



Evaluating the Proficiency of ChatGPT in Undergraduate Data Structures and Algorithms

An Analysis of Standardized Test Performance



By - Mokshith Ramendra Yaganti

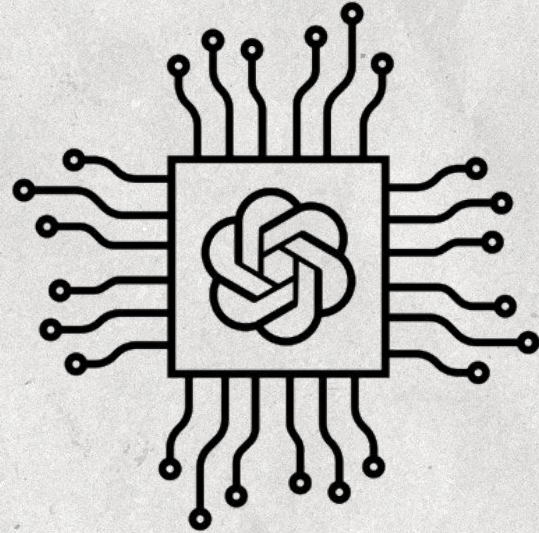


Contents

Context	The Importance of DSA in AI Code Generation
Challenges	Variability in Code Efficiency and Correctness
Questions	Enhancing ChatGPT's DSA Performance
Hypothesis	The Impact of Prompt Engineering and Automated Evaluation
Design/Methodology	The design of the pipeline and Methodology
Data	Understanding the Input/Output and distribution of Data used.
Metrics	The key metrics used for performance evaluation
Results/Analysis	Results for each prompt and an analysis of performance
Limitations	Factors which limited the current study
Future scope	Actionable goals to expand the study
Appendix	Figures of topicwise performance of ChatGPT for every prompt - in order from 1-9

The Backbone of AI-Driven Software Development

- Large Language Models (LLMs) like ChatGPT have revolutionized code generation, making strides in software development.
- Data Structures and Algorithms (DSA) are pivotal for efficient code creation, affecting execution speed and resource management.
- ChatGPT's proficiency in DSA underlines its potential as a powerful tool in software development, from problem analysis to creating algorithmic solutions.



ChatGPT





Challenges in AI-Generated Code Quality

1

Discussion 1

Despite ChatGPT's capabilities, there's a noticeable challenge in consistently generating efficient and correct code, especially in complex DSA tasks.

2

Discussion 2

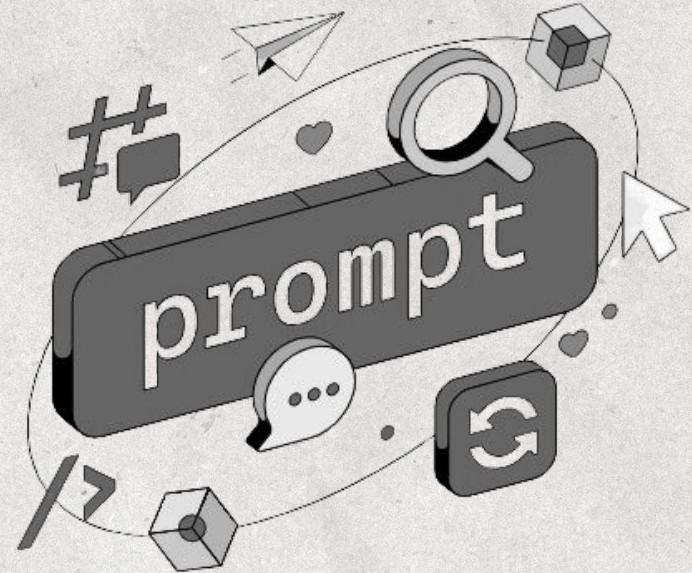
Variability in output quality highlights the need for refined interaction techniques to elicit high-quality code, addressing issues from minor bugs to system failures.



Traditional methods of evaluating AI-generated code often rely on manual strategies, limiting the scope and scalability of assessments.

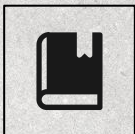
The Quest for Optimal AI-Generated Solutions

- How can prompt engineering influence ChatGPT's performance in solving undergraduate-level DSA problems?
- The study focuses on whether strategic prompt design can lead to more accurate, efficient, and optimized solutions in code generation.
- Investigates the potential of automating the study of ChatGPT's DSA code generation capabilities for a scalable and rigorous evaluation.





The Potential of Tailored Prompts and Automation



Hypothesis 1

The hypothesis posits that through strategic prompt engineering, ChatGPT's ability to generate correct and efficient DSA solutions can be significantly enhanced.

Employing an automated system for evaluating ChatGPT's code generation will offer a more scalable and objective assessment compared to traditional manual methods.

Hypothesis 2



This research aims to empirically test this hypothesis, exploring the relationship between prompt specificity, automated evaluation, and the quality of generated code solutions.



Design

01

Fetch

Fetch questions, code stubs and test cases.

02

Prompt

Create a set of prompts and leverage it as context for ChatGPT.

03

Generate

ChatGPT generates a solution given the prompt and prepared code stub

04





Evaluate

Generated solutions are tested against the test-cases.

Pipeline



Methodology

Methods		
	Fetch	A custom scraper was created to make an API call to the graphql db behind leetcode.com and fetch DSA questions along with the provided code stub and constraints. The test cases were also fetched and re-formatted as a pytest method.
	Prompt	A set of 9 unique prompts are created which will be used as context for ChatGPT. The code stub and its corresponding question and constraints are prepared as one prompt string.
	Generate	Given the context (prompt) and its corresponding question, an API request is made to ChatGPT to generate the solution.
	Evaluate	Each solution is tested against its test-cases using pytest and the results are recorded for further performance analysis.



Data

Input		Output
Fetches	Processed	<p><u>Solution:</u> For each question a corresponding solution file is generated using the response from ChatGPT.</p> <p><u>Evaluation:</u> Each solution file is tested and the evaluation results are recorded in a .csv file.</p>
<ul style="list-style-type: none">A total of ~550 standard DSA questions & constraints.Code stub provided by leetcode.Test cases provided as a pytest module.	Prompts	
	A set of 9 prompts carefully crafted to elicit various levels of expertise and solving capabilities.	
	Final Input	
Context: Prompt ChatGPT prompt: Code stub, question, constraints as one string.		



Data contd.

The questions are divided into the following categories for a diversified analysis of ChatGPT's performance.

- **Easy** - basic questions.
- **Medium** - Advanced DSA or basic questions with harder twists.
- **Hard** - Advanced DSA and advanced questions with harder twists.

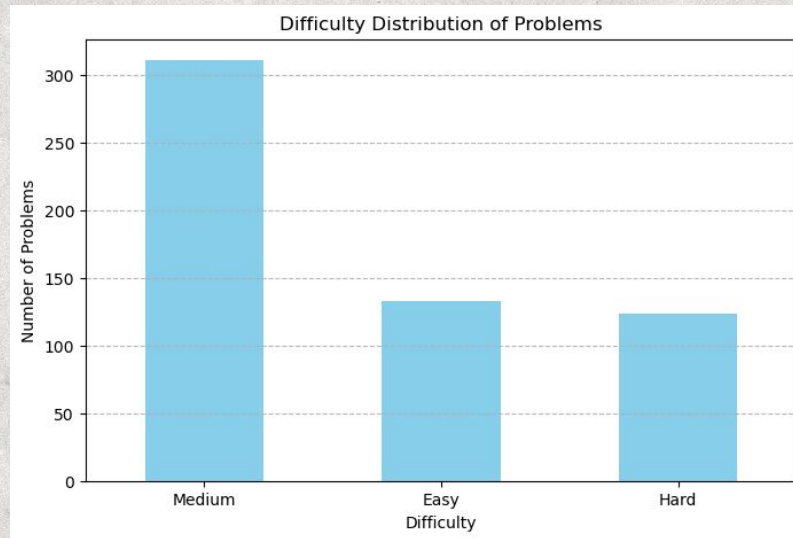


Figure: This figure shows the distribution of the number of easy, medium and hard category questions. No. of questions for medium category is higher in aims for testing ChatGPT's performance on a normal distribution of what a student might encounter in a real life coding round.



Metrics

The solutions generated by ChatGPT 3.5 are evaluated using the following metrics to establish a key benchmarking performance on standard undergraduate DSA questions.

Pass	Fail	Error	<u>Acceptance</u>
<p>This metric indicates if the solution generated passes the test cases.</p> <p>Ex: The solution passes 2/3 test cases, then pass = 2</p>	<p>This metric indicates if the solution generated fails the test cases.</p> <p>Ex: The solution passes 2/3 test cases, then fail = 1</p>	<p>This metric indicates if the solution generated fails the test cases because of an error produced by the code.</p> <p>Ex: The solution passes 2/3 test cases & fails 1 because of an error, then Error = 1</p>	<p>This is the <u>key</u> metric of choice which indicates if a generated solution passes all the test cases without any errors.</p> <p>Ex: The solution passes 3/3 test cases without any errors, then Acceptance = yes.</p>



Results

For prompt 1: “Your role as an expert in Data Structures and Algorithms involves providing strictly accurate Python solutions. Provide only the code, with no extra text”

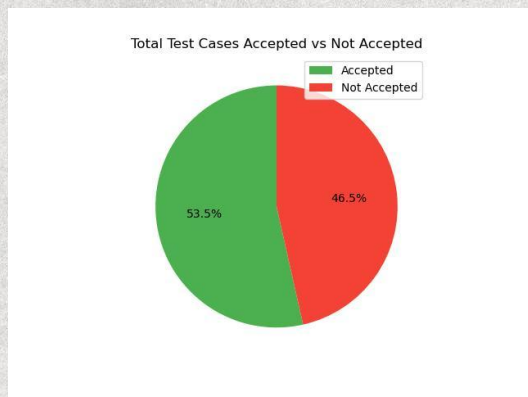


Figure: This figure shows the total percentage of the generated solutions accepted vs not accepted against their corresponding test-cases. This figure is also the key benchmark factor in evaluating the performance effects of each of the 9 prompts.

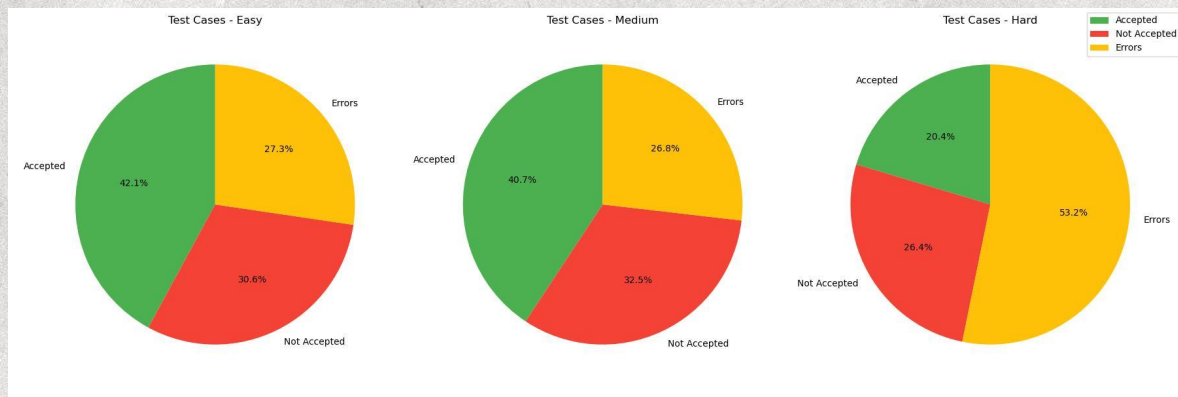


Figure: This figure shows the percentage of all the metrics for each of the categories mentioned earlier. It showcases how good ChatGPT is corresponding to the difficulty level of the question set.



Results contd.

For prompt 2: “You, a Python expert specializing in data structures and algorithms, are tasked with writing a function based on the stub, strictly following the problem's specifications for optimal time and space usage. Avoid any additional text, focusing solely on the code.”

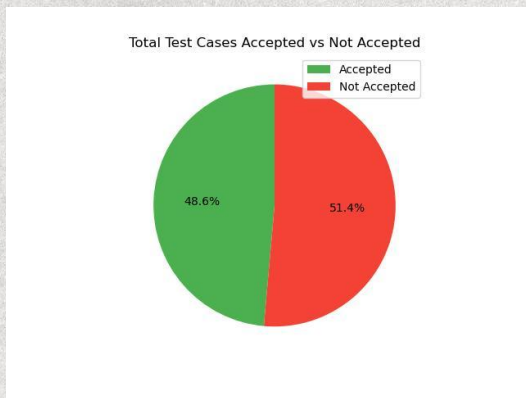


Figure: This figure shows the total percentage of the generated solutions accepted vs not accepted against their corresponding test-cases. This figure is also the key benchmark factor in evaluating the performance effects of each of the 9 prompts.

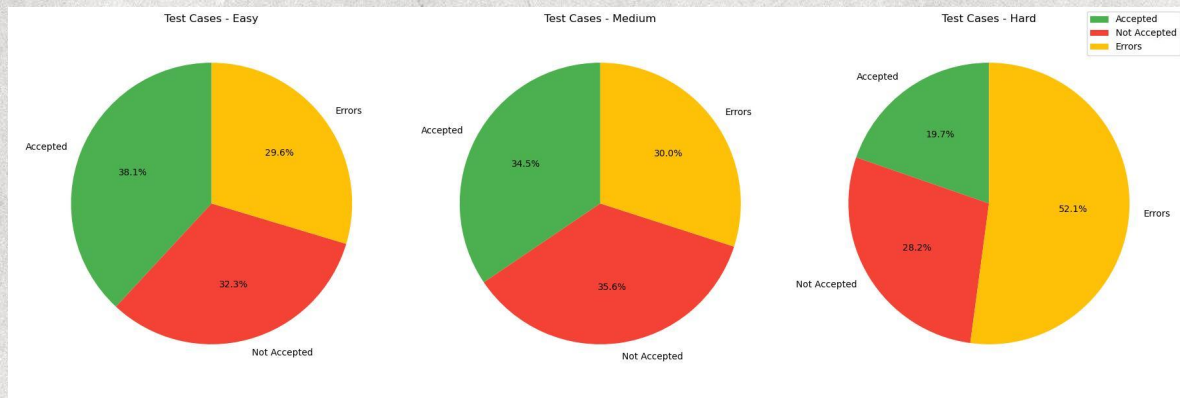


Figure: This figure shows the percentage of all the metrics for each of the categories mentioned earlier. It showcases how good ChatGPT is corresponding to the difficulty level of the question set.



Results contd.

For prompt 3: “You are an expert in Data Structures and Algorithms & you only give accurate solutions in python. You do not generate any additional text apart from the code”

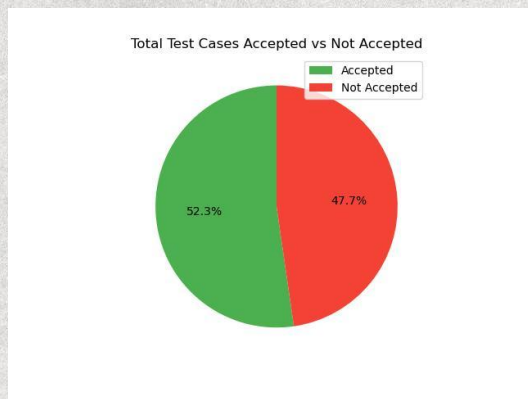


Figure: This figure shows the total percentage of the generated solutions accepted vs not accepted against their corresponding test-cases. This figure is also the key benchmark factor in evaluating the performance effects of each of the 9 prompts.

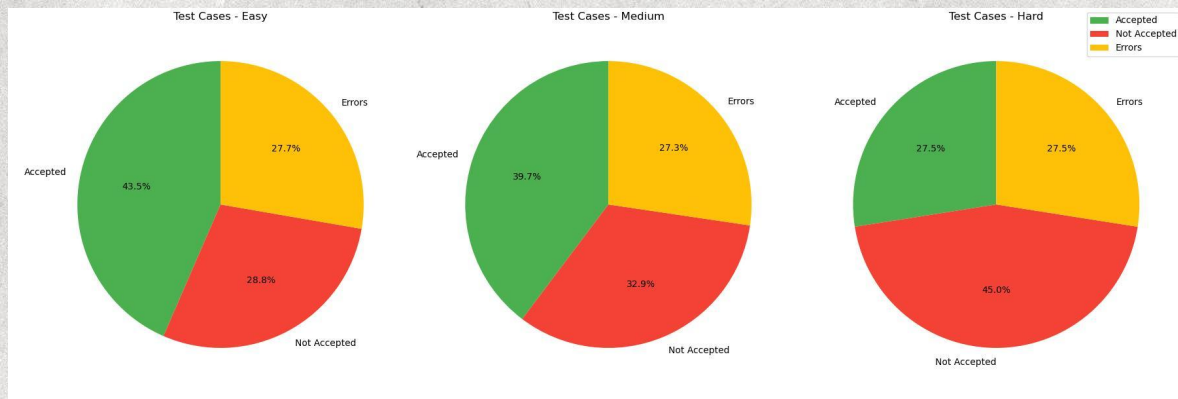


Figure: This figure shows the percentage of all the metrics for each of the categories mentioned earlier. It showcases how good ChatGPT is corresponding to the difficulty level of the question set.



Results contd.

For prompt 4: “You are an expert Python developer focused on data structures and algorithms. Write an efficient Python function as provided in the code stub that adheres strictly to the problem's specifications and optimizes for both time and space. You do not generate any additional text or characters apart from the code”

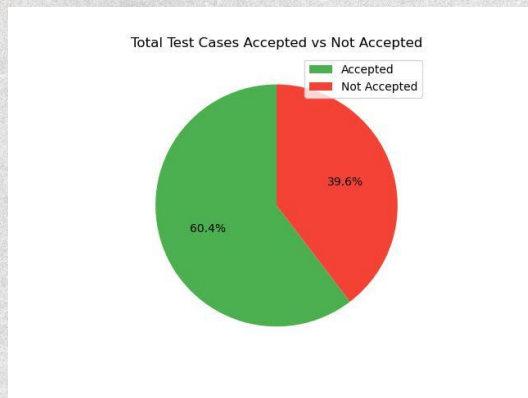


Figure: This figure shows the total percentage of the generated solutions accepted vs not accepted against their corresponding test-cases. This figure is also the key benchmark factor in evaluating the performance effects of each of the 9 prompts.

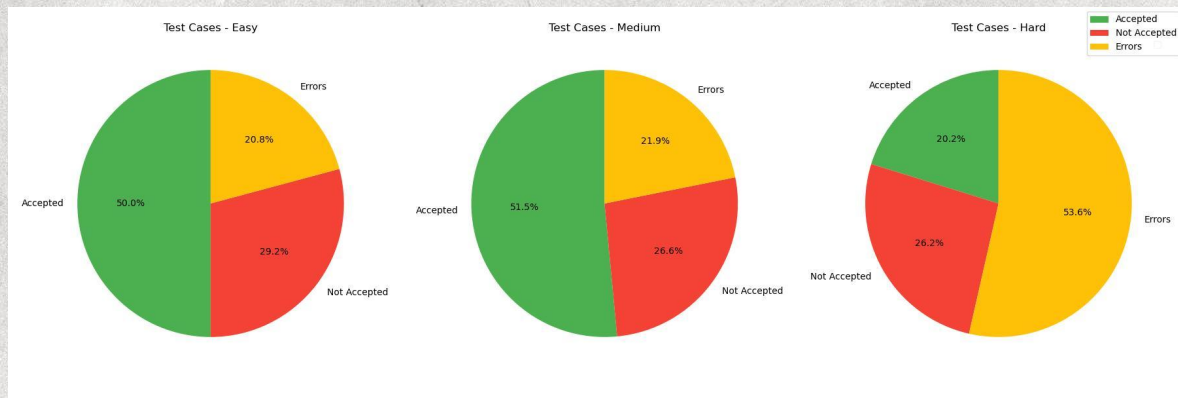


Figure: This figure shows the percentage of all the metrics for each of the categories mentioned earlier. It showcases how good ChatGPT is corresponding to the difficulty level of the question set.



Results contd.

For prompt 5: "As a professional Python programmer specializing in data structures and algorithms, implement a solution for this LeetCode problem using the exact code signature provided. Ensure your code handles all constraints, edge cases and is optimized for performance. You do not generate any additional text apart from the code."

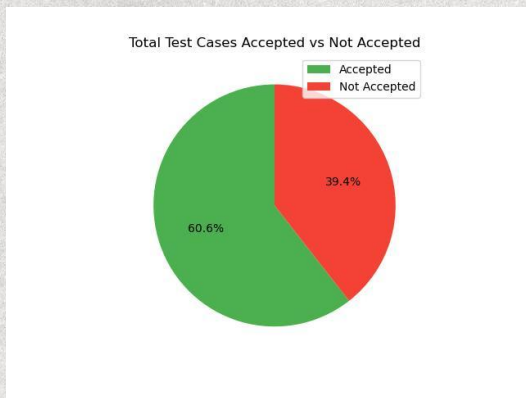


Figure: This figure shows the total percentage of the generated solutions accepted vs not accepted against their corresponding test-cases. This figure is also the key benchmark factor in evaluating the performance effects of each of the 9 prompts.

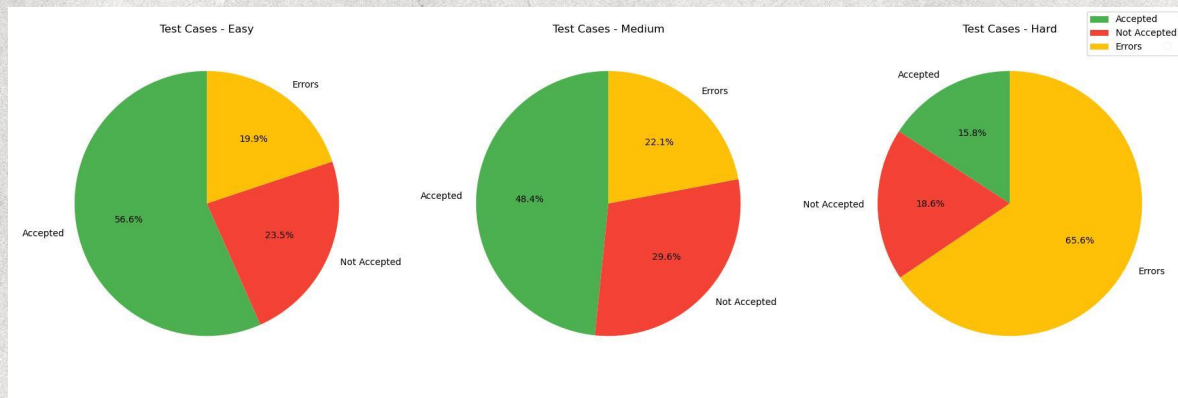


Figure: This figure shows the percentage of all the metrics for each of the categories mentioned earlier. It showcases how good ChatGPT is corresponding to the difficulty level of the question set.



Results contd.

For prompt 6: “You are tasked as a Python software engineer to develop code solving this algorithmic challenge. Write the cleanest and most efficient code, considering all given constraints and using the specified function names. You do not generate any additional text apart from the code”

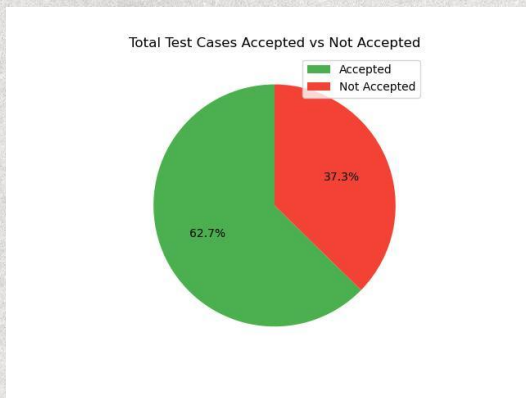


Figure: This figure shows the total percentage of the generated solutions accepted vs not accepted against their corresponding test-cases. This figure is also the key benchmark factor in evaluating the performance effects of each of the 9 prompts.

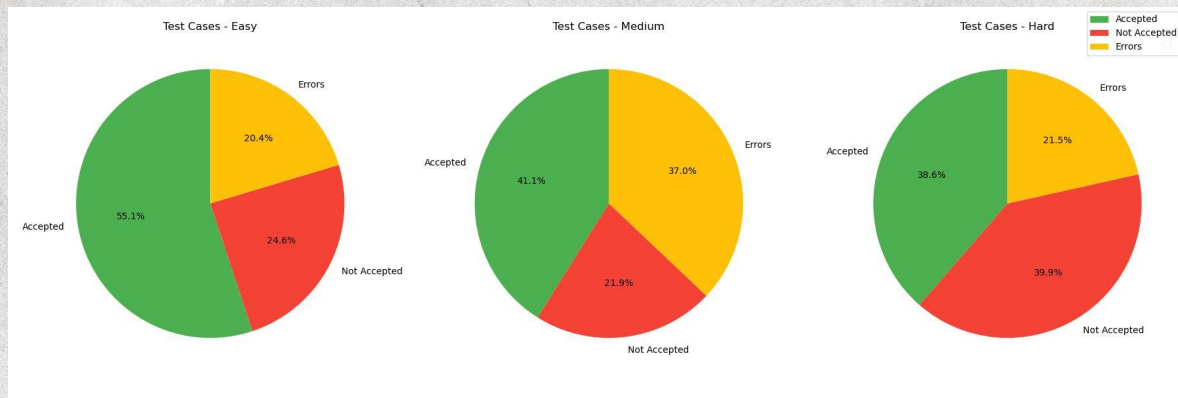


Figure: This figure shows the percentage of all the metrics for each of the categories mentioned earlier. It showcases how good ChatGPT is corresponding to the difficulty level of the question set.



Results contd.

For prompt 7: “As a junior programmer, attempt to solve this problem in Python. Use what you’ve learned so far about data structures and algorithms. You do not generate any additional text apart from the code”

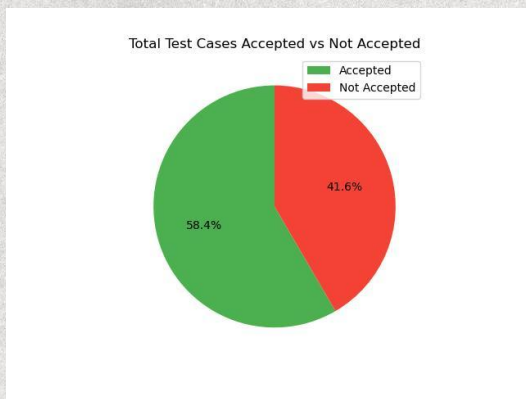


Figure: This figure shows the total percentage of the generated solutions accepted vs not accepted against their corresponding test-cases. This figure is also the key benchmark factor in evaluating the performance effects of each of the 9 prompts.

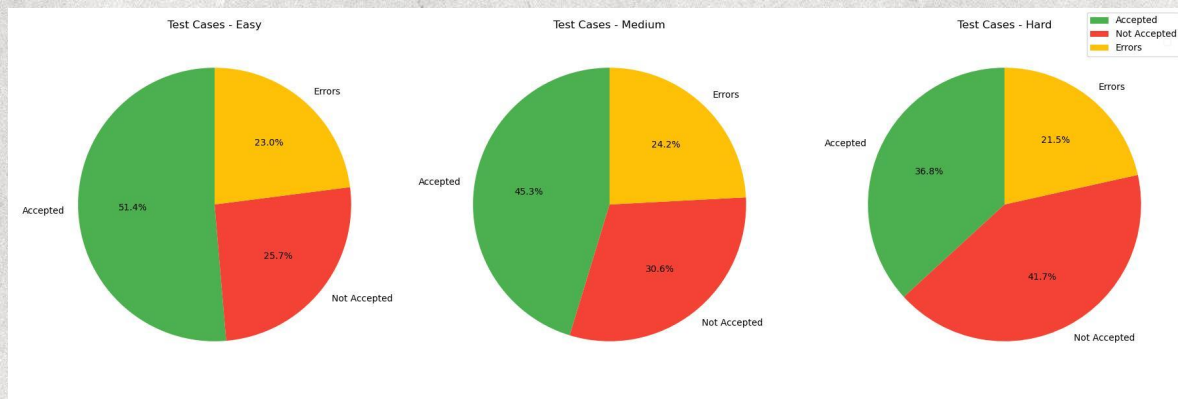


Figure: This figure shows the percentage of all the metrics for each of the categories mentioned earlier. It showcases how good ChatGPT is corresponding to the difficulty level of the question set.



Results contd.

For prompt 8: “You are a software engineering student. Please try to write Python code to solve this problem based on your current knowledge. You do not generate any additional text apart from the code”

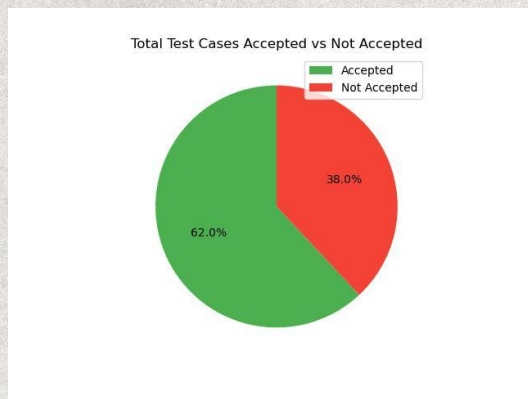


Figure: This figure shows the total percentage of the generated solutions accepted vs not accepted against their corresponding test-cases. This figure is also the key benchmark factor in evaluating the performance effects of each of the 9 prompts.

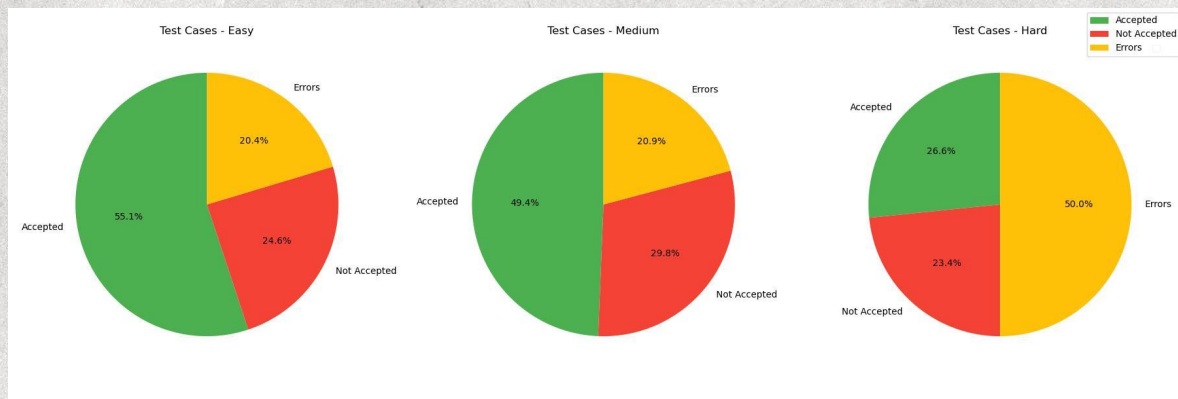


Figure: This figure shows the percentage of all the metrics for each of the categories mentioned earlier. It showcases how good ChatGPT is corresponding to the difficulty level of the question set.



Results contd.

For prompt 9: “As an enthusiastic hobbyist coder, draft a Python script to tackle this algorithm problem efficiently. You do not generate any additional text apart from the code.”

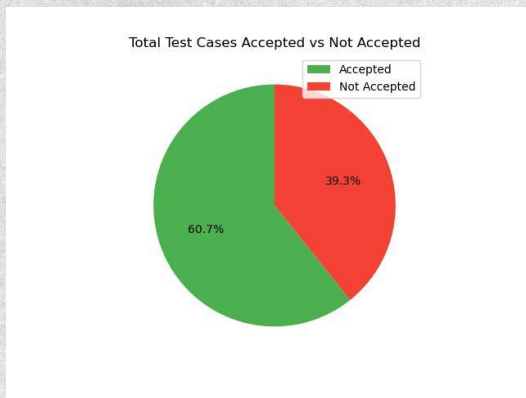


Figure: This figure shows the total percentage of the generated solutions accepted vs not accepted against their corresponding test-cases. This figure is also the key benchmark factor in evaluating the performance effects of each of the 9 prompts.

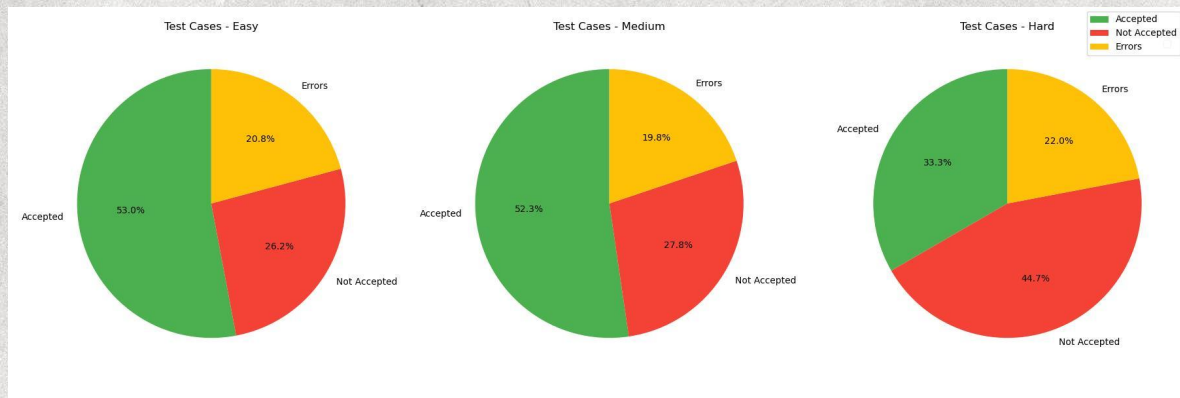


Figure: This figure shows the percentage of all the metrics for each of the categories mentioned earlier. It showcases how good ChatGPT is corresponding to the difficulty level of the question set.



Analysis

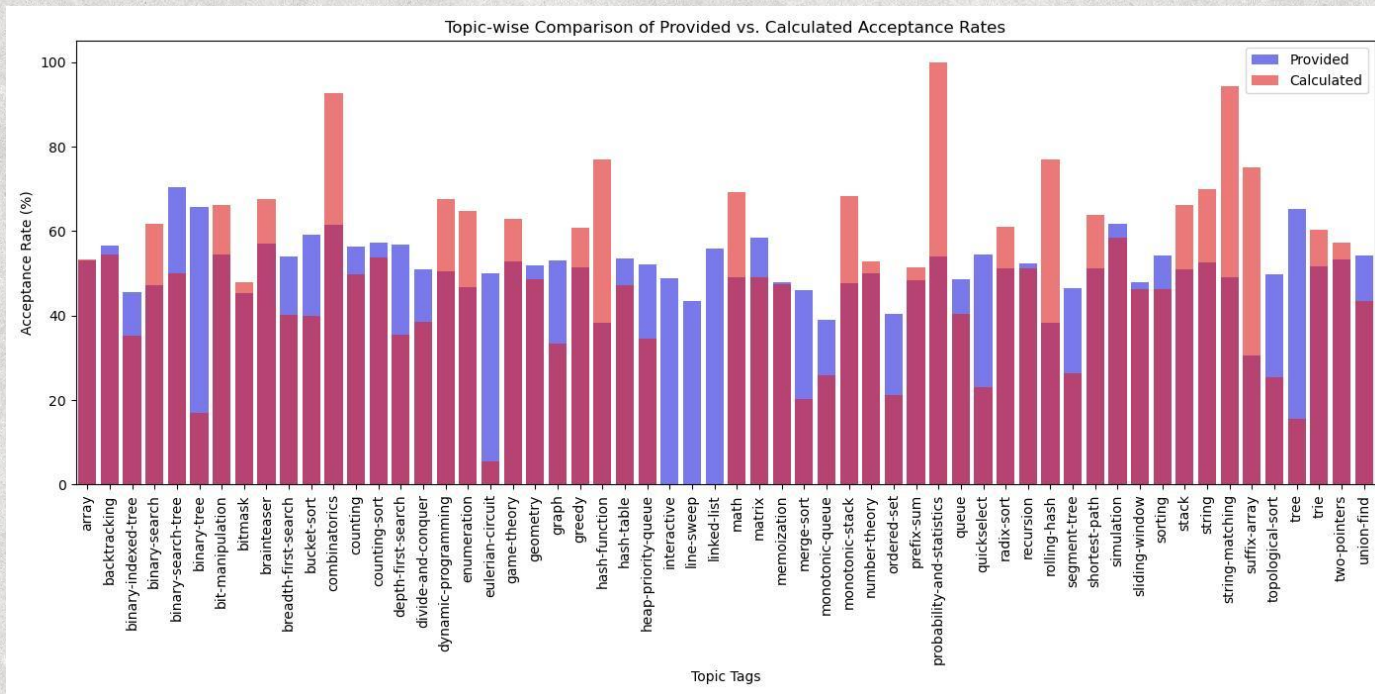


Figure: This figure shows how good/bad the performance of ChatGPT is with respect to the average acceptance rate of the general public's solutions (provided by leetcode) per topic. It clearly indicates the topics ChatGPT 3.5 exceeds humans by a huge margin and also the topics where it lacks severely. This figure provides a key average understanding of the performance of ChatGPT on DSA questions overall and also shows the specific scope for improvement.



Analytics contd.

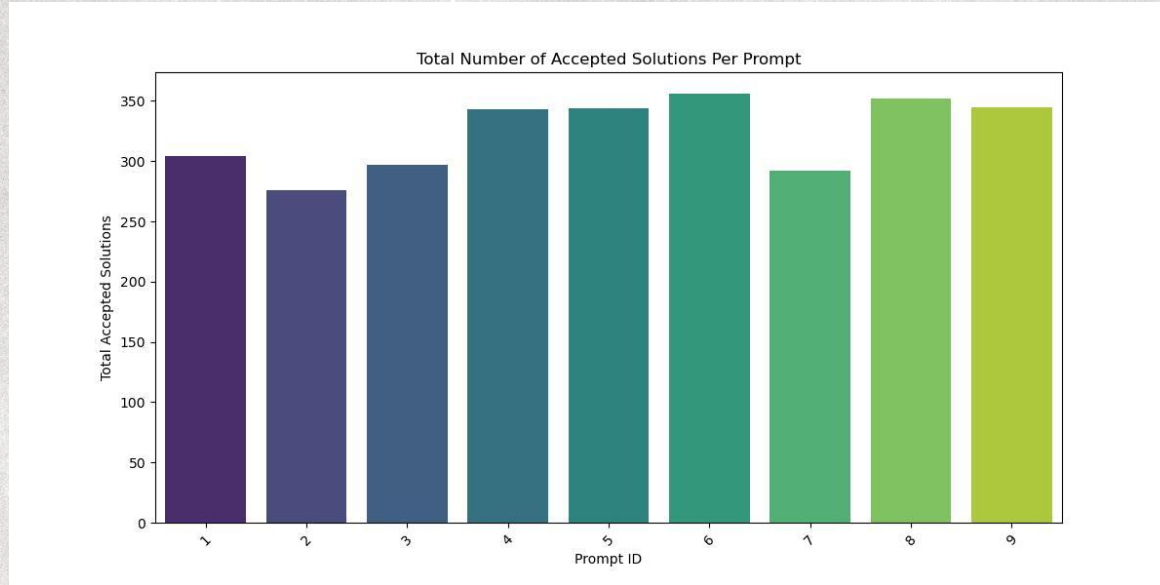


Figure: This figure shows the performance effect of each of the 9 prompts on the number of solutions accepted. It clearly shows that prompt number 6 is the best performing prompt and prompt number 2 being the worst of the bunch. This analysis also helps understand the various prompt verbage and its effect in enhancing performance in the projects context.



Limitations

- ChatGPT's API allows approximately 550 questions to be processed with the same context. This effectively sets a hard limit on the number of questions a certain prompt can be evaluated on.
- Leetcode currently does not have a full end-to-end API system to request questions, code stubs, constraints and test cases. Therefore querying the graphql behind leetcode was written from scratch to obtain the desired information.
- Additionally, since there is no functionality for submitting the ChatGPT generated solutions directly to leetcode, only the visible test-cases were fetched and used for evaluation, limiting the study to not include the comprehensive list of hidden test-cases.
- ChatGPT 3.5 was used to conduct the study as the cost of making API calls to ChatGPT 4 is budget limiting.



Future Scope

- This study can be expanded to include the performance analysis of a greater number of context prompts (~50) to truly evaluate the effect of prompting in performance boosting.
- Larger context prompts can potentially improve the throughput of good solutions and must be tested.
- LLM's such as Llama 3, Claude & other open source models can also be evaluated to obtain a comprehensive view of the performance of these models on DSA questions.



Thanks!

Do you have any questions?

myaganti@leomail.tamuc.edu



Appendix

