The Benefits of a Concise Chain of Thought on Problem-Solving in LLMs

Matthew Renze and Erhan Guven

Johns Hopkins University

A Dilemma

Chain of Thought

A Dilemma

Chain of Thought

Beneficial for problem-solving

A Dilemma

Chain of Thought

Beneficial for problem-solving

Creates longer responses

A Dilemma

Chain of Thought

Beneficial for problem-solving

Creates longer responses

More cost, time, power

A Dilemma

Chain of Thought

Beneficial for problem-solving

Creates longer responses

More cost, time, power

Concise Prompting

A Dilemma

Chain of Thought

Beneficial for problem-solving

Creates longer responses

More cost, time, power

Concise Prompting

Instruct LLM to “be concise.”

A Dilemma

Chain of Thought

Beneficial for problem-solving

Creates longer responses

More cost, time, power

Concise Prompting

Instruct LLM to “be concise.”

Reduces response length

A Dilemma

Chain of Thought	**Concise Prompting**
Beneficial for problem-solving	Instruct LLM to “be concise.”
Creates longer responses	Reduces response length
More cost, time, power	Less cost, time, power

A Dilemma

Chain of Thought

Beneficial for problem-solving

Creates longer responses

More cost, time, power

or

Concise Prompting

Instruct LLM to “be concise.”

Reduces response length

Less cost, time, power

Can we combine chain-of-thought and concise prompting?

The Benefits of a Concise Chain of Thought on Problem-Solving in Large Language Models

1st Matthew Renze
Johns Hopkins University
Baltimore, MD, USA
mrenze1@jhu.edu

2nd Erhan Guven
Johns Hopkins University
Baltimore, MD, USA
eguven2@jhu.edu

***Abstract*—In this paper, we introduce Concise Chain-of-Thought (CCoT) prompting. We compared standard CoT and CCoT prompts to see how conciseness impacts response length and correct-answer accuracy. We evaluated this using GPT-3.5 and GPT-4 with a multiple-choice question-and-answer (MCQA) benchmark. CCoT reduced average response length by 48.70% for both GPT-3.5 and GPT-4 while having a negligible impact on problem-solving performance. However, on math problems, GPT-3.5 with CCoT incurred a performance penalty of 27.69%. Overall, CCoT leads to an average per-token cost reduction of 22.67%. All code, data, and supplemental materials are available on GitHub at https://github.com/matthewrenze/jhu-concise-cot**

***Index Terms*—large language model, LLM, chain-of-thought, CoT, concise**

the cost of using the LLM with CoT grows in proportion to response length.

C. Concise Prompting

Concise prompting is a prompt-engineering technique used to reduce LLM response verbosity. The main benefit is that it decreases the per-token cost of using the LLM. In addition, it can reduce the LLM's energy consumption, minimize response time, and improve communication efficiency with the end user.

There are two main implementations of concise prompting. Zero-shot prompting instructs the LLM to "be concise" in its response [6], [7]. Few-shot prompting requires the prompt

Background

Chain-of-Thought (CoT) Prompting

Sources:

Kojima, et al. (2023)
Large language models are zero-shot reasoners

Wei, et al. (2022)
Chain-of-thought prompting elicits reasoning in large language models

Mialon, et al. (2023)
Augmented language models: a survey

Spencer-Smith and Schmidt (2023)
A prompt pattern catalog to enhance prompt engineering with chatgpt

Chain-of-Thought (CoT) Prompting

Zero-shot: “think step by step”

Sources:

Kojima, et al. (2023)
Large language models are zero-shot reasoners

Wei, et al. (2022)
Chain-of-thought prompting elicits reasoning in large language models

Mialon, et al. (2023)
Augmented language models: a survey

Spencer-Smith and Schmidt (2023)
A prompt pattern catalog to enhance prompt engineering with chatgpt

Chain-of-Thought (CoT) Prompting

Zero-shot: “think step by step”

Few-shot: provide examples

Sources:

Kojima, et al. (2023)
Large language models are zero-shot reasoners

Wei, et al. (2022)
Chain-of-thought prompting elicits reasoning in large language models

Mialon, et al. (2023)
Augmented language models: a survey

Spencer-Smith and Schmidt (2023)
A prompt pattern catalog to enhance prompt engineering with chatgpt

Chain-of-Thought (CoT) Prompting

Zero-shot: “think step by step”

Few-shot: provide examples

Increases performance

Sources:

Kojima, et al. (2023)
Large language models are zero-shot reasoners

Wei, et al. (2022)
Chain-of-thought prompting elicits reasoning in large language models

Mialon, et al. (2023)
Augmented language models: a survey

Spencer-Smith and Schmidt (2023)
A prompt pattern catalog to enhance prompt engineering with chatgpt

Chain-of-Thought (CoT) Prompting

Zero-shot: “think step by step”

Few-shot: provide examples

Increases performance

Increases response length

Sources:

Kojima, et al. (2023)
Large language models are zero-shot reasoners

Wei, et al. (2022)
Chain-of-thought prompting elicits reasoning in large language models

Mialon, et al. (2023)
Augmented language models: a survey

Spencer-Smith and Schmidt (2023)
A prompt pattern catalog to enhance prompt engineering with chatgpt

Concise Prompting

Sources:

Crispino, et al. (2023)
Agent instructs large language models
to be general zero-shot reasoners

Kadous (2023)
Numbers every LLM developer should know

Concise Prompting

Zero-shot: “be concise”

Sources:

Crispino, et al. (2023)
Agent instructs large language models
to be general zero-shot reasoners

Kadous (2023)
Numbers every LLM developer should know

Concise Prompting

Zero-shot: “be concise”

Few-shot: concise examples

Sources:

Crispino, et al. (2023)
Agent instructs large language models
to be general zero-shot reasoners

Kadous (2023)
Numbers every LLM developer should know

Concise Prompting

Zero-shot: “be concise”

Few-shot: concise examples

Reduces response length

Sources:

Crispino, et al. (2023)
Agent instructs large language models
to be general zero-shot reasoners

Kadous (2023)
Numbers every LLM developer should know

Concise Prompting

Zero-shot: “be concise”

Few-shot: concise examples

Reduces response length

May reduce performance

Sources:

Crispino, et al. (2023)
Agent instructs large language models
to be general zero-shot reasoners

Kadous (2023)
Numbers every LLM developer should know

Concise Chain-of-Thought (CCoT) Prompting

Combines CoT with conciseness

Sources:

Madaan and Yazdanbakhsh
Text and patterns: For effective chain of thought, it takes two to tango

Concise Chain-of-Thought (CCoT) Prompting

Combines CoT with conciseness

Shortest possible correct solution

Sources:

Madaan and Yazdanbakhsh
Text and patterns: For effective chain of thought, it takes two to tango

Concise Chain-of-Thought (CCoT) Prompting

Combines CoT with conciseness

Shortest possible correct solution

Prior research: Decompose CoT

Sources:

Madaan and Yazdanbakhsh
Text and patterns: For effective chain of thought, it takes two to tango

Concise Chain-of-Thought (CCoT) Prompting

Combines CoT with conciseness

Shortest possible correct solution

Prior research: Decompose CoT

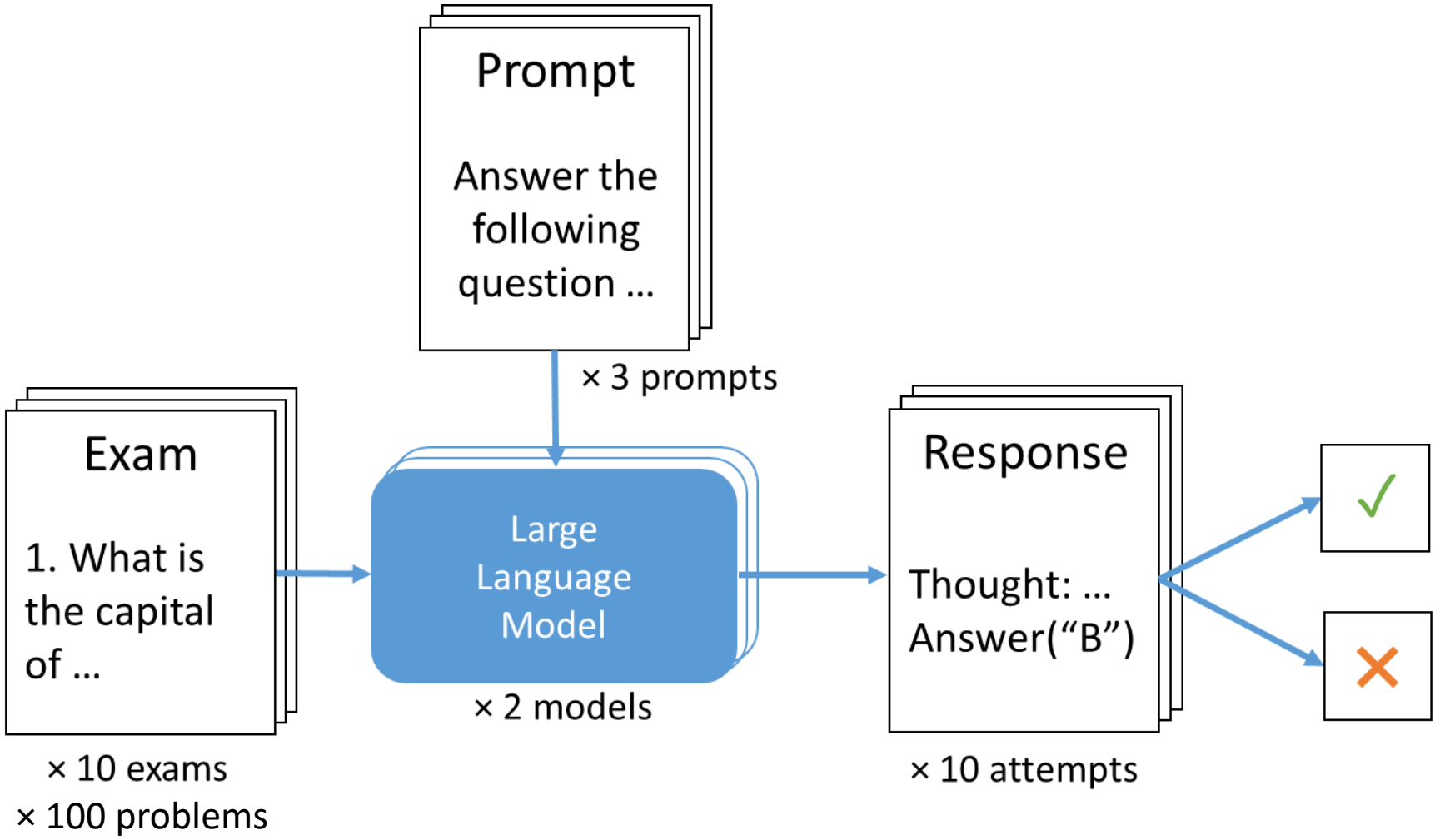
No other known research on CCoT

Sources:

Madaan and Yazdanbakhsh
Text and patterns: For effective chain of thought, it takes two to tango

Methods

Experiment


Prompt
Answer the following question …
× 3 prompts
Exam
1. What is the capital of …
× 10 exams
× 100 problems
Large Language Model
× 2 models
Response
Thought: …
Answer("B")
× 10 attempts

Models

Name	Vendor	Released	License	Source
GPT-3.5 Turbo	OpenAI	2022-11-30	Closed	OpenAI (2022)
GPT-4	OpenAI	2023-03-14	Closed	OpenAI (2023)

Exams

Problem Set	Benchmark	Domain	Questions	License	Source
ARC Challenge Test	ARC	Science	1,173	CC BY-SA	Clark (2018)
AQUA-RAT	AGI Eval	Math	254	Apache v2.0	Zhong (2023)
Hellaswag Val	Hellaswag	Common Sense Reasoning	10,042	MIT	Zellers (2019)
LogiQA (English)	AGI Eval	Logic	651	GitHub	Liu (2020)
LSAT-AR	AGI Eval	Law (Analytic Reasoning)	230	MIT	Wang (2021)
LSAT-LR	AGI Eval	Law (Logical Reasoning)	510	MIT	Wang (2021)
LSAT-RC	AGI Eval	Law (Reading Comprehension)	260	MIT	Wang (2021)
MedMCQA Valid	MedMCQA	Medicine	6,150	MIT	Pal (2022)
SAT-English	AGI Eval	English	206	MIT	Zhong (2023)
SAT-Math	AGI Eval	Math	220	MIT	Zhong (2023)

Prompts

Answer only – answer the question

Chain of Thought – think step by step

Concise CoT – think step by step *and* be concise

Answer-Only Prompt

[System Prompt]

You are an intelligent assistant.

Your task is to answer the following multiple-choice questions.

Answer the question using the following format ‘Action: Answer("[choice]")’

The parameter [choice] is the letter or number of the answer
you want to select (e.g., "A", "B", "C", or "D").

For example, ‘Answer("C")’ will select choice "C" as the best answer.

You MUST select one of the available choices;
the answer CANNOT be "None of the Above".

Verbose CoT Prompt

[System Prompt]

You are an intelligent assistant.

Your task is to answer the following multiple-choice questions.

Think step-by-step through the problem to ensure you have the correct answer.

Then, answer the question using the following format ‘Action: Answer("[choice]")’

The parameter [choice] is the letter or number of the answer
you want to select (e.g., "A", "B", "C", or "D").

For example, ‘Answer("C")’ will select choice "C" as the best answer.

You MUST select one of the available choices;
the answer CANNOT be "None of the Above".

Concise CoT Prompt

[System Prompt]

You are an intelligent assistant.

Your task is to answer the following multiple-choice questions.

Think step-by-step through the problem to ensure you have the correct answer.

Then, answer the question using the following format ‘Action: Answer("[choice]")’

The parameter [choice] is the letter or number of the answer
you want to select (e.g., "A", "B", "C", or "D").

For example, ‘Answer("C")’ will select choice "C" as the best answer.

You MUST select one of the available choices;
the answer CANNOT be "None of the Above".

Be concise.

Example Problem

Question: What is the capital of the state where Johns Hopkins University is located?

Choices:

A: Baltimore
B: Annapolis
C: Des Moines
D: Las Vegas

Answer-Only Example Solution

Action: Answer("B")

Verbose CoT Example Solution

Thought:

Johns Hopkins University is located in Baltimore.

Baltimore is a city located in the State of Maryland.

The capital of Maryland is Annapolis.

Therefore, the capital of the state where Johns Hopkins University is located is Annapolis.

The answer is B: Annapolis.

Action: Answer("B")

Concise CoT Example Solution

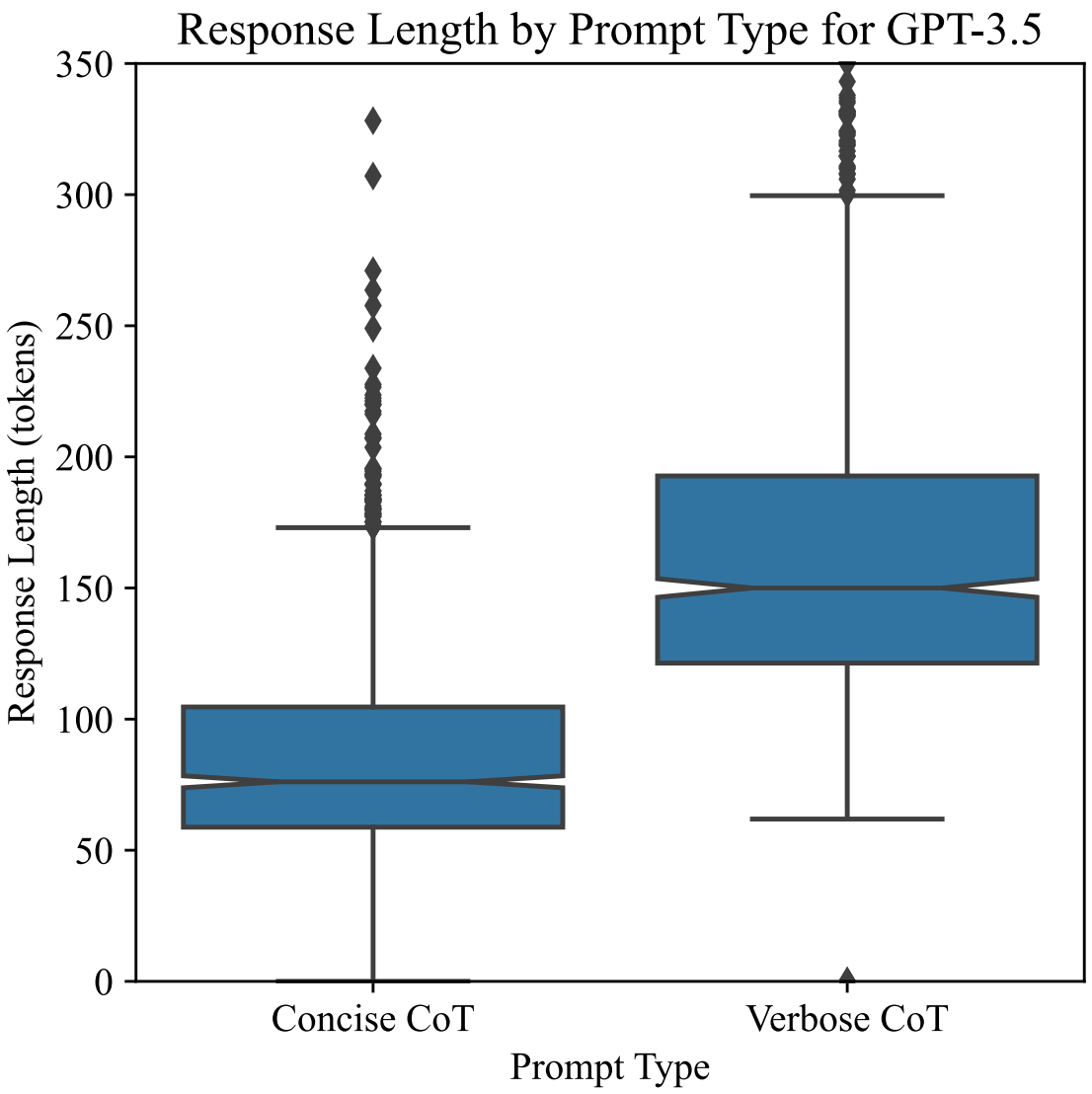
Thought:

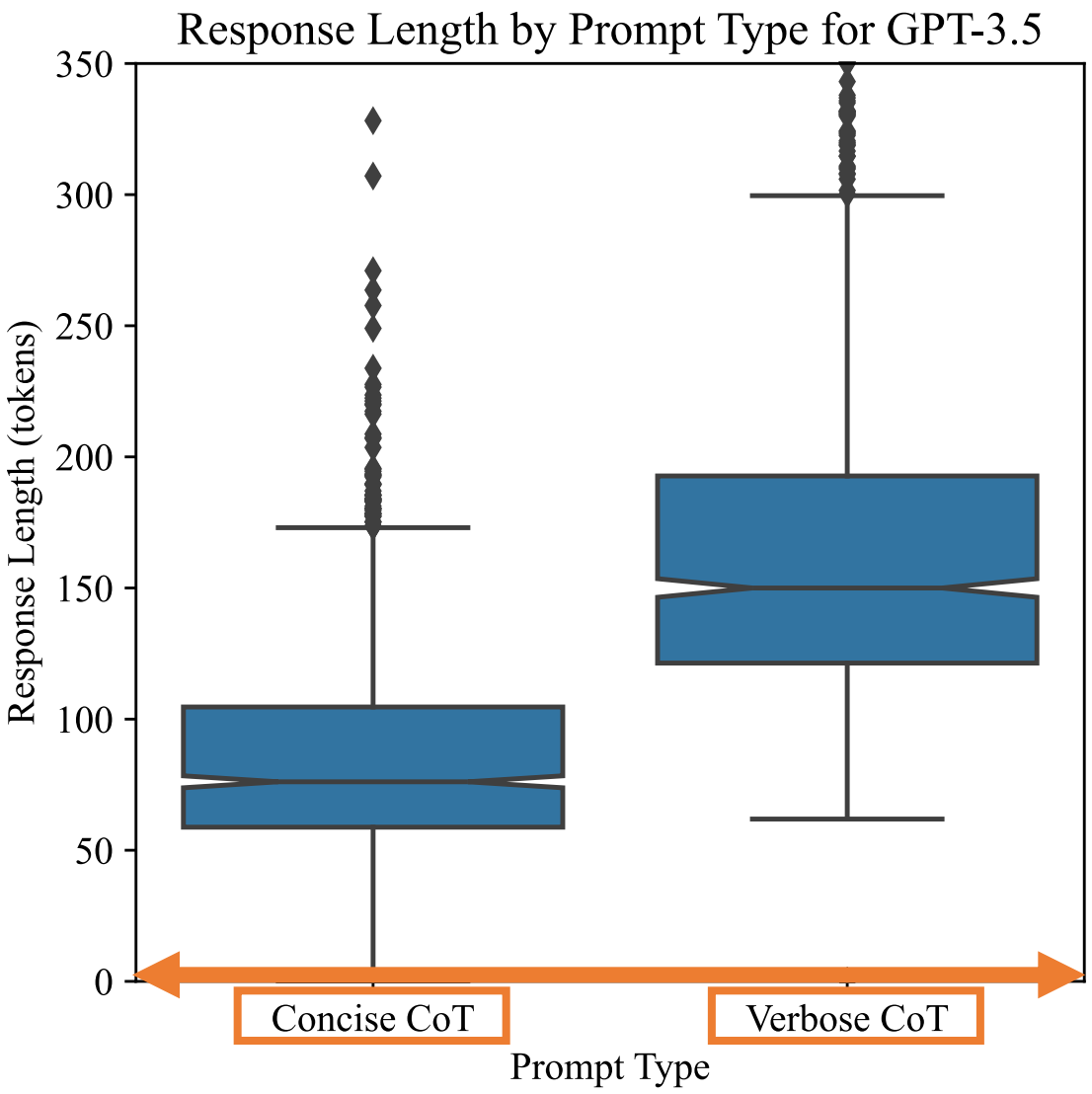
Johns Hopkins University is located in Baltimore, Maryland.

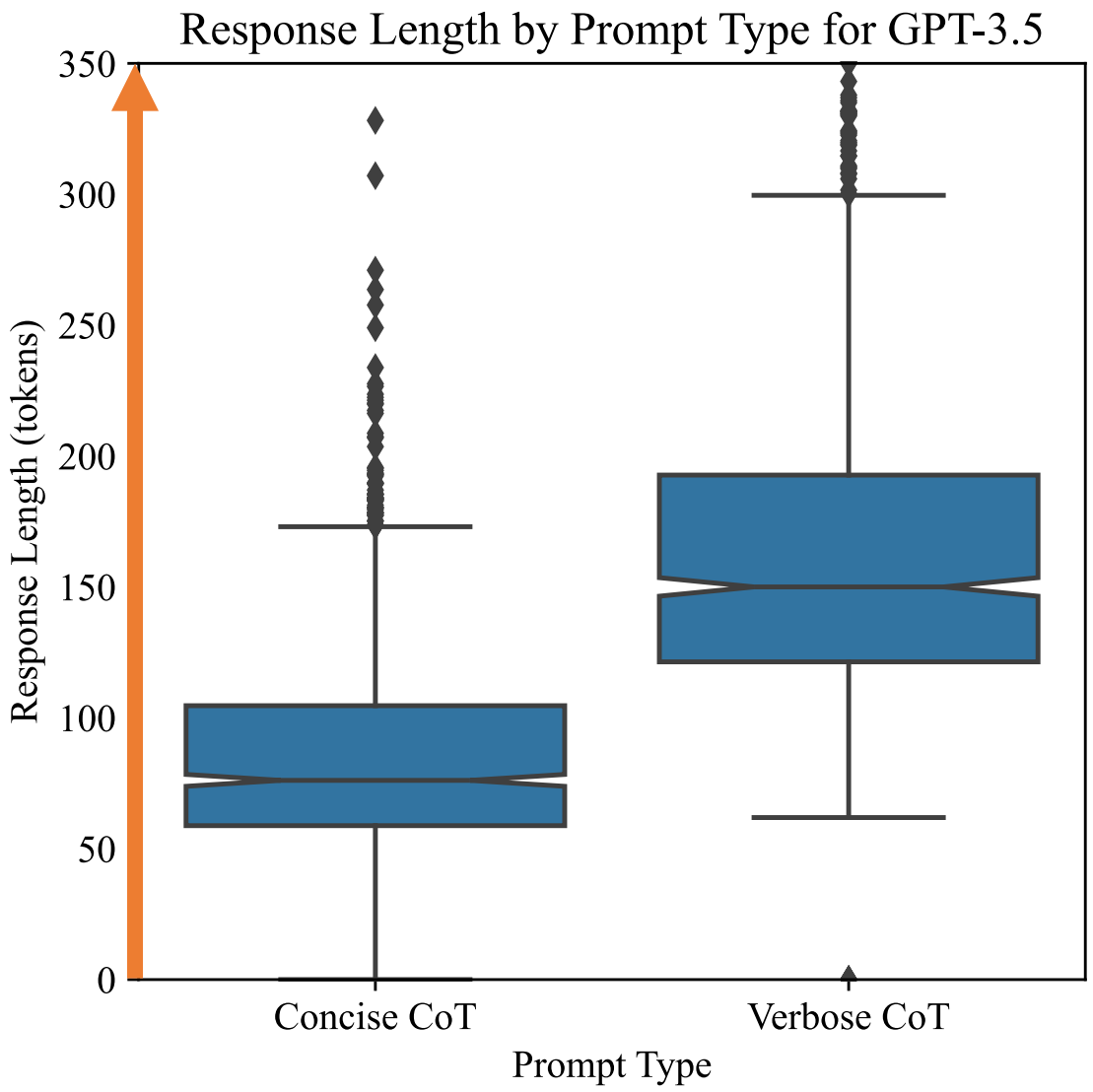
The capital of Maryland is Annapolis.

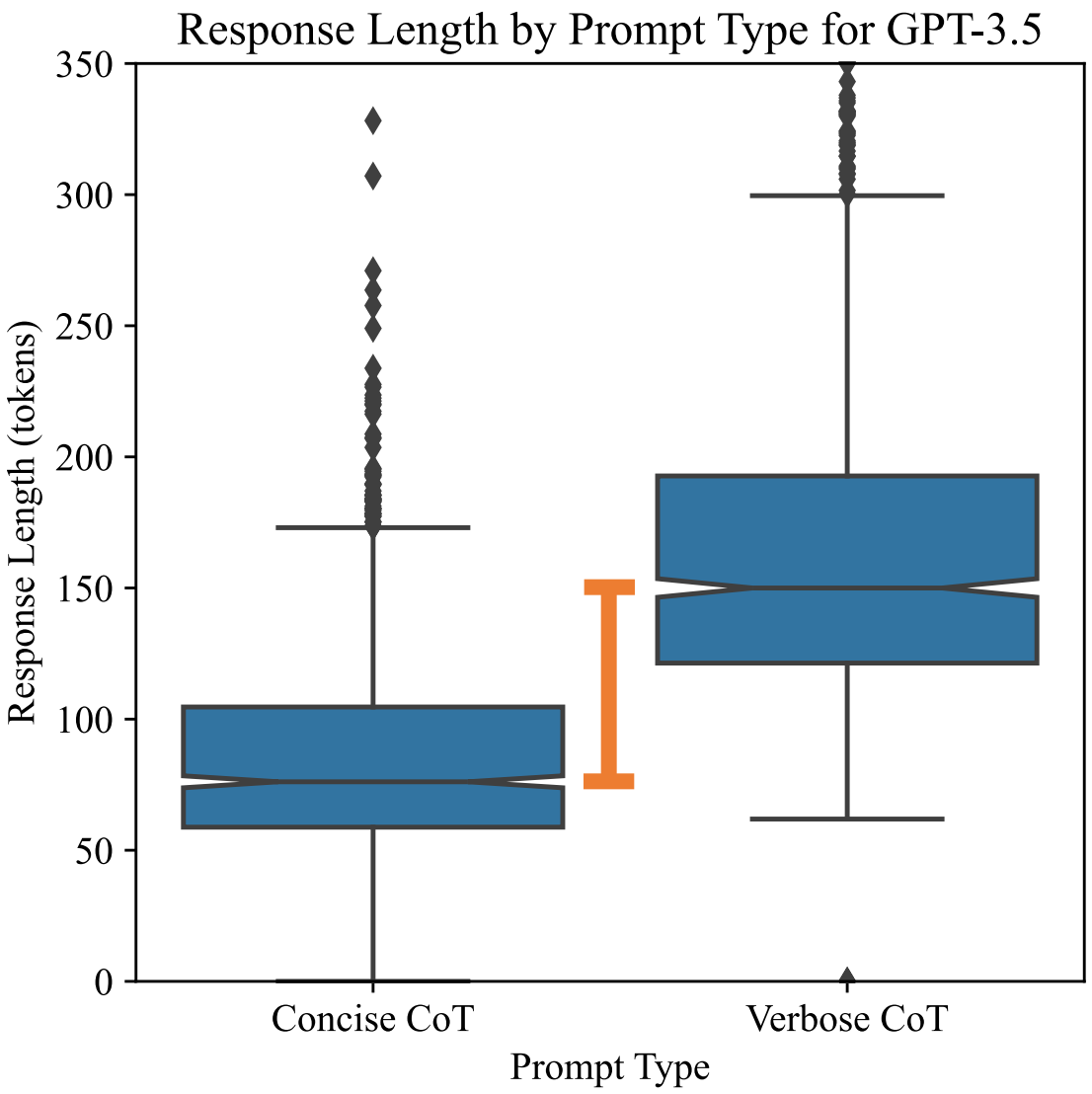
Action: Answer("B")

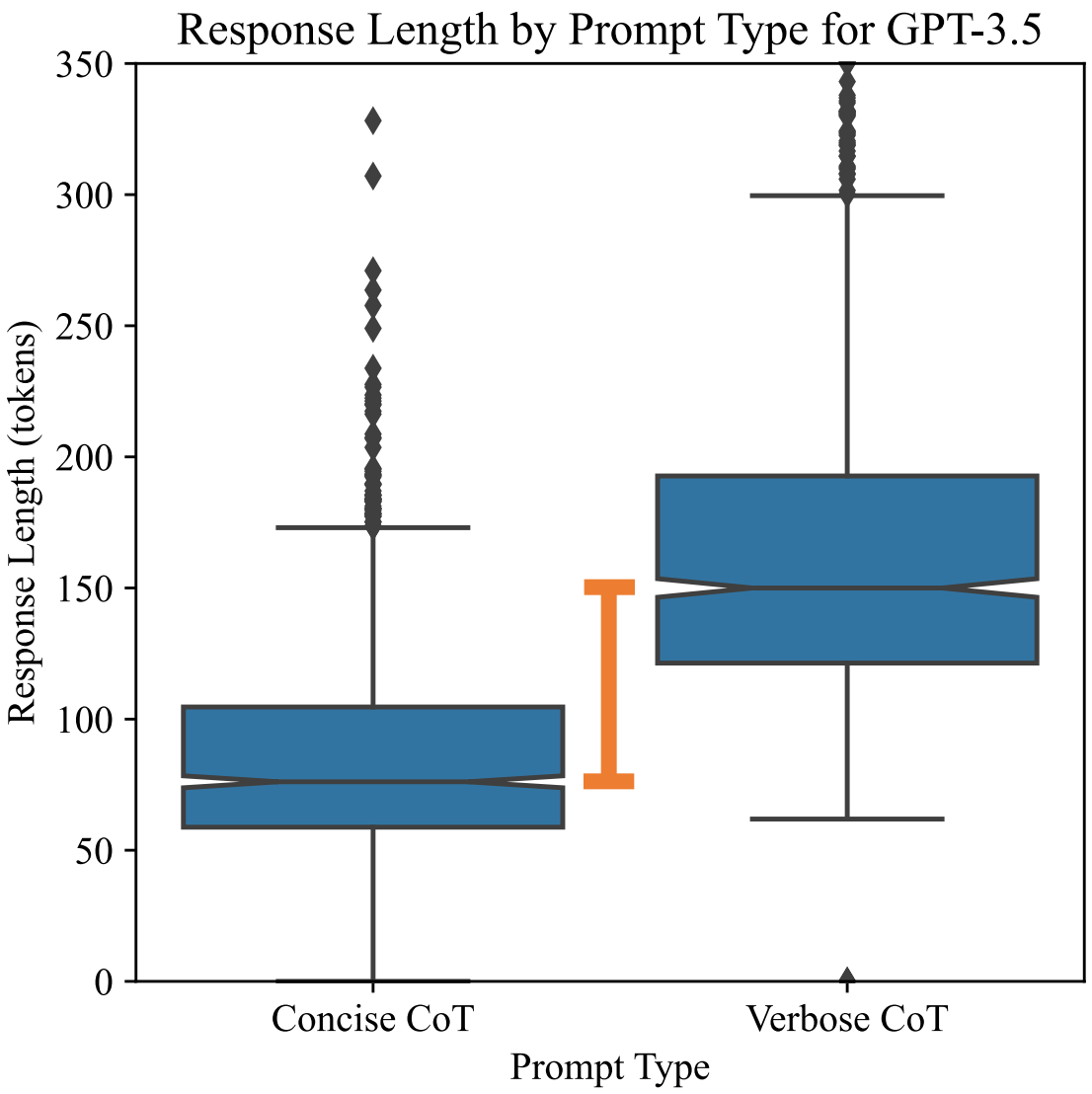
Results

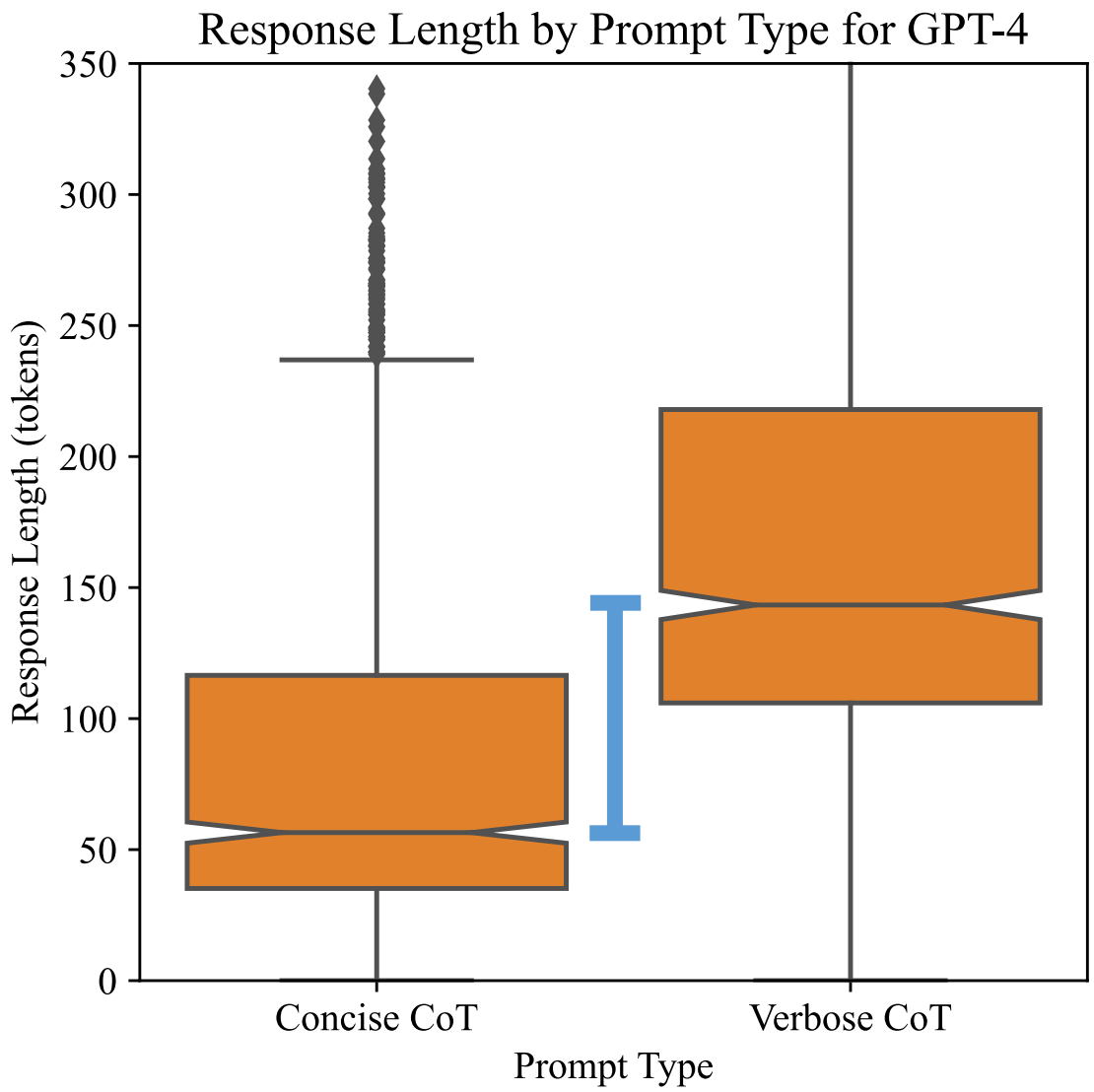

Response Length by Prompt Type for GPT-3.5
Response Length (tokens)
350
300
250
200
150
100
50
0
Concise CoT
Verbose CoT
Prompt Type

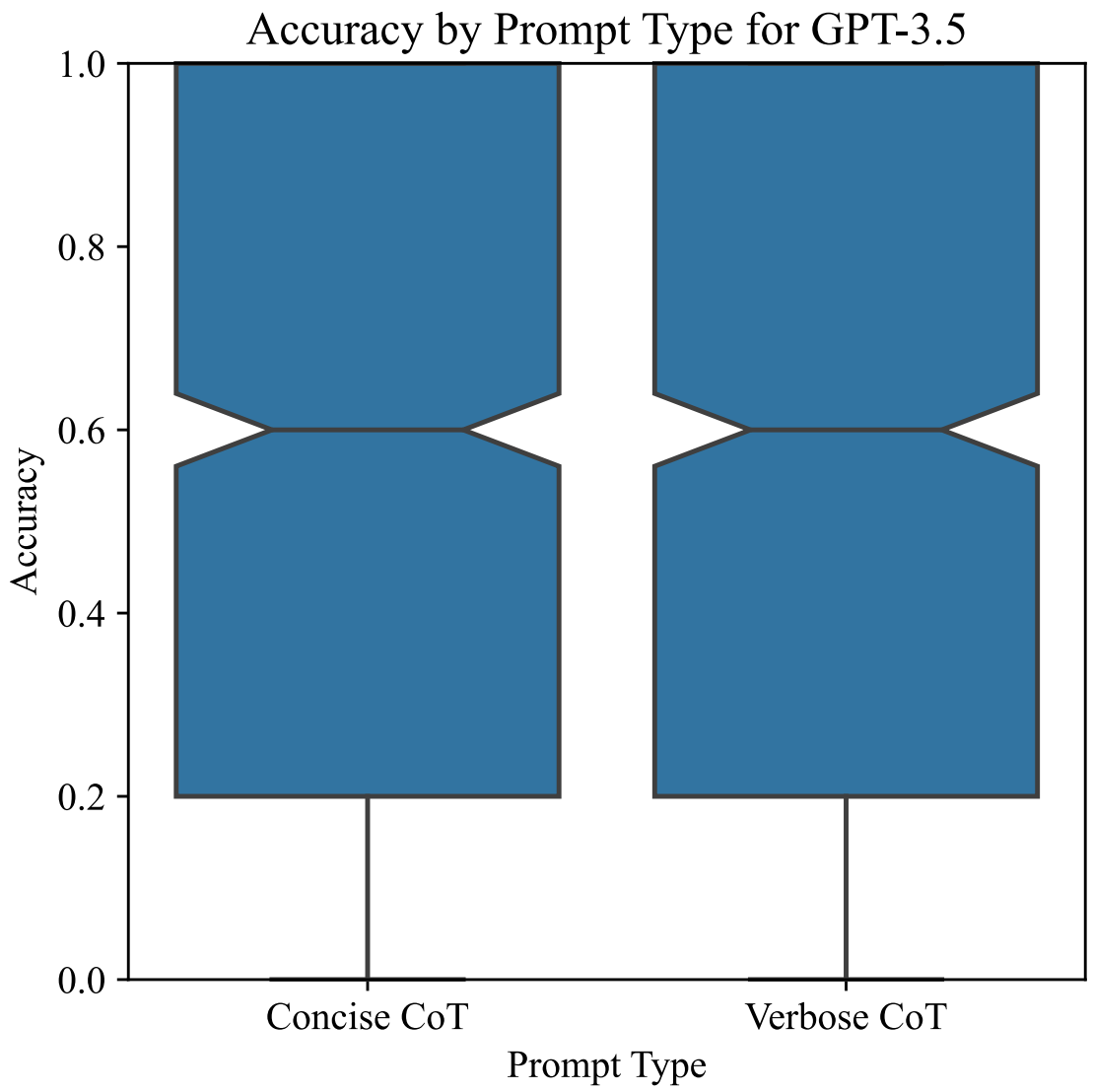

Response Length by Prompt Type for GPT-3.5
Response Length (tokens)
350
300
250
200
150
100
50
0
Concise CoT
Verbose CoT
Prompt Type

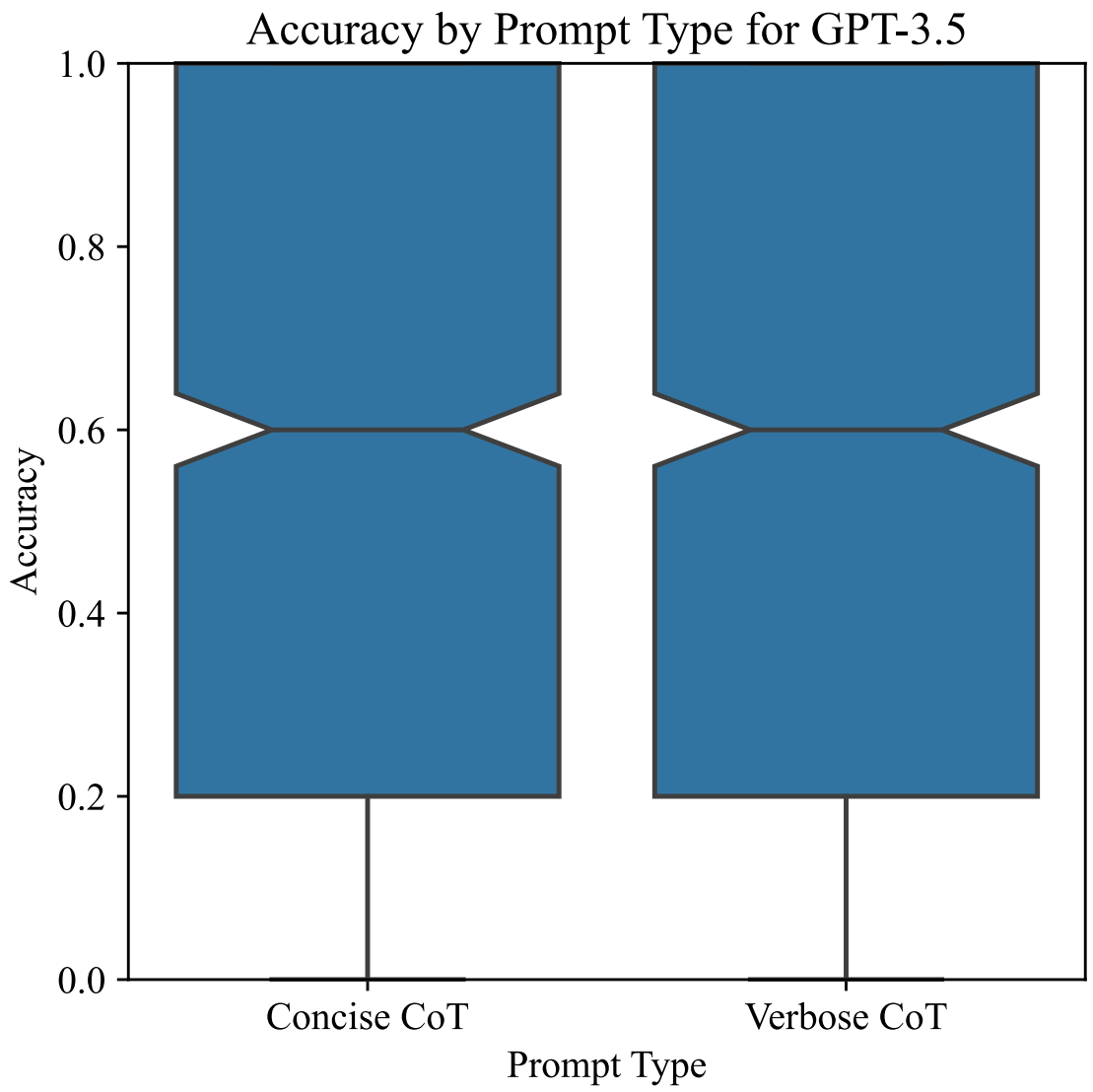

Response Length by Prompt Type for GPT-3.5
Response Length (tokens)
350
300
250
200
150
100
50
0
Concise CoT
Verbose CoT
Prompt Type

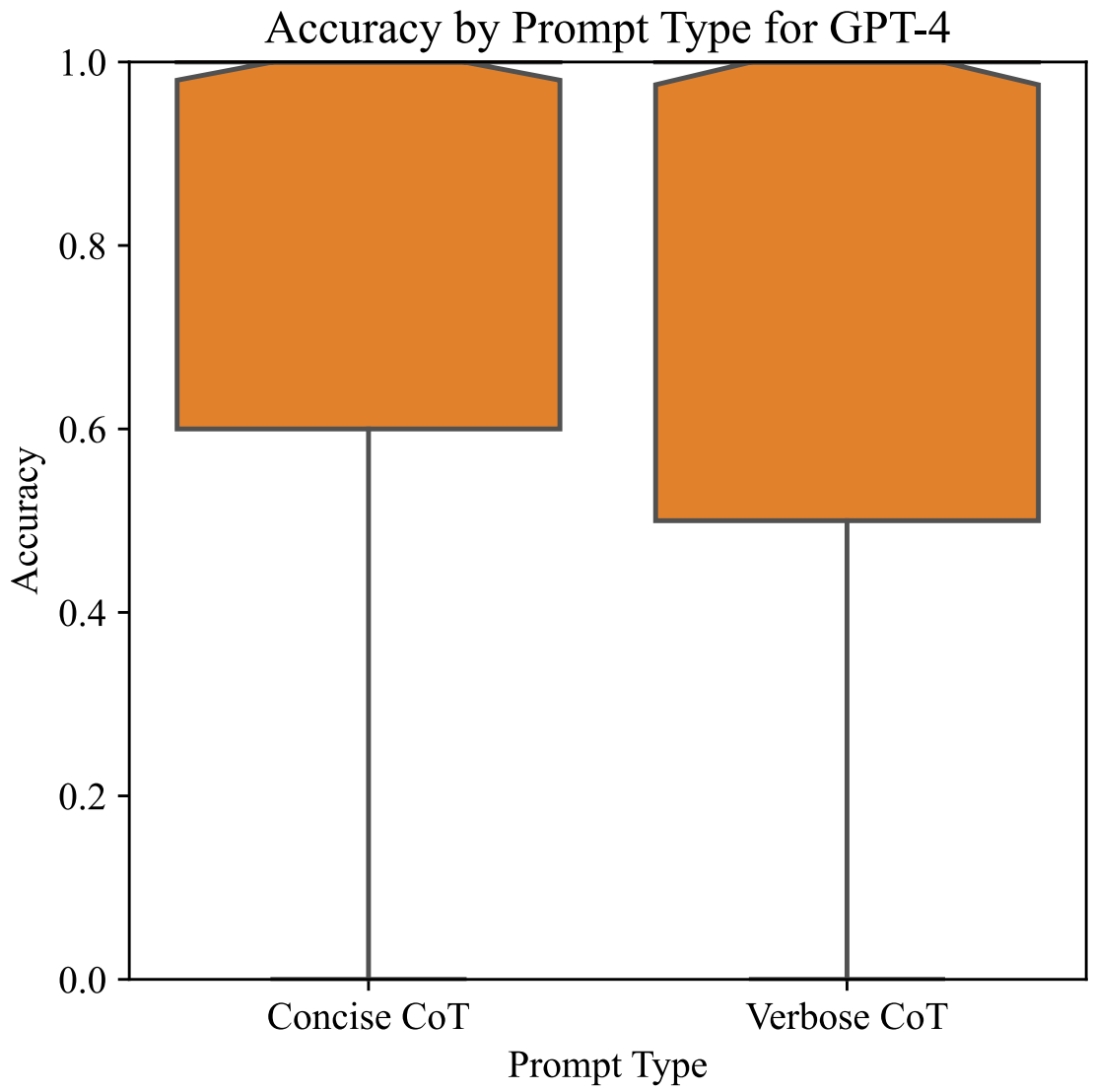

Response Length by Prompt Type for GPT-3.5
Response Length (tokens)
350
300
250
200
150
100
50
0
Concise CoT
Verbose CoT
Prompt Type

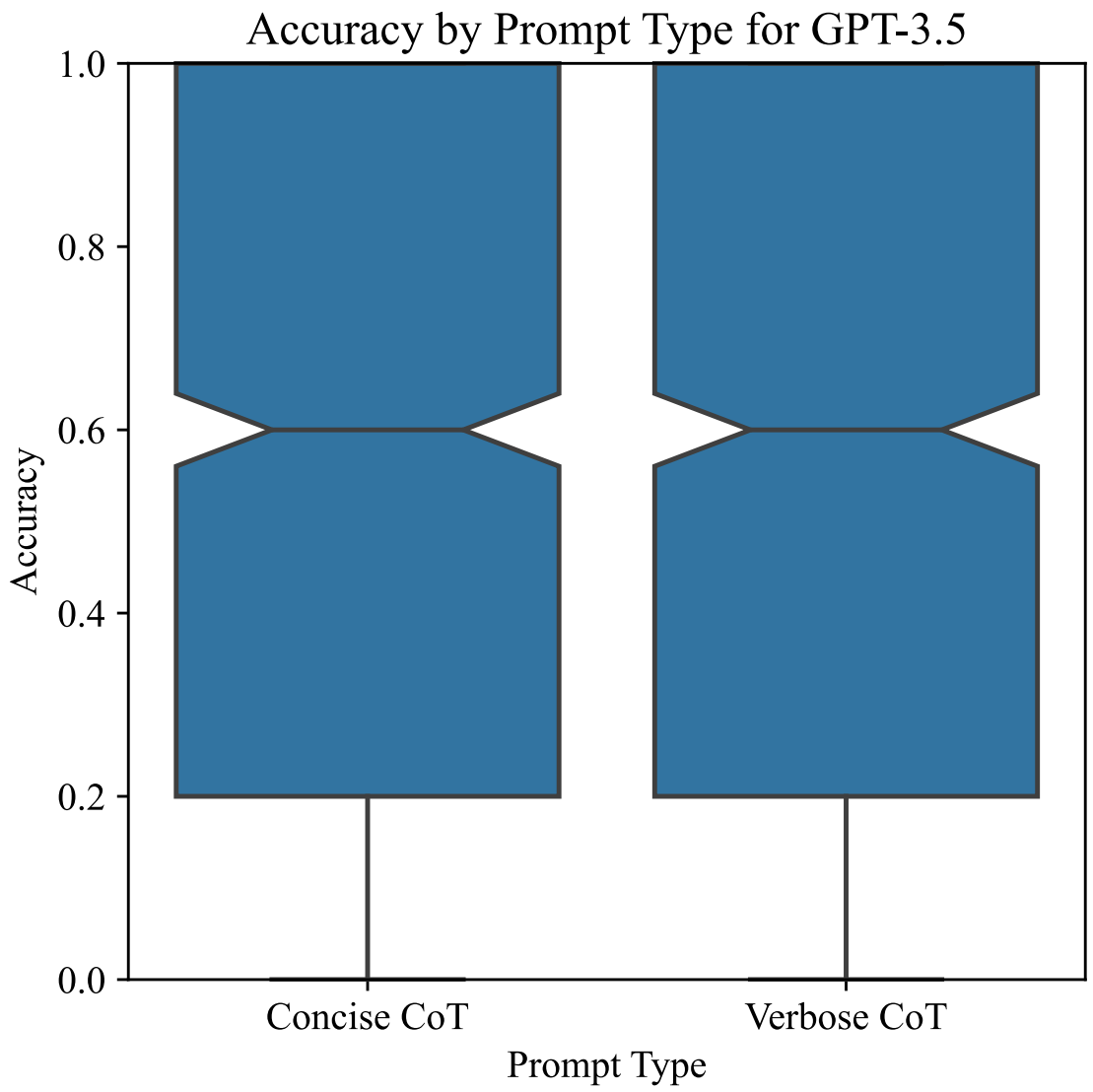

Response Length by Prompt Type for GPT-3.5
Response Length (tokens)
350
300
250
200
150
100
50
0
Concise CoT
Verbose CoT
Prompt Type

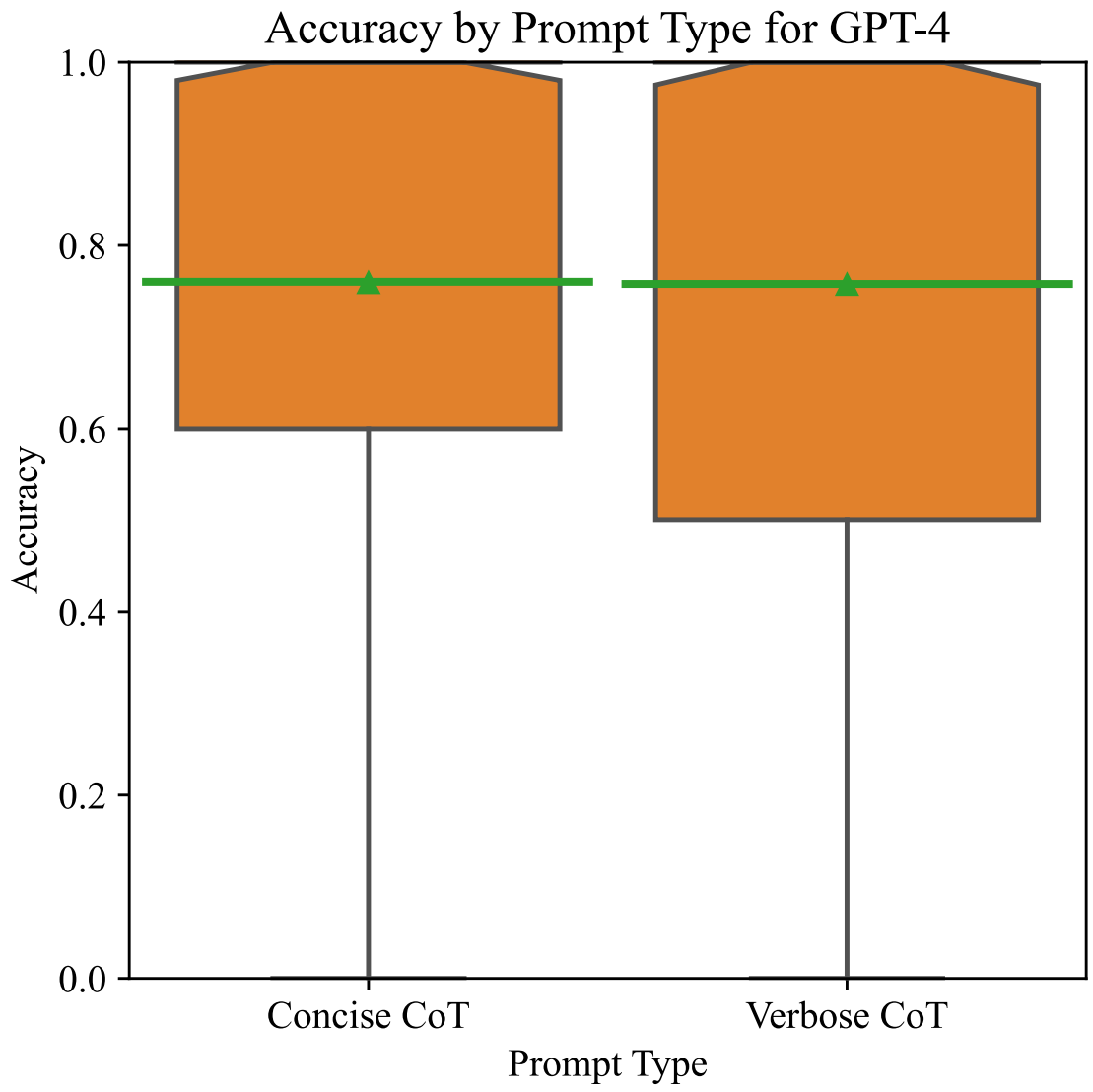

Response Length by Prompt Type for GPT-4
Response Length (tokens)
350
300
250
200
150
100
50
0
Concise CoT
Verbose CoT
Prompt Type

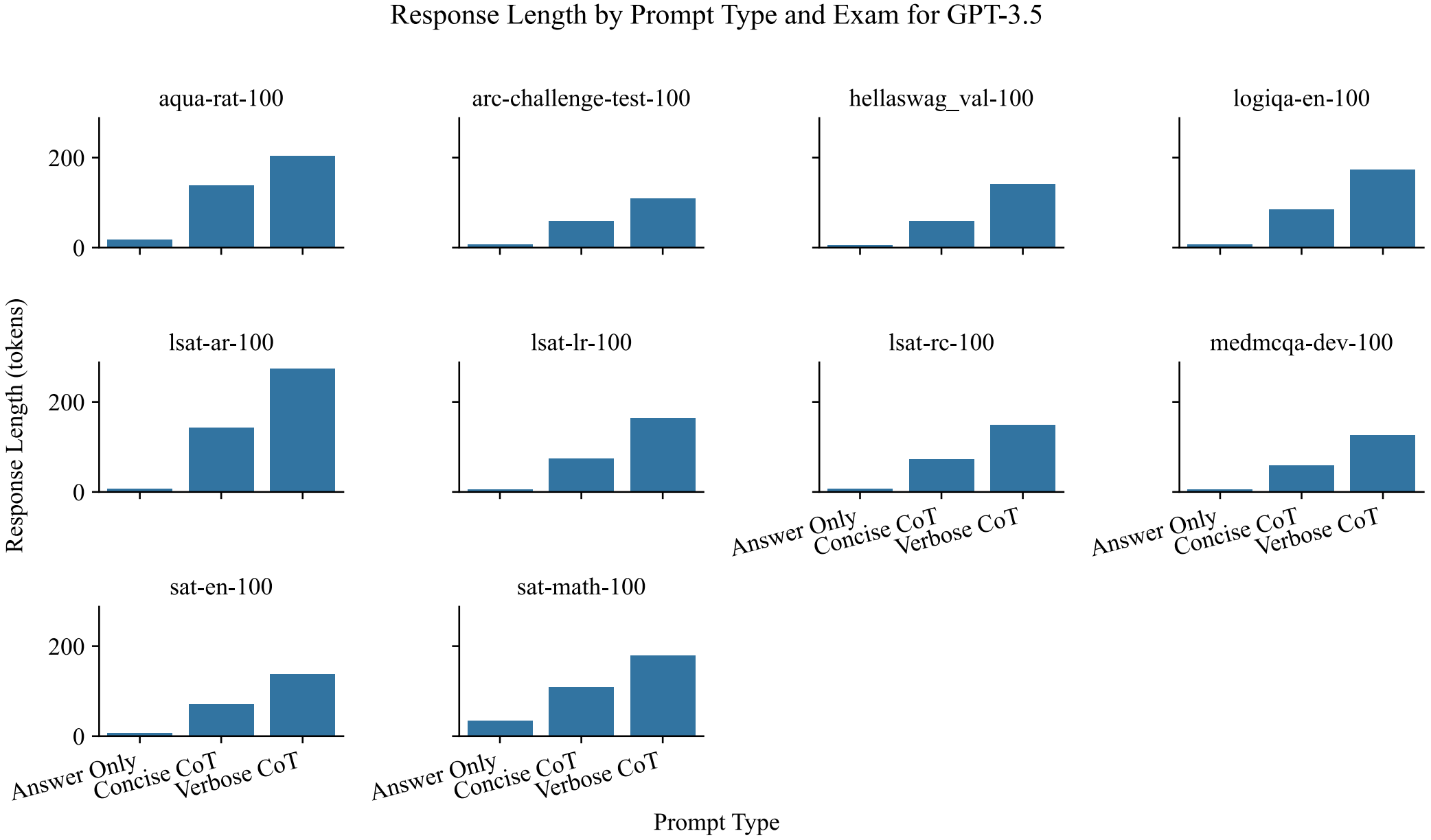

Accuracy by Prompt Type for GPT-3.5
1.0
0.8
0.6
0.4
0.2
0.0
Accuracy
Concise CoT
Verbose CoT
Prompt Type

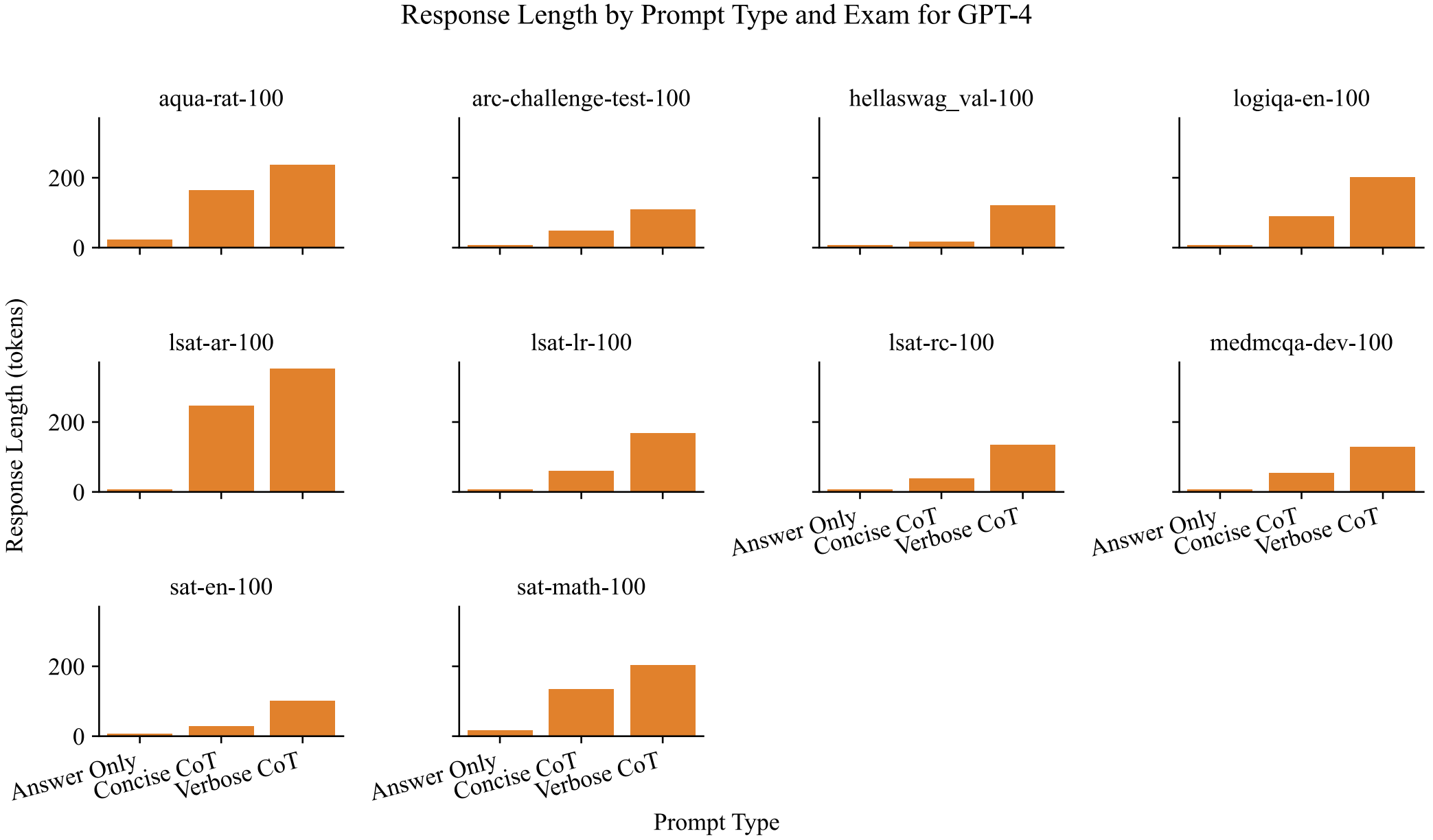

Accuracy by Prompt Type for GPT-3.5
1.0
0.8
0.6
0.4
0.2
0.0
Accuracy
Concise CoT
Verbose CoT
Prompt Type

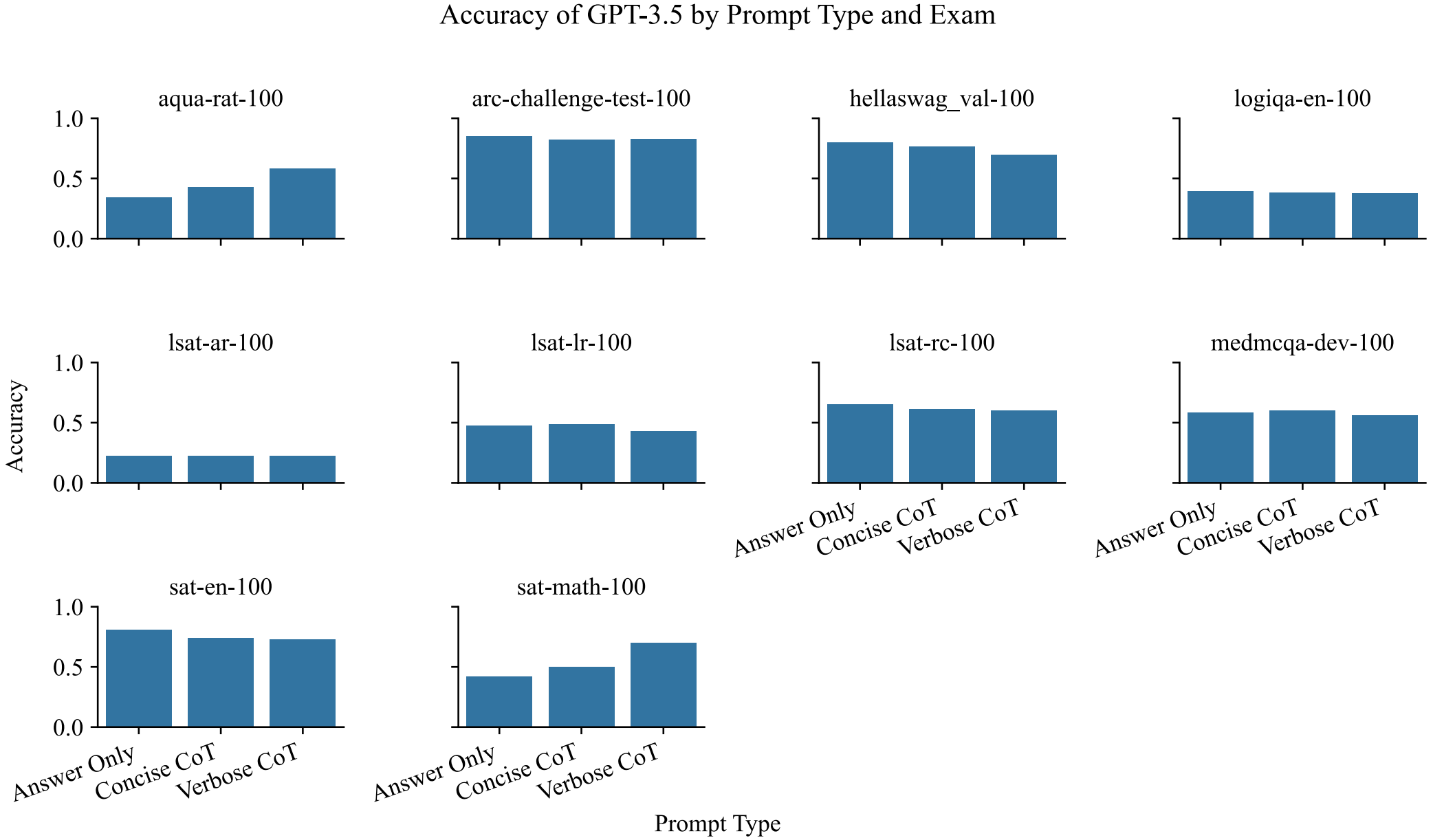

Accuracy by Prompt Type for GPT-4
1.0
0.8
0.6
0.4
0.2
0.0
Accuracy
Concise CoT
Verbose CoT
Prompt Type

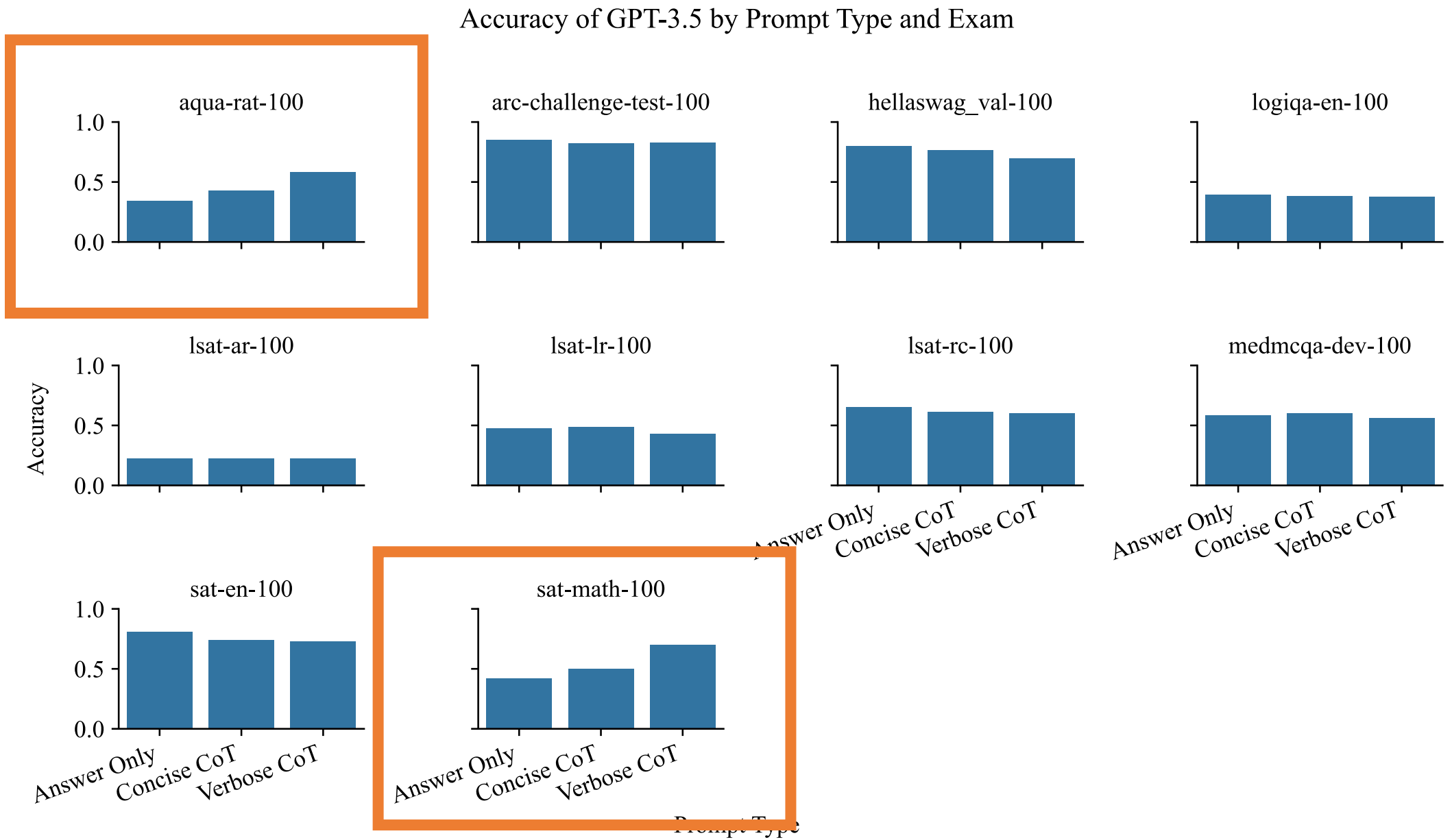

Accuracy by Prompt Type for GPT-3.5
1.0
0.8
0.6
0.4
0.2
0.0
Accuracy
Concise CoT
Verbose CoT
Prompt Type

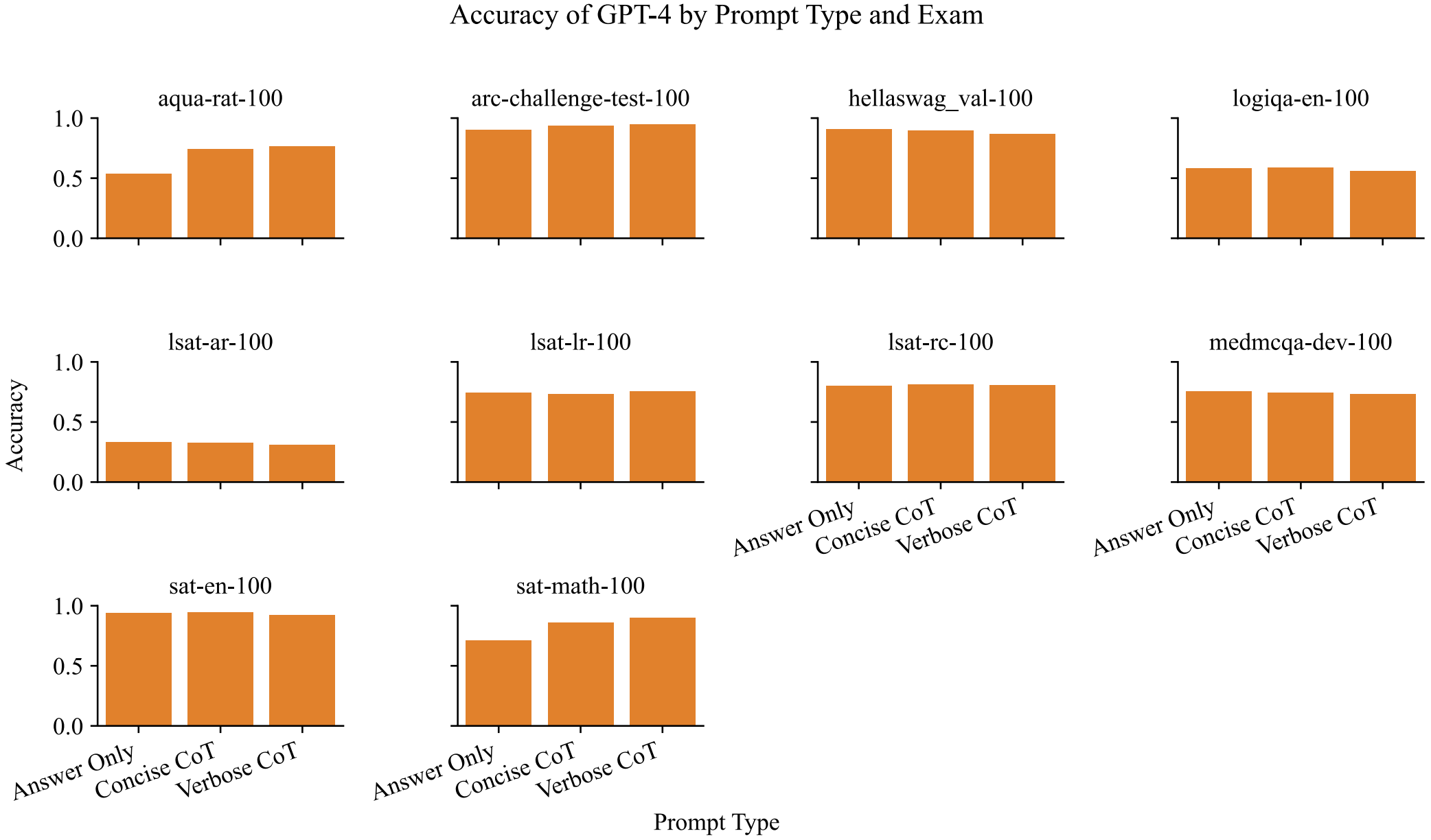

Accuracy by Prompt Type for GPT-4
1.0
0.8
0.6
0.4
0.2
0.0
Accuracy
Concise CoT
Verbose CoT
Prompt Type

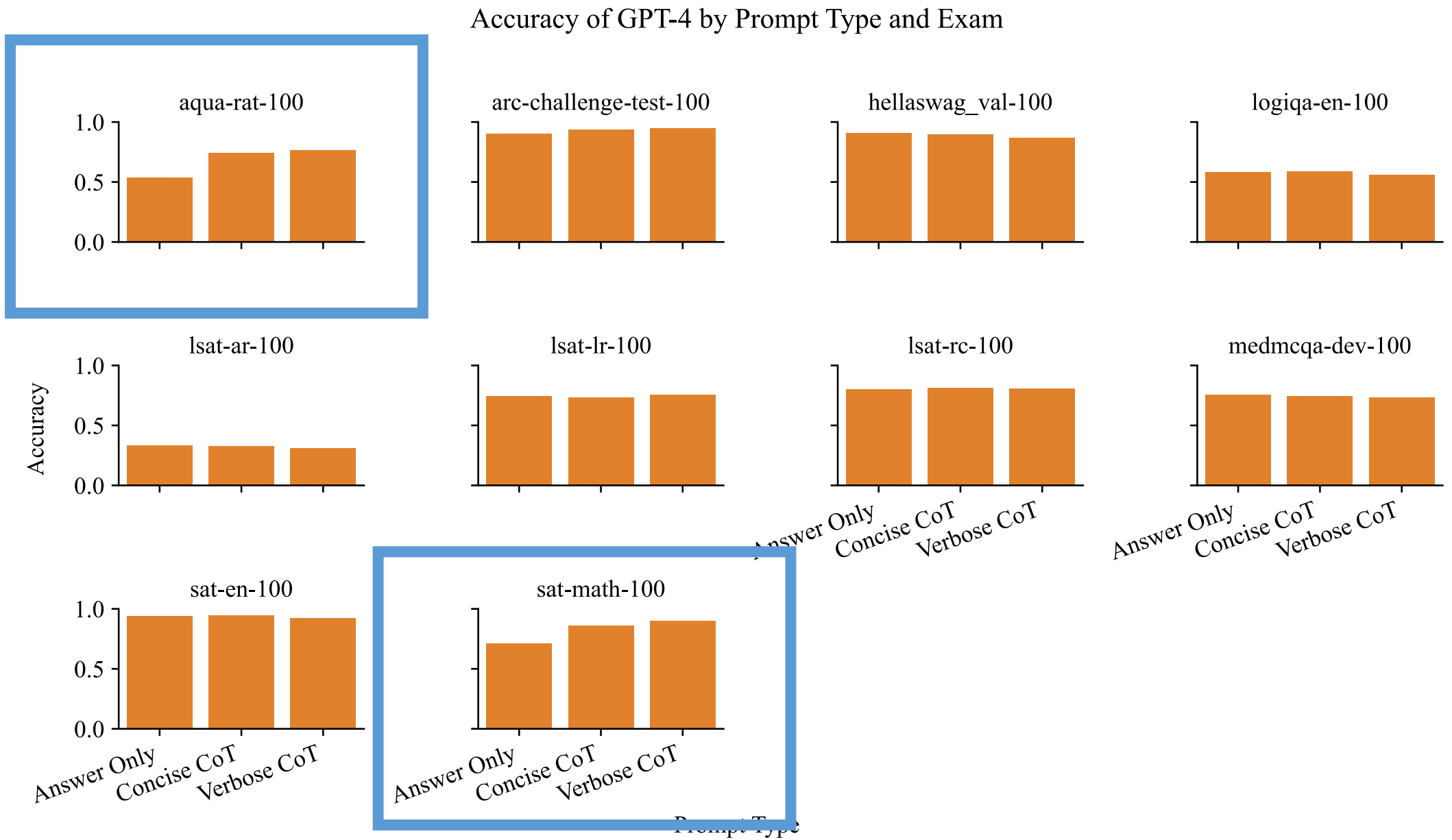

Response Length by Prompt Type and Exam for GPT-3.5
aqua-rat-100
arc-challenge-test-100
hellaswag_val-100
logiqa-en-100
lsat-ar-100
lsat-lr-100
lsat-rc-100
medmcqa-dev-100
sat-en-100
sat-math-100
Response Length (tokens)
200
0
Answer Only
Concise CoT
Verbose CoT
Prompt Type

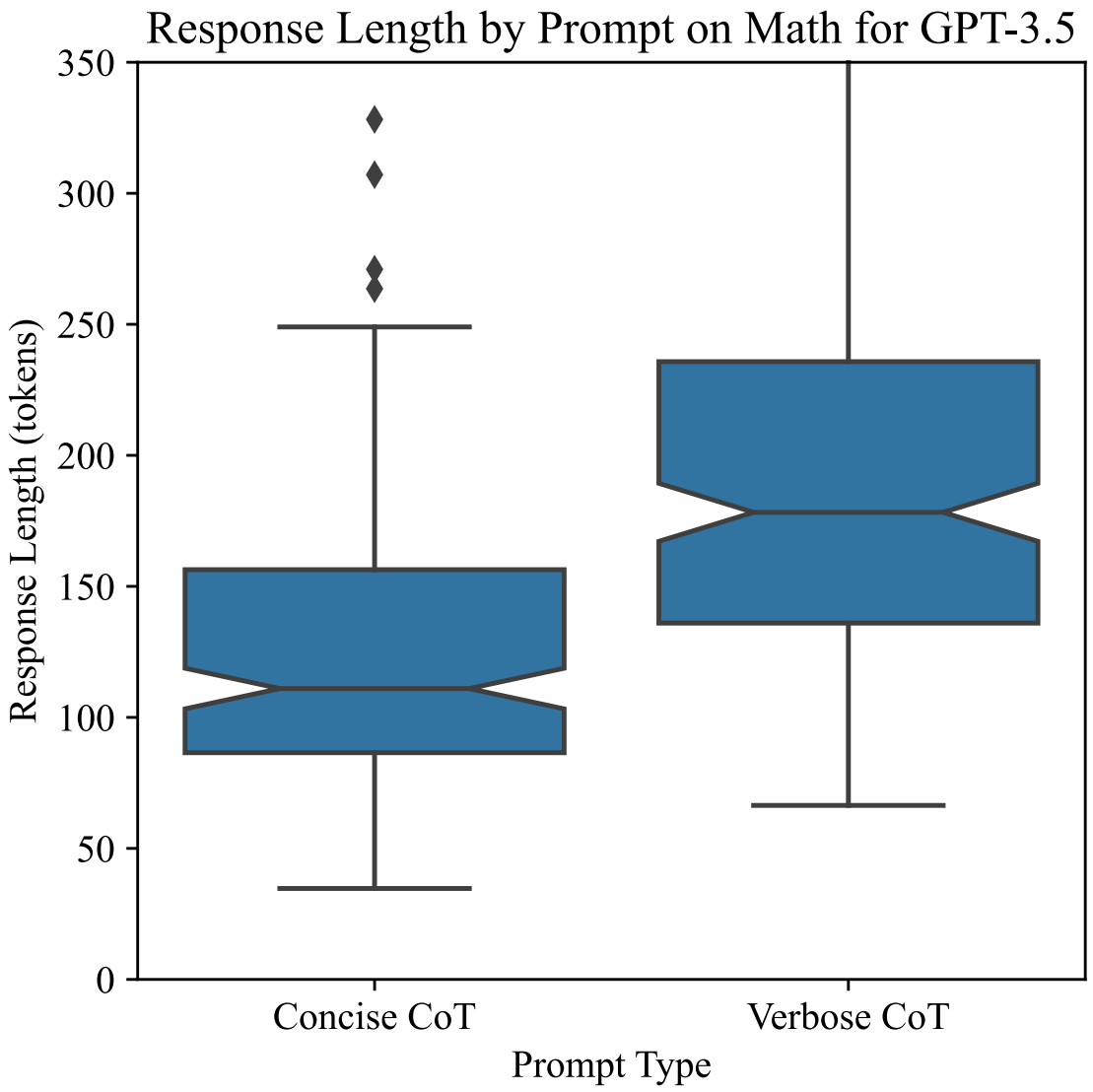

Response Length by Prompt Type and Exam for GPT-4
aqua-rat-100
arc-challenge-test-100
hellaswag_val-100
logiqa-en-100
lsat-ar-100
lsat-lr-100
lsat-rc-100
medmcqa-dev-100
sat-en-100
sat-math-100
Response Length (tokens)
0
200
Answer Only
Concise CoT
Verbose CoT
Prompt Type

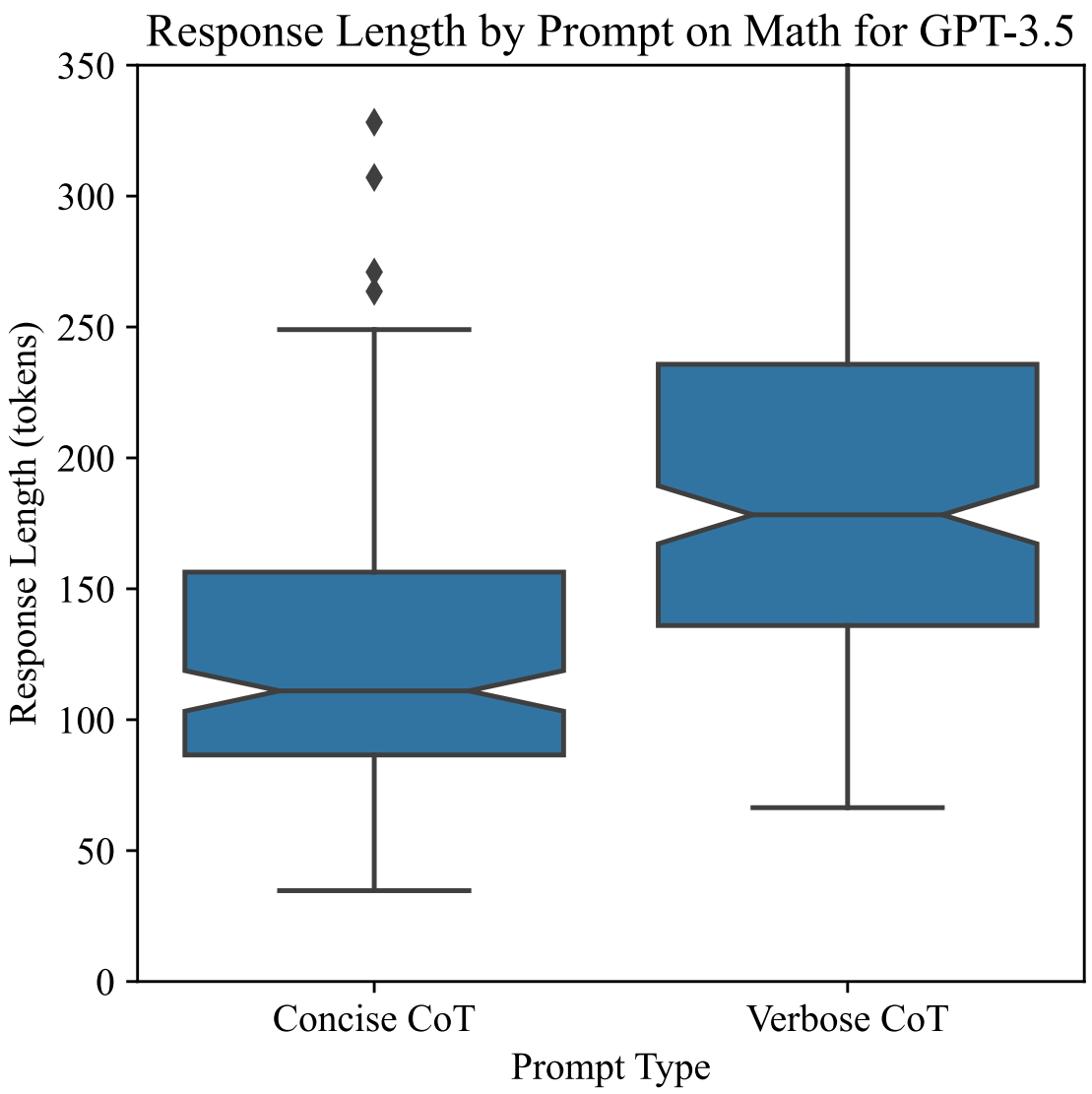

Accuracy of GPT-3.5 by Prompt Type and Exam
aqua-rat-100
arc-challenge-test-100
hellaswag_val-100
logiqa-en-100
lsat-ar-100
lsat-lr-100
lsat-rc-100
medmcqa-dev-100
sat-en-100
sat-math-100
Accuracy
1.0
0.5
0.0
Answer Only
Concise CoT
Verbose CoT
Prompt Type

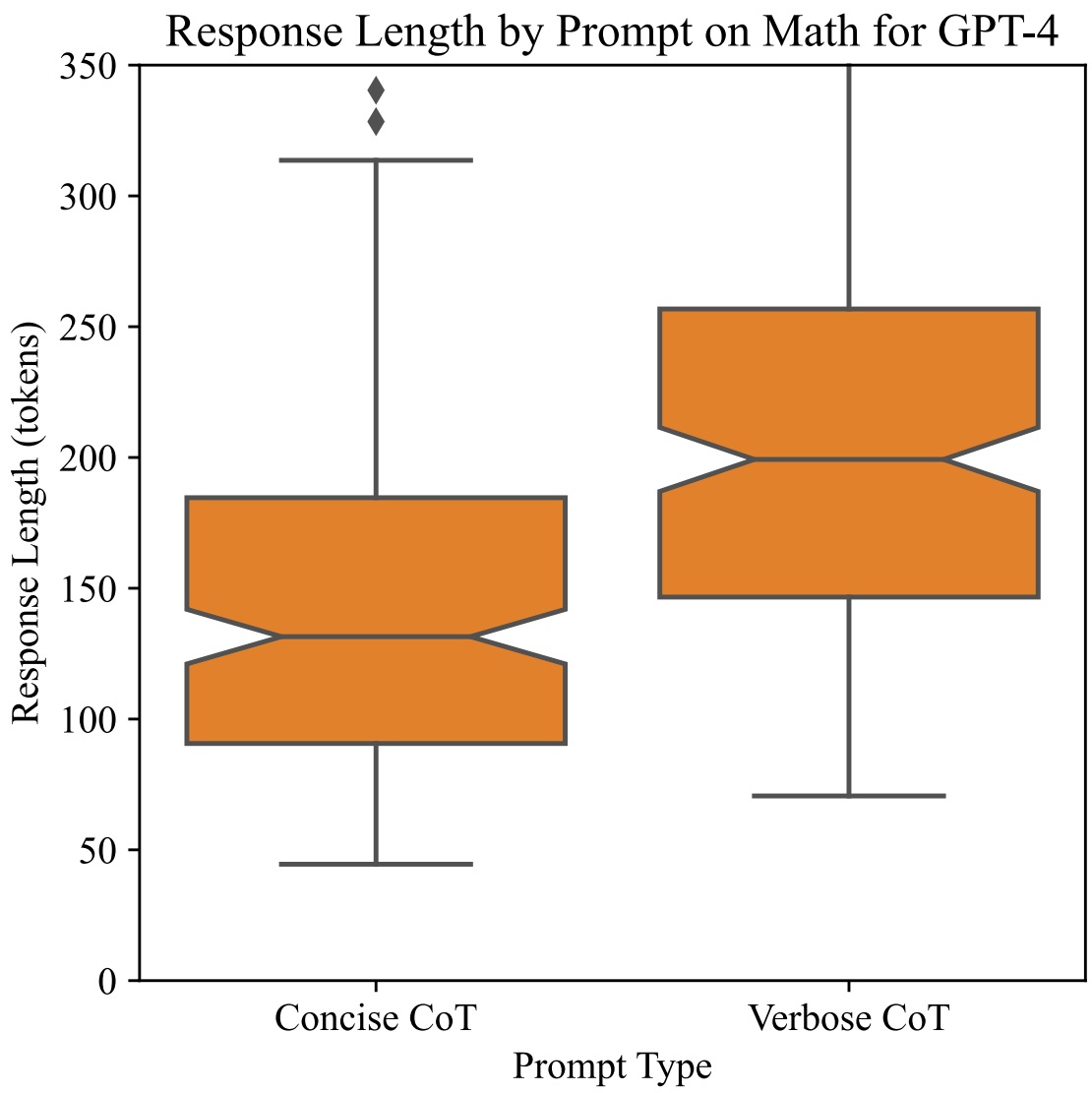

Accuracy of GPT-3.5 by Prompt Type and Exam
aqua-rat-100
arc-challenge-test-100
hellaswag_val-100
logiqa-en-100
lsat-ar-100
lsat-lr-100
lsat-rc-100
medmcqa-dev-100
sat-en-100
sat-math-100
Accuracy
1.0
0.5
0.0
Answer Only
Concise CoT
Verbose CoT

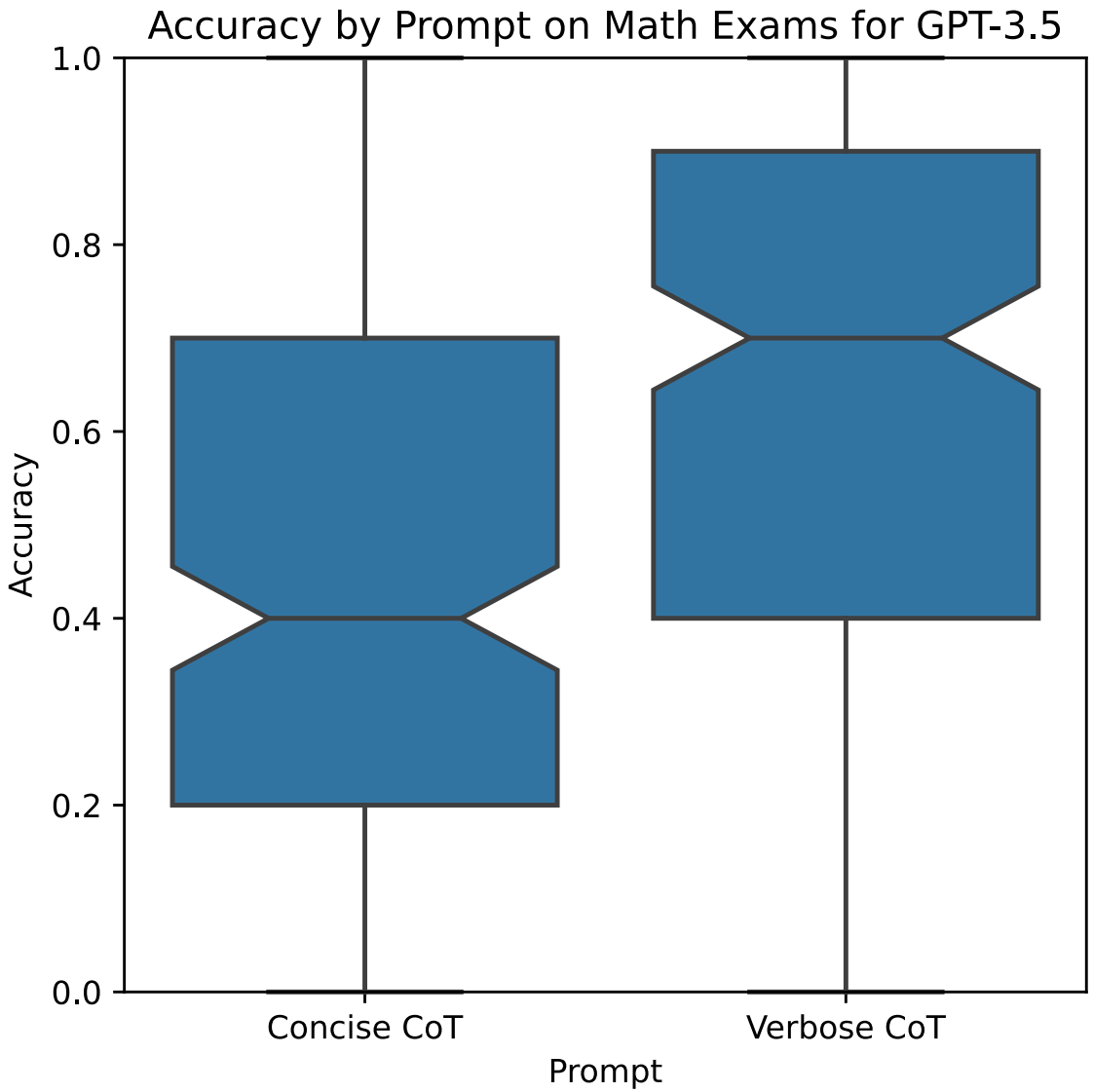

Accuracy of GPT-4 by Prompt Type and Exam
aqua-rat-100
arc-challenge-test-100
hellaswag_val-100
logiqa-en-100
lsat-ar-100
lsat-lr-100
lsat-rc-100
medmcqa-dev-100
sat-en-100
sat-math-100
Accuracy
1.0
0.5
0.0
Answer Only
Concise CoT
Verbose CoT
Prompt Type

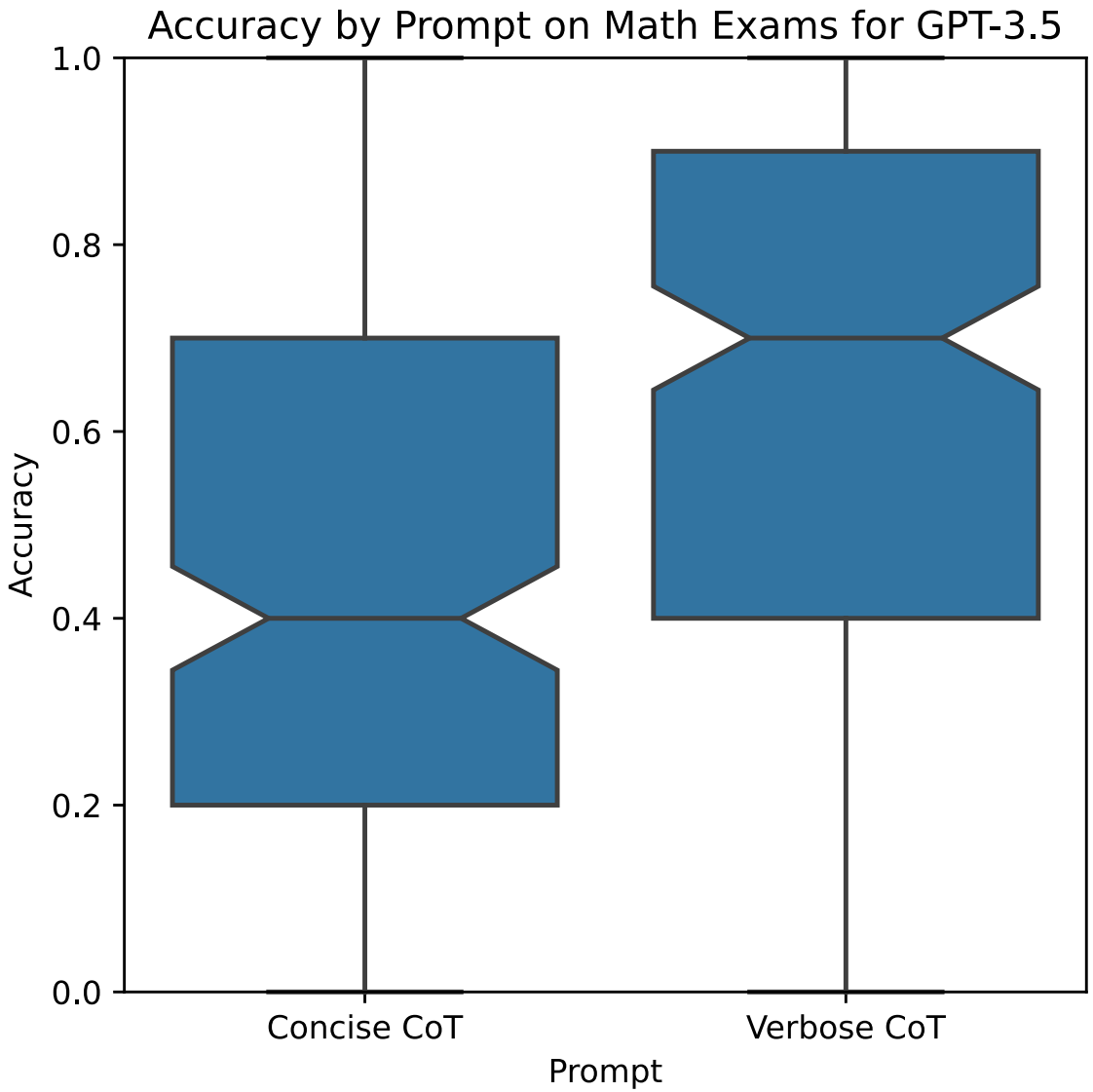

Accuracy of GPT-4 by Prompt Type and Exam
aqua-rat-100
arc-challenge-test-100
hellaswag_val-100
logiqa-en-100
lsat-ar-100
lsat-lr-100
lsat-rc-100
medmcqa-dev-100
sat-en-100
sat-math-100
Accuracy
1.0
0.5
0.0
Answer Only
Concise CoT
Verbose CoT

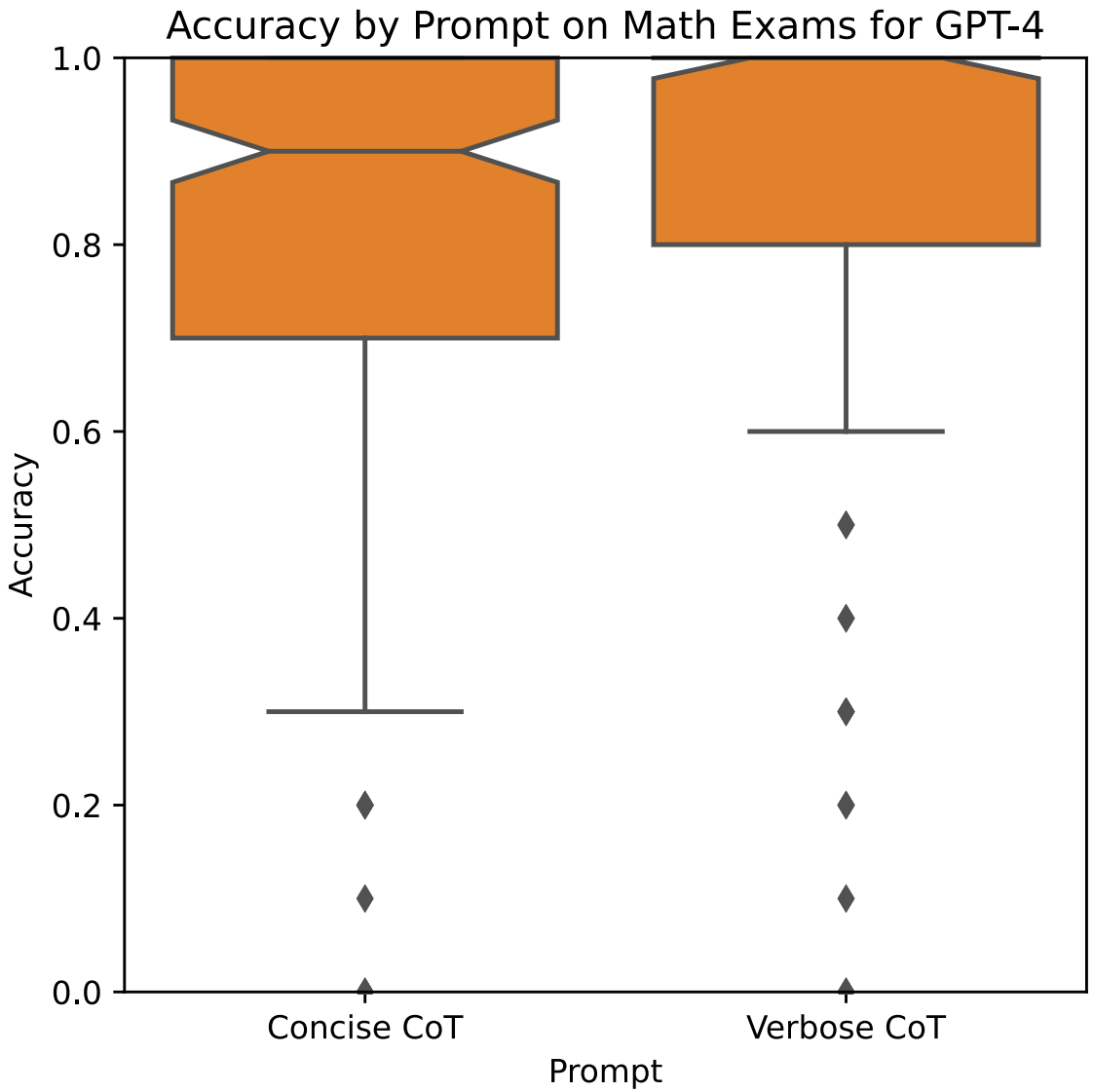

Response Length by Prompt on Math for GPT-3.5
Response Length (tokens)
350
300
250
200
150
100
50
0
Concise CoT
Verbose CoT
Prompt Type

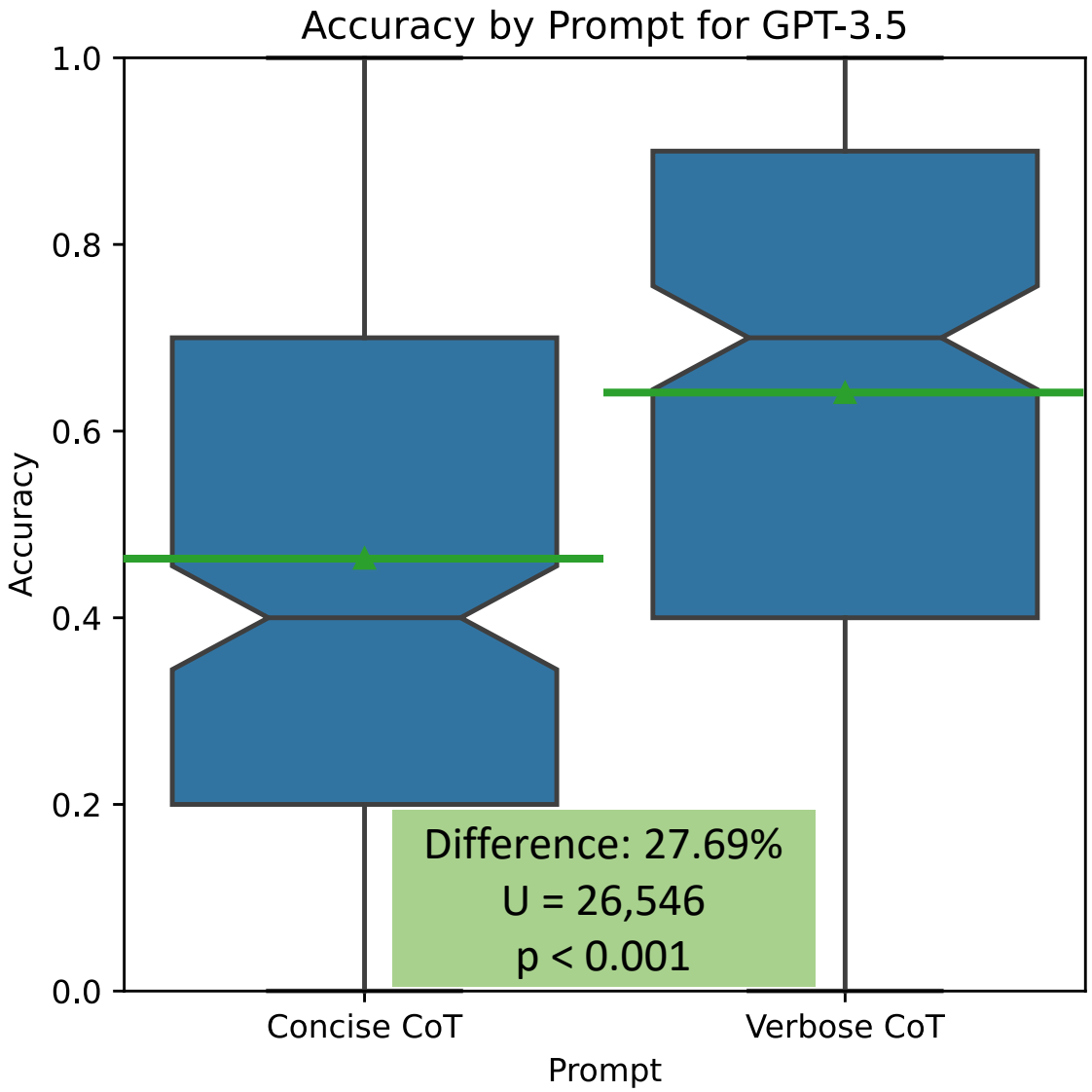

Response Length by Prompt on Math for GPT-3.5
Response Length (tokens)
350
300
250
200
150
100
50
0
Concise CoT
Verbose CoT
Prompt Type

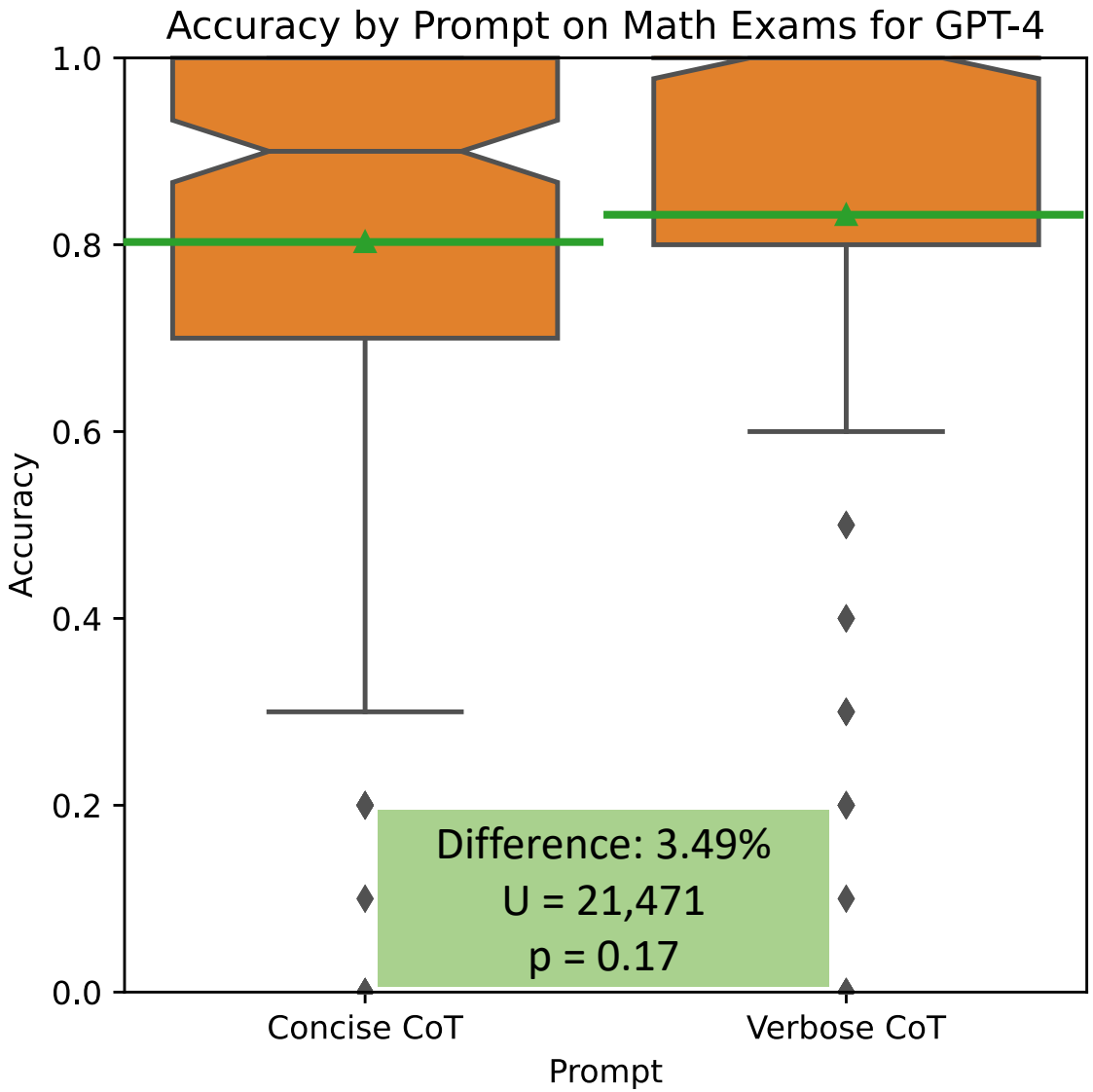

Response Length by Prompt on Math for GPT-4
Response Length (tokens)
350
300
250
200
150
100
50
0
Concise CoT
Verbose CoT
Prompt Type


Accuracy by Prompt on Math Exams for GPT-3.5
1.0
0.8
0.6
0.4
0.2
0.0
Accuracy
Concise CoT
Verbose CoT
Prompt


Accuracy by Prompt on Math Exams for GPT-3.5
1.0
0.8
0.6
0.4
0.2
0.0
Accuracy
Concise CoT
Verbose CoT
Prompt


Accuracy by Prompt on Math Exams for GPT-4
1.0
0.8
0.6
0.4
0.2
0.0
Accuracy
Concise CoT
Verbose CoT
Prompt


Accuracy by Prompt for GPT-3.5
1.0
0.8
0.6
0.4
0.2
0.0
Accuracy
Difference: 27.69%
U = 26,546
p < 0.001
Concise CoT
Verbose CoT
Prompt


Accuracy by Prompt on Math Exams for GPT-4
1.0
0.8
0.6
0.4
0.2
0.0
Accuracy
Difference: 3.49%
U = 21,471
p = 0.17
Concise CoT
Verbose CoT
Prompt

Cost per 1,000 problems

	GPT-3.5 CoT	GPT-3.5 CCoT	GPT-4 CoT	GPT-4 CCoT
Input Cost ($)	0.55	0.51	16.37	15.29
Output Cost ($)	0.33	0.17	10.53	5.29
Total Cost ($)	0.88	0.69	26.9	20.58
Cost Savings (%)		*21.85*		*23.49*

Cost per 1,000 problems

	GPT-3.5 CoT	GPT-3.5 CCoT	GPT-4 CoT	GPT-4 CCoT
Input Cost ($)	0.55	0.51	16.37	15.29
Output Cost ($)	0.33	0.17	10.53	5.29
Total Cost ($)	0.88	0.69	26.9	20.58
Cost Savings (%)		*21.85*		*23.49*

Cost per 1,000 problems

	GPT-3.5 CoT	GPT-3.5 CCoT	GPT-4 CoT	GPT-4 CCoT
Input Cost ($)	0.55	0.51	16.37	15.29
Output Cost ($)	0.33	0.17	10.53	5.29
Total Cost ($)	0.88	0.69	26.9	20.58
Cost Savings (%)		*21.85*		*23.49*

Discussion

Limitations

Limitations

Only 2 LLMs

Limitations

Only 2 LLMs

Only 3 prompts

Limitations

Only 2 LLMs

Only 3 prompts

Only 10 domains

Limitations

Only 2 LLMs

Only 3 prompts

Only 10 domains

Possible ceiling effect

Future Research

More models

More prompts

More domains

More in-depth analysis

Concise Thoughts: Impact of Output Length on LLM Reasoning and Cost

Sania Nayab [1] **Giulio Rossolini** [1] **Giorgio Buttazzo** [1] **Nicolamaria Manes** [2] **Fabrizio Giacomelli** [2]

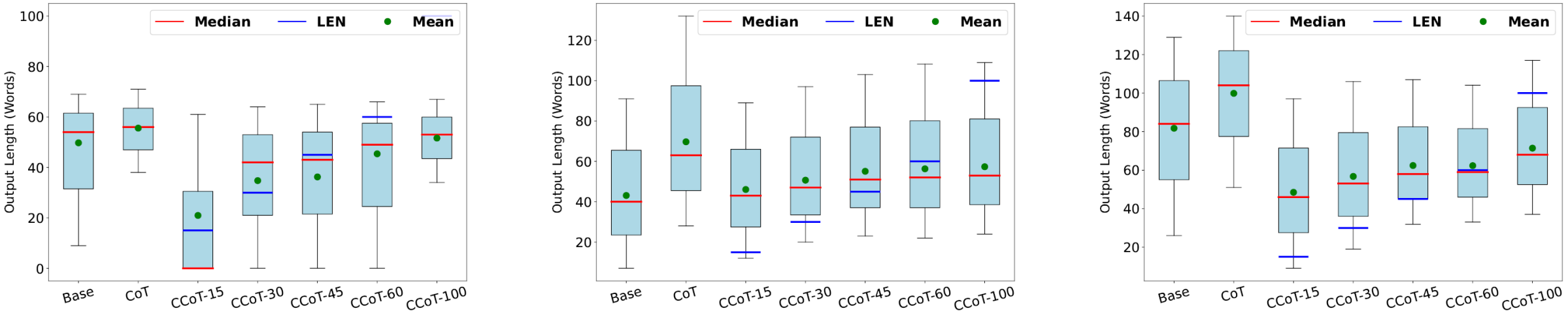
Abstract

Today's large language models (LLMs) can solve challenging question-answering tasks, and prompt engineering techniques, such as chain-of-thought (CoT), have gained attention for enhancing the explanation and correctness of outputs. Nevertheless, models require significant time to generate answers augmented with lengthy reasoning details. To address this issue, this paper analyzes the impact of output lengths on LLM inference pipelines and proposes novel metrics to evaluate them in terms of *correct conciseness*. It also examines the impact of controlling output length through a refined prompt engineering strategy, Constrained-CoT (CCoT), which encourages the model to limit output length. Experiments on pre-trained LLMs demonstrated the benefit of the proposed metrics and the effectiveness of CCoT across different models. For instance, constraining the reasoning of LLaMA2-70b to 100 words improves the accuracy from 36.01% (CoT) to 41.07% (CCoT) on the GSM8K dataset, while reducing the average output length by 28 words.

answer through intermediate reasoning steps.

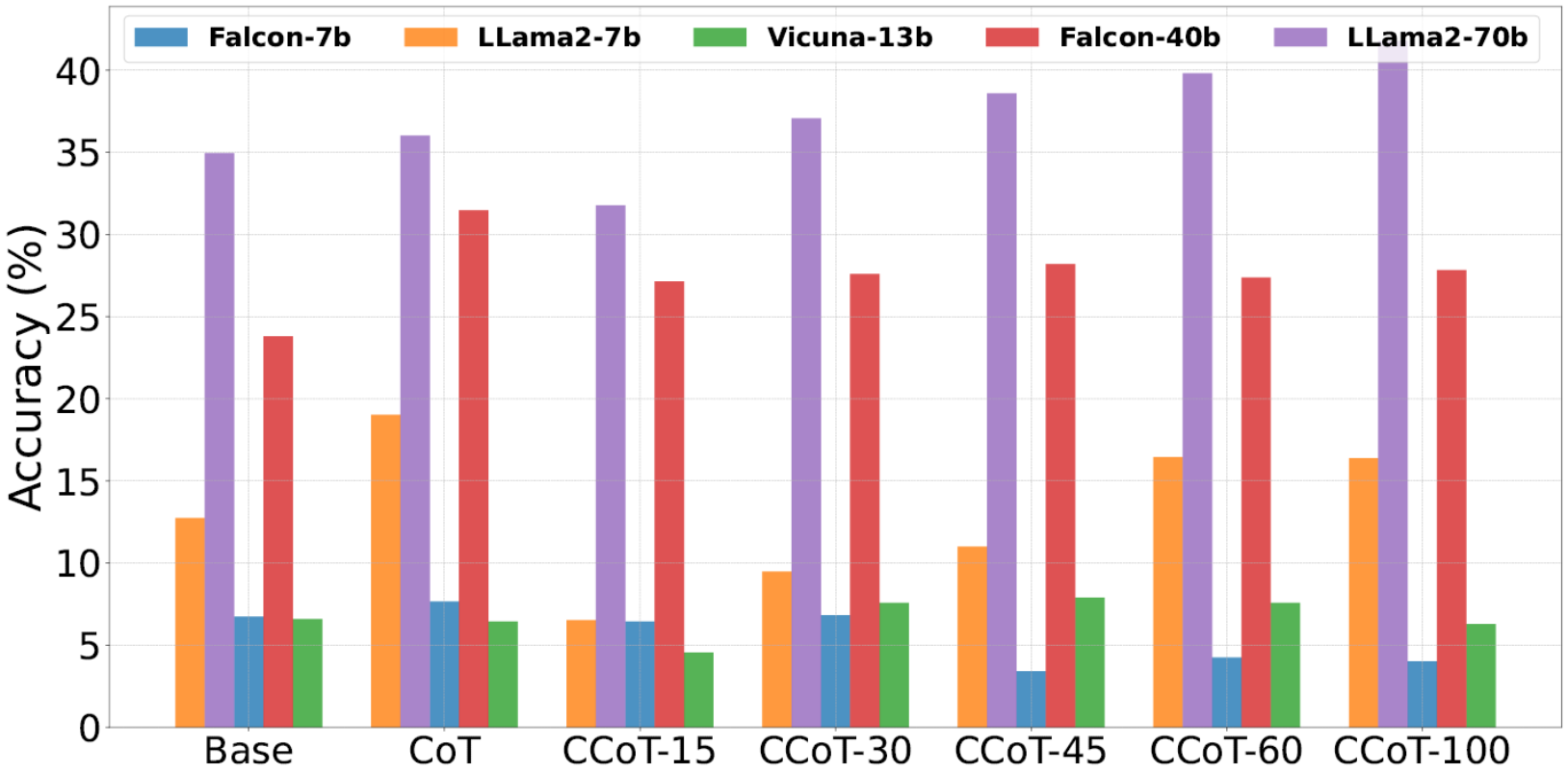
Despite the mentioned advantages, the CoT prompting can lead to longer outputs, increasing the time required for the model to generate a response. This is due to the nature of autoregressive transformers, which decode text word by word, each time running a new inference pass of the decoder module (Vaswani et al., 2017; Shekhar et al., 2024). This implies that the time required to generate a response is heavily influenced by the length of the reasoning provided, which can also vary depending on the prompt. Such long and variable delays in the responses are undesirable when the LLM has to relate with a user through an interactive conversation. This issue highlights the need to consider i) metrics for evaluating the conciseness of the outputs and ii) solutions to avoid excessively long chains of reasoning.

To this end, the first part of this work presents some motivational experiments to show the relation between output length and inference time of an LLM. Then, it proposes three novel metrics to account for the conciseness and correctness of a generated answer. The objective of the proposed metrics is to reweight the accuracy of a given model by considering aspects related to output lengths that affect the inference time of the model and its time predictability.



(a) Vicuna-13b (b) Falcon 40b (c) Llama2-70b

Figure 5. Distribution (between the 5th and 95th percentiles) of the output lengths (y-axis) given by different models and prompting strategies with the GSM8K test set.



(b) Accuracy

Source: https://arxiv.org/abs/2407.19825

Conclusion

Conclusion

Conclusion

CCoT reduces response token length by 48.70%

GPT-4 incurred no performance penalty

GPT-3.5 had 27.69% penalty on math

Reduces cost by 22.67%

Conclusion

Use Concise CoT for GPT-4

Don’t use for GPT-3.5

Need to test others

Learn more

https://matthewrenze.com/research/the-benefits-of-a-concise-chain-of-thought/