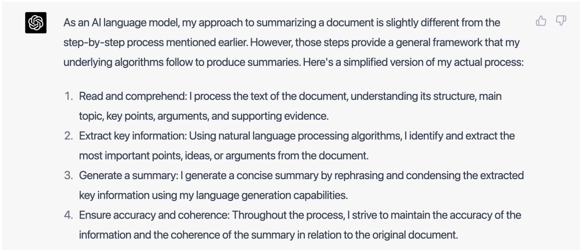

Figure 1: ChatGPT's response to the question 'How do you summarize a document?'

The growing interest in applying advanced large language models such as ChatGPT [1] for text summarization tasks has sparked significant attention. A recent study by (Goyal et al., 2022) compared GPT-3 with traditional fine-tuning methods and found that, despite lower Rouge scores, human annotators preferred the GPT-3 generated text. Another study by (Zhang et al., 2023) conducted a comprehensive analysis of large language models for news summarization and found that the generated summaries were comparable to those produced by humans. However, existing research (Yang et al., 2023; Luo et al., 2023) has only focused on abstractive summary approaches, and the performance of ChatGPT for extractive summarization remains an open question. Moreover, the hallucination problem has dramatically hindered the practical use of abstractive summarization systems, highlighting the need to explore extractive summarization with LLMs and its potential benefits for faithful summaries. As shown in Figure 1, even ChatGPT itself favors the extract-then-generate pipeline for summarization.

[1] https://chat.openai.com/chat

In this study, we comprehensively evaluate ChatGPT's performance on extractive summarization and investigate the effectiveness of in-context learning and chain-of-thought explanation approaches. Our experimental analysis demonstrates that there is still a performance gap between ChatGPT's summarization capabilities and traditional supervised fine-tuning methods in terms of ROUGE scores. Additionally, we observe that using an extract-then-generate pipeline with ChatGPT yields large performance improvements over abstractive baselines in terms of summary faithfulness. The findings of this research provide valuable insights into the potential of ChatGPT for text summarization tasks and highlight the need for innovative approaches in LLM-based two-stage summarization systems.

We summarize the contributions of this paper as follows:

- This study represents the first attempt to extend the application of ChatGPT to extractive summarization and evaluate its performance.
- We investigate the effectiveness of in-context learning and chain-of-thought reasoning approaches for extractive summarization using ChatGPT.
- We further extend the extraction step to abstractive summarization and find that the extract-then-generate framework could improve the generated summary faithfulness by a large margin compared to abstractive-only baselines without hurting summary qualities.

2 Related Work

2.1 Extractive Summarization

Significant advancements in deep neural networks have played a pivotal role in accelerating the progress of extractive summarization systems. Most works formulate the task as a sequence classification problem and use sequential neural models with diverse encoders such as recurrent neural networks (Cheng and Lapata, 2016; Nallapati et al., 2016) and pre-trained language models (Egonmwan and Chali, 2019; Liu and Lapata, 2019). Another group of works formulated extractive summarization as a node classification problem and applied graph neural networks to model inter-sentence dependencies (Xu et al., 2019; Wang et al., 2020; Zhang et al., 2020, 2022a). Recent works also propose summary-level formulations such as text matching (Zhong et al., 2020; An et al., 2022) and reinforcement learning (Narayan et al., 2018b; Bae et al., 2019) have been proposed. However, most previous works fall into supervised training or fine-tuning categories, and extractive summarization with large language models like ChatGPT has not been much explored.

2.2 Large Language Models

In recent years, there has been a growing interest in the field of natural language processing towards large language models such as GPT-3 (Brown et al., 2020) and ChatGPT, which are trained on vast amounts of text data and have achieved impressive results on a variety of NLP tasks.

In the context of text summarization, several studies have explored the use of large language models. Goyal et al. (2022) compared the performance of GPT-3-generated summaries with traditional fine-tuning methods and found that while the former obtained slightly lower Rouge scores, human evaluators preferred them. Likewise, Zhang et al. (2023) reported that large language model-generated summaries were on par with human-written summaries in the news domain. In addition, Yang et al. (2023) explored the limits of ChatGPT on query-based summarization other than generic summarization. Luo et al. (2023) explored the use of ChatGPT as a factual inconsistency evaluator for abstractive text summarization. While most of the existing research has focused on abstractive summarization, this work aims to investigate the applicability of ChatGPT to extractive summarization and examine whether extractive methods could enhance abstractive summarization faithfulness.

3 Methods

3.1 Task Formulation

Extractive summarization systems form a summary by identifying and concatenating the most salient sentences from a given document. These approaches have gained widespread traction in various real-world applications owing to their ability to produce accurate and trustworthy summaries devoid of grammatical inconsistencies.

Formally, given a document consisting of n sentences, the goal of an extractive summarization system is to produce a summary S comprising of $m(m \ll n)$ sentences, by directly extracting relevant sentences from the source document. Most existing work formulates it as a sequence labeling problem, where each sentence is assigned a binary label $0, 1$ to indicate whether it should be included in the summary S. However, given that extractive ground-truth labels (ORACLE) are unavailable for human-written gold summaries, a common strategy is to employ a greedy algorithm that generates an ORACLE by selecting multiple sentences that maximize the ROUGE score relative to the gold summary following (Nallapati et al., 2017). In this study, we first empirically evaluate ChatGPT's performance on a range of extractive summarization benchmarks.

3.2 In-context Learning and Explanation

Recent studies have shown that large language models have strong few-shot performance on various downstream tasks, known as in-context learning (ICL) (Brown et al., 2020). Let q be the input query, the standard ICL prompts a language model, M, with a set of exemplar input-output pairs, $\{(q_1, a_1)...(q_m, a_m)\}$, and predict an answer $\hat{a}$ for the query by:

$$\hat{a} = \arg\max_a p_M(a \mid q, \{(q_1, a_1)...(q_m, a_m)\}). \tag{1}$$

Besides simple input-output pairs, previous works also show that including explanations and chain-of-thought reasoning in prompts (Nye et al., 2021; Wei et al., 2022; Ye et al., 2022) also benefits language models, represented as:

$$\hat{a} = \arg\max_a \sum_e p_M(a, e \mid q, C), \tag{2}$$

where $C = \{(q_1, e_1, a_1)...(q_m, e_m, a_m)\}$ is the set of input-explanation-output triplets in prompts. This study also investigates the impact of in-context learning on extractive summarization, with and without the provision of explanations.

3.3 Extract-abstract Summarization

It is not new to use extractive summaries to guide abstractive summary generations (Dou et al., 2020). We can use language models in a two-step manner: extract salient sentences to form extractive summaries (S) first, and then ask the LM to generate summaries guided by the extractive summaries, represented as:

$$\hat{a} = \arg\max_a p_M(a \mid q, S). \tag{3}$$

We explore the extract-then-generate pipeline in this study, aiming to alleviate the hallucination problem in summary generation.

3.4 Prompts

Here we list prompts used in our experiments for extracted and generated summaries in Table 1. Note that according to OpenAI's document, the model could receive two categories of prompts: system prompt and user prompt, where the system prompt functions as the global instruction to initialize the model and the user prompt as the question proposed by users. In our experiment, we leverage both prompts to guide the model and select the best prompts on a dev set of 50 examples.

Setting	Prompt
Extractive	**System**: You are an extractive summarizer that follows the output pattern. **User**: Please extract sentences as the summary. The summary should contain **m** sentences. Document: [*Test Document*] [*Format Instruction*].
Abstractive	**System**: You are an abstractive summarize that follows the output pattern. **User**: Please write a summary for the document. Document: [*Test Document*] [*Format Instruction*]
In-context	**System**: You are an extractive summarizer that follows the output pattern. **User**: The following examples are successful extractive summarization instances: [*n Document-Summary Pairs*]. Please summarize the following document. Document: [*Test Document*]. The summary should contain m sentences. [*Format Instruction*].
Explanation	**System**: You are an extractive summarizer that follows the output pattern. **User**: The following examples are successful extractive summarization instances: [*n Document-Summary-Reason Triads*]. Please summarize the following document and give the reason. Document: [*Test Document*]. The summary should contain m sentences. [*Format Instruction*].
Extract-abstract	**System**: You are an abstractive summarizer that follows the output pattern. **User**: Please revise the extracted summary based on the document. The revised summary should include the information in the extracted summary. Document: [*Test Docuemnt*] Extractive Summary: [*Extractive Summary*] [*Format Instruction*].

Table 1: Prompts used for both extractive and abstractive summarization. m is the number of extracted sentences defined in Table 3. Document-summary pairs and document-summary-reason triads are the input contexts. n is the number of context instances.

Dataset	Ext-SOTA			Ext-GPT			Abs-SOTA			Abs-GPT		
	R1	**R2**	**RL**	**R1**	**R2**	**RL**	**R1**	**R2**	**RL**	**R1**	**R2**	**RL**
Reddit	25.09	6.17	20.13	21.40	4.69	14.62	32.03	11.13	25.51	24.64	5.86	18.54
XSum	24.86	4.66	18.41	19.85	2.96	13.29	48.12	24.95	40.00	26.30	7.53	20.21
PubMed	41.21	14.91	36.75	36.15	11.94	25.30	-	-	-	36.05	12.11	28.46
CNN/DM	44.41	20.86	40.55	39.25	17.09	25.64	47.16	22.55	43.87	38.48	14.46	28.39

Table 2: The experimental results for extractive and abstractive methods are presented on benchmark datasets. 'Ext' denotes extractive outcomes, while 'Abs' indicates abstractive outcomes. 'GPT' is used to represent the evaluation results of ChatGPT responses, whereas 'SOTA' signifies the current state-of-the-art performance. For state-of-the-art models, MatchSum (Zhong et al., 2020) is used as the extractive benchmark, and SummaReranker(Ravaut et al., 2022) is used as the abstractive benchmark. The scores for state-of-the-art models are evaluated using the full test dataset, while the ChatGPT results are assessed based on a random selection of 1,000 samples from the test dataset.

Dataset	Domain	Doc #words	Sum #words	#Ext
Reddit	Social Media	482.2	28.0	2
XSum	News	430.2	23.3	2
PubMed	Paper	444	209.5	6
CNN/DM	News	766.1	58.2	3

Table 3: Statistics of the experimental datasets. Doc # words and Sum # words refer to the average word number in the source document and summary. # Ext refers to the number of sentences to extract.

4 Experiments and Analysis

4.1 Experiment Settings

Datasets: We chose four publicly available benchmark datasets as listed in Table 3, ensuring that they are consistent with previous fine-tuning approaches. CNN/DailyMail (Hermann et al., 2015) is the most widely-adopted summarization dataset that contains news articles and corresponding human-written news highlights as summaries. We use the non-anonymized version in this work and follow the common training, validation, and testing splits (287,084/13,367/11,489). XSum (Narayan et al., 2018a) is a one-sentence news summarization dataset with all summaries professionally written by the original authors of the documents. We follow the common training, validation, and testing splits (204,045/11,332/11,334). PubMed (Cohan et al., 2018) is a scientific paper summarization dataset of long documents. We follow the setting in (Zhong et al., 2020) and use the introduction section as the article and the abstract section as the summary. The training/validation/testing split is (83,233/4,946/5,025).

Evaluation: We conducted an evaluation of ChatGPT's performance utilizing Rouge scores as our evaluation metric. We selected the best prompts on a dev set of 50 examples and randomly sampled 1000 examples from each test set for evaluation. Previous research efforts (Goyal et al., 2022; Zhang et al., 2023) have also been limited in their testing of GPT-3 on a small number of instances. We plan to conduct large-scale human evaluation later to better understand the performance of ChatGPT.

4.2 Experiments Results

The overall results are shown in Table 2. It is observed that ChatGPT generally achieves lower ROUGE scores in comparison to traditional fine-tuning methods for all datasets under both extractive and abstractive settings. The results are consistent with the previous conclusion in (Goyal et al., 2022; Zhang et al., 2023). It is also observed that Ext-GPT outperforms Abs-GPT in two extractive datasets CNN/DM and PubMed while performing worse in the other two abstractive datasets. We argue the results are due to the bias within the reference summaries of the dataset and the limit of ROUGE scores. We plan to conduct human evaluations to examine the performance further. Nonetheless, we note that despite being a large language model primarily designed for generation tasks, ChatGPT achieves impressive results in extractive summarization, which requires comprehension of the problem formulation and semantic meanings of sentences. The decoder-only structure of ChatGPT doesn't degrade its comprehension ability compared to encoder-decoder models like BERT.

In addition to the zero-shot setting, we investigate the effects of in-context learning for ChatGPT on extractive summarization, as shown in Table 4. The results indicate that in-context learning improves model performance in the extractive dataset CNN/DM, but is less effective in the abstractive dataset XSum. This is likely because more ex-

# Context	CNN/DM			XSum		
	R1	**R2**	**RL**	**R1**	**R2**	**RL**
0	39.25 ± 0.23	15.36 ± 1.10	25.90 ± 0.97	19.85 ± 2.59	2.96 ± 2.59	13.29 ± 1.30
1	40.62 ± 0.70	17.00 ± 1.06	26.44 ± 0.84	15.33 ± 0.50	2.48 ± 0.19	11.48 ± 0.13
1w/R	38.83 ± 0.91	14.94 ± 2.53	25.36 ± 1.82	17.86 ± 1.73	3.29 ± 0.85	12.55 ± 1.29
2	40.91 ± 0.69	15.68 ± 0.61	26.13 ± 0.83	18.61 ± 0.39	4.42 ± 0.97	14.06 ± 2.01
2w/R	41.70 ± 0.70	15.95 ± 0.92	26.98 ± 1.33	17.95 ± 3.03	4.11 ± 1.01	13.46 ± 1.76
3	**42.38 ± 0.13**	17.27 ± 0.23	**28.41 ± 0.31**	17.49 ± 1.87	3.86 ± 1.55	12.94 ± 2.16
3w/R	42.26 ± 1.38	17.02 ± 1.60	27.42 ± 1.62	**20.37 ± 1.61**	**4.78 ± 0.44**	**14.21 ± 1.07**
4	42.26 ± 0.50	**17.41 ± 0.83**	27.96 ± 0.83	16.68 ± 1.56	3.72 ± 0.20	12.12 ± 1.19
4w/R	41.23 ± 0.93	17.08 ± 0.38	28.25 ± 0.93	18.17 ± 0.28	4.05 ± 0.38	12.74 ± 0.94
5	40.71 ± 1.92	16.96 ± 0.91	27.42 ± 1.26	17.43 ± 1.08	3.53 ± 0.96	12.33 ± 0.51
5w/R	40.18 ± 0.83	15.15 ± 1.44	25.98 ± 1.91	19.55 ± 0.64	4.29 ± 0.46	13.13 ± 0.68

Table 4: In-context learning experimental results on CNN/DM and XSum datasets. For each dataset, we randomly sampled 50 data from the test set. In each section, w/R means we provide human written reasons for each context document. For the test document, we also ask the system to generate the reason why it choose selected sentences.

Dataset	R1	R2	RL	FactCC
Reddit-A	24.64	5.86	18.54	9.46
Reddit-EA	24.45(-0.19)	5.64(-0.22)	18.26(-0.28)	**60.4**
Reddit-OA	**26.03(+1.39)**	**6.61(+0.75)**	**19.37(+0.83)**	59.75
XSum-A	26.30	7.53	20.21	5.42
XSum-EA	24.31(-1.99)	5.75(-1.78)	18.55(-1.66)	**55.73**
XSum-OA	**28.50(+2.20)**	**8.29(+0.76)**	**21.10(+0.89)**	55.03
PubMed-A	36.05	12.11	28.46	8.37
PubMed-EA	**36.15(+0.10)**	10.12(-1.99)	26.50(-1.96)	26.38
PubMed-OA	33.44(-2.61)	**11.88(-0.23)**	**26.51(-1.95)**	**27.35**
CNN/DM-A	28.38	14.46	28.39	6.35
CNN/DM-EA	39.60(+1.12)	15.21(+0.75)	29.16(+0.77)	51.65
CNN/DM-OA	**44.60(+6.12)**	**19.42(+4.96)**	**33.32(+4.93)**	**53.67**

Table 5: Abstractive refinement of extractive summary. -A is the baseline abstractive result shown in Table 2, -EA is the result of extractive then abstractive setting, and -OA represents we input the ORACLE summary sentences to generate a refined abstractive summary. The number in the parenthesis is the absolute difference between the corresponding baseline abstractive result in Table 2.

tractive datasets have clearer patterns of ORACLE sentences, and ChatGPT can benefit from learning such patterns with a few examples. While for the more abstractive dataset XSum, since the ORACLE is only one sentence, providing a few examples of ORACLE creates more confusion for the model, resulting in degraded performance. However, with chain-of-thought human-written explanations, ChatGPT can better understand the pattern, which is why in-context learning with explanations works much better for the XSum dataset compared to examples without explanation. In summary, in-context is generally effective and beneficial for the extractive summarization task, and surprisingly, ChatGPT can have comparable extractive scores to supervised fine-tuned models with very few examples.

We conducted further experiments to examine the effectiveness of the extract-generate framework for ChatGPT, as presented in Table 5. The results demonstrate that the performance of ChatGPT generally improves largely in terms of ROUGE scores when grounded with the ORACLE summaries. However, the performance of the extract-generate framework relies heavily on the extractive performance when grounded with its own extractive summaries. Specifically, ChatGPT shows a significant improvement in ROUGE scores for the CNN/DM dataset but experiences a decrease in performance for the other three datasets.

To investigate the faithfulness of the generated summaries, we employed FactCC (Kryściński et al., 2019) as an evaluation metric. The results in the last column in Table 5 show that the extract-then-generate framework effectively alleviates the hallucination problem of abstractive summaries. Similar to previous study(Goyal et al., 2022), the generated summaries by ChatGPT get poor FactCC scores (less than 10 percent). However, if we let ChatGPT guide summary generation with its extractive summaries, the score improves by a very large margin (around 50 percent). We plan to conduct further experiments with other metrics and human evaluation to investigate the effectiveness of the extract-generate framework in addressing the faithfulness issue of abstractive summarization.

4.3 Analysis

Position bias is a common phenomenon in extractive summarization, especially in the news domain. We show the position bias analysis in Figure 2, which indicates that the distribution of the ChatGPT summary sentences is skewed towards a higher position bias than the ORACLE sentences.

We further investigate the influence of document length on the summarization performance, as presented in Figure 3. Our findings suggest that ChatGPT maintains consistent performance across documents of different lengths, indicating the model's robustness in the context of extractive summarization. Moving forward, we intend to conduct further comprehensive analyses to gain a deeper understanding of ChatGPT's behavior.

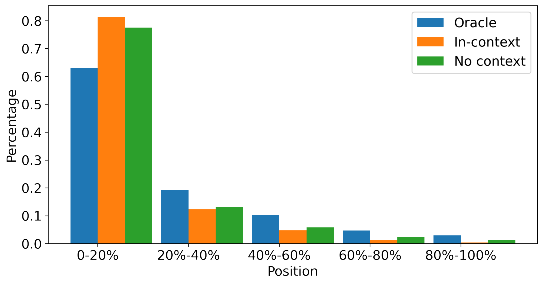

Figure 2: Position distribution of extracted sentences.

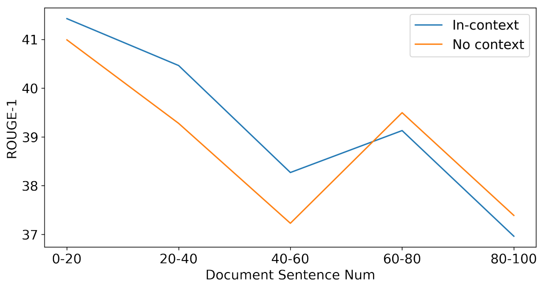

Figure 3: ROUGE-1 Score distribution over document of different lengths.

5 Conclusion

This paper presents a thorough evaluation of ChatGPT's performance on extractive summarization across a range of datasets and the effectiveness of in-context learning. The results indicate ChatGPT's strong potential for the task and the possibility of generating factual summaries using the extract-generate framework. However, the limitations of the ROUGE score as a reliable evaluation metric for text summarization tasks are also highlighted. To obtain a more comprehensive assessment of ChatGPT's performance, we plan to conduct human evaluations of the generated text. Overall, this study suggests that ChatGPT is a powerful tool for text summarization, and we hope the insights gained from this work can guide future research in this area.

References

Chenxin An, Ming Zhong, Zhiyong Wu, Qin Zhu, Xuanjing Huang, and Xipeng Qiu. 2022. Colo: A contrastive learning based re-ranking framework for one-stage summarization. *arXiv preprint arXiv:2209.14569*.

Sanghwan Bae, Taeuk Kim, Jihoon Kim, and Sang-goo Lee. 2019. Summary level training of sentence rewriting for abstractive summarization. *arXiv preprint arXiv:1909.08752*.

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. 2020. Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901.

Jianpeng Cheng and Mirella Lapata. 2016. Neural summarization by extracting sentences and words. *arXiv preprint arXiv:1603.07252*.

Arman Cohan, Franck Dernoncourt, Doo Soon Kim, Trung Bui, Seokhwan Kim, Walter Chang, and Nazli Goharian. 2018. A discourse-aware attention model for abstractive summarization of long documents. *arXiv preprint arXiv:1804.05685*.

Zi-Yi Dou, Pengfei Liu, Hiroaki Hayashi, Zhengbao Jiang, and Graham Neubig. 2020. Gsum: A general framework for guided neural abstractive summarization. *arXiv preprint arXiv:2010.08014*.

Elozino Egonmwan and Yllias Chali. 2019. Transformer-based model for single documents neural summarization. In *Proceedings of the 3rd Workshop on Neural Generation and Translation*, pages 70–79.

Tanya Goyal, Junyi Jessy Li, and Greg Durrett. 2022. News summarization and evaluation in the era of gpt-3. *arXiv preprint arXiv:2209.12356*.

Som Gupta and Sanjai Kumar Gupta. 2019. Abstractive summarization: An overview of the state of the art. *Expert Systems with Applications*, 121:49–65.

Karl Moritz Hermann, Tomas Kocisky, Edward Grefenstette, Lasse Espeholt, Will Kay, Mustafa Suleyman, and Phil Blunsom. 2015. Teaching machines to read and comprehend. In *Advances in neural information processing systems*, pages 1693–1701.

Wojciech Kryściński, Bryan McCann, Caiming Xiong, and Richard Socher. 2019. Evaluating the factual consistency of abstractive text summarization. *arXiv preprint arXiv:1910.12840*.

Haoran Li, Junnan Zhu, Jiajun Zhang, Chengqing Zong, and Xiaodong He. 2020. Keywords-guided abstractive sentence summarization. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 8196–8203.

Yang Liu and Mirella Lapata. 2019. Text summarization with pretrained encoders. *arXiv preprint arXiv:1908.08345*.

Zheheng Luo, Qianqian Xie, and Sophia Ananiadou. 2023. Chatgpt as a factual inconsistency evaluator for abstractive text summarization. *arXiv preprint arXiv:2303.15621*.

Ramesh Nallapati, Feifei Zhai, and Bowen Zhou. 2017. Summarunner: A recurrent neural network based sequence model for extractive summarization of documents. In *Thirty-first AAAI conference on artificial intelligence*.

Ramesh Nallapati, Bowen Zhou, Caglar Gulcehre, Bing Xiang, et al. 2016. Abstractive text summarization using sequence-to-sequence rnns and beyond. *arXiv preprint arXiv:1602.06023*.

Shashi Narayan, Shay B Cohen, and Mirella Lapata. 2018a. Don't give me the details, just the summary! topic-aware convolutional neural networks for extreme summarization. *arXiv preprint arXiv:1808.08745*.

Shashi Narayan, Shay B Cohen, and Mirella Lapata. 2018b. Ranking sentences for extractive summarization with reinforcement learning. *arXiv preprint arXiv:1802.08636*.

Maxwell Nye, Anders Johan Andreassen, Guy Gur-Ari, Henryk Michalewski, Jacob Austin, David Bieber, David Dohan, Aitor Lewkowycz, Maarten Bosma, David Luan, et al. 2021. Show your work: Scratchpads for intermediate computation with language models. *arXiv preprint arXiv:2112.00114*.

Mathieu Ravaut, Shafiq Joty, and Nancy F Chen. 2022. Summareranker: A multi-task mixture-of-experts reranking framework for abstractive summarization. *arXiv preprint arXiv:2203.06569*.

Danqing Wang, Pengfei Liu, Yining Zheng, Xipeng Qiu, and Xuanjing Huang. 2020. Heterogeneous graph neural networks for extractive document summarization. *arXiv preprint arXiv:2004.12393*.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Ed Chi, Quoc Le, and Denny Zhou. 2022. Chain of thought prompting elicits reasoning in large language models. *arXiv preprint arXiv:2201.11903*.

Jiacheng Xu, Zhe Gan, Yu Cheng, and Jingjing Liu. 2019. Discourse-aware neural extractive model for text summarization. *arXiv preprint arXiv:1910.14142*.

Xianjun Yang, Yan Li, Xinlu Zhang, Haifeng Chen, and Wei Cheng. 2023. Exploring the limits of chatgpt for query or aspect-based text summarization. *arXiv preprint arXiv:2302.08081*.

Xi Ye, Srinivasan Iyer, Asli Celikyilmaz, Ves Stoyanov, Greg Durrett, and Ramakanth Pasunuru. 2022. Complementary explanations for effective in-context learning. *arXiv preprint arXiv:2211.13892*.

Haopeng Zhang, Xiao Liu, and Jiawei Zhang. 2022a. Hegel: Hypergraph transformer for long document summarization. *arXiv preprint arXiv:2210.04126*.

Haopeng Zhang, Semih Yavuz, Wojciech Kryscinski, Kazuma Hashimoto, and Yingbo Zhou. 2022b. Improving the faithfulness of abstractive summarization via entity coverage control. *arXiv preprint arXiv:2207.02263*.

Jiawei Zhang, Haopeng Zhang, Li Sun, and Congying Xia. 2020. Graph-bert: Only attention is needed for learning graph representations. *arXiv preprint arXiv:2001.05140*.

Tianyi Zhang, Faisal Ladhak, Esin Durmus, Percy Liang, Kathleen McKeown, and Tatsunori B Hashimoto. 2023. Benchmarking large language models for news summarization. *arXiv preprint arXiv:2301.13848*.

Ming Zhong, Pengfei Liu, Yiran Chen, Danqing Wang, Xipeng Qiu, and Xuanjing Huang. 2020. Extractive summarization as text matching. *arXiv preprint arXiv:2004.08795*.

Appendix

A Example summaries

Here we show the ChatGPT-generated summaries with different prompt settings in Table 6 for one example from CNNDM dataset, together with their golden references.

Document	Daredevil Nik Wallenda says he'll walk untethered on top of a 400-foot observation wheel in Orlando, Florida, this month. Wallenda said Monday at a New York City news conference that the Orlando Eye will be moving when he attempts his feat on April 29. The Orlando Eye, part of a new entertainment complex, will offer views of central Florida from inside 30 enclosed, air-conditioned glass capsules when it opens to the public on May 4. Eyes on the prize: high-wire performer Nik Wallenda announces his latest stunt at the 400-foot Orlando Eye, during a news conference, in New York on Monday. Tough challenge: the 36-year-old daredevil will walk atop the Orlando Eye as it turns on April 29. The Orlando Eye team issued a statement saying it's excited to have Wallenda attempt the 'amazing stunt.' No distance for the performance has been set yet, but Wallenda, 36, said he was not likely to walk the entire 20 minutes or so that it takes the wheel to go a full circle. Wallenda previously walked atop a Ferris wheel in Santa Cruz, California, but he said the size of the much larger Orlando wheel and the fact that he will not use a pole sets this stunt apart. The seventh-generation member of the 'Flying Wallenda' family of acrobats has walked across the Grand Canyon and Niagara Falls. In November, he walked twice between two Chicago skyscrapers without a net or harness, doing the second walk blindfolded. Wallenda is the great-grandson of Karl Wallenda, who fell to his death in Puerto Rico at 73.
Reference	The 36-year-old will stage his next stunt on April 29. In November, Wallenda walked back and forth between two Chicago skyscrapers in a live television event. His great-grandfather Karl Wallenda died in a tightrope walk in Puerto Rico in 1978. Wallenda has also tightrope walked across Niagara Falls and the Grand Canyon.
ORACLE	Tough challenge: the 36-year-old daredevil will walk atop the Orlando Eye as it turns on April 29. The seventh-generation member of the 'Flying Wallenda' family of acrobats has walked across the Grand Canyon and Niagara Falls. In November, he walked twice between two Chicago skyscrapers without a net or harness, doing the second walk blindfolded.
Abstractive	Daredevil Nik Wallenda plans to walk untethered atop the 400-foot Orlando Eye observation wheel on April 29th. The 36-year-old high-wire performer, a seventh-generation member of the "Flying Wallenda" acrobat family, has previously completed daring walks across the Grand Canyon, Niagara Falls, and between Chicago skyscrapers. Although the exact distance for this stunt hasn't been determined, Wallenda stated he's unlikely to walk the entire 20 minutes it takes for the wheel to complete a full circle.
Extractive	Daredevil Nik Wallenda says he'll walk untethered on top of a 400-foot observation wheel in Orlando, Florida, this month. Wallenda said Monday at a New York City news conference that the Orlando Eye will be moving when he attempts his feat on April 29. The 36-year-old daredevil will walk atop the Orlando Eye as it turns on April 29.
In-Context	Daredevil Nik Wallenda says he'll walk untethered on top of a 400-foot observation wheel in Orlando, Florida, this month. Wallenda said Monday at a New York City news conference that the Orlando Eye will be moving when he attempts his feat on April 29. Wallenda is the great-grandson of Karl Wallenda, who fell to his death in Puerto Rico at 73.
Extract-abstract	Nik Wallenda, a daredevil and seventh-generation member of the "Flying Wallenda" family of acrobats, will attempt to walk untethered on top of a 400-foot observation wheel in Orlando, Florida, on April 29. The Orlando Eye, part of a new entertainment complex, will offer views of central Florida from inside 30 enclosed, air-conditioned glass capsules when it opens to the public on May 4. Wallenda previously walked atop a Ferris wheel in Santa Cruz, California, but he said the size of the much larger Orlando wheel and the fact that he will not use a pole sets this stunt apart. Wallenda is the great-grandson of Karl Wallenda, who fell to his death in Puerto Rico at 73.

Table 6: Case study of different settings