



**UNIVERSIDADE FEDERAL DE CAMPINA GRANDE
CENTRO DE ENGENHARIA ELÉTRICA E INFORMÁTICA
CURSO DE BACHARELADO EM CIÊNCIA DA COMPUTAÇÃO**

DÉBORA LÊDA DE LUCENA SOUZA

**ESTUDO DE CASO: USO DO CHATGPT PARA RESOLUÇÃO DE
PROBLEMAS DE PROGRAMAÇÃO**

CAMPINA GRANDE - PB

2023

DÉBORA LÊDA DE LUCENA SOUZA

**ESTUDO DE CASO: USO DO CHATGPT PARA RESOLUÇÃO DE
PROBLEMAS DE PROGRAMAÇÃO**

**Trabalho de Conclusão Curso
apresentado ao Curso Bacharelado em
Ciência da Computação do Centro de
Engenharia Elétrica e Informática da
Universidade Federal de Campina
Grande, como requisito parcial para
obtenção do título de Bacharel em
Ciência da Computação.**

Orientador : Rohit Gheyi

CAMPINA GRANDE - PB

2023

DÉBORA LÊDA DE LUCENA SOUZA

**ESTUDO DE CASO: USO DO CHATGPT PARA RESOLUÇÃO DE
PROBLEMAS DE PROGRAMAÇÃO**

**Trabalho de Conclusão Curso
apresentado ao Curso Bacharelado em
Ciência da Computação do Centro de
Engenharia Elétrica e Informática da
Universidade Federal de Campina
Grande, como requisito parcial para
obtenção do título de Bacharel em
Ciência da Computação.**

BANCA EXAMINADORA:

Rohit Gheyi

Orientador – UASC/CEEI/UFCG

Roberto Medeiros de Faria

Examinador – UASC/CEEI/UFCG

Francisco Vilar Brasileiro

Professor da Disciplina TCC – UASC/CEEI/UFCG

Trabalho aprovado em: 29 de JUNHO de 2023.

CAMPINA GRANDE - PB

RESUMO

A geração de programas a partir de linguagem natural visa transformar frases ou comandos em linguagem natural em código de programação. O ChatGPT é um chatbot de propósito geral baseado no modelo de linguagem GPT-3 desenvolvido para gerar texto como humano, e treinado em uma forma de conversação usando aprendizagem por reforço com feedback humano. Nesse cenário, questionamentos a respeito da confiabilidade das respostas geradas pelo ChatGPT 3.5 foram levantados. Assim, neste estudo, foi realizada uma avaliação do desempenho do modelo na resolução de 100 problemas de programação selecionados aleatoriamente de plataformas populares como LeetCode e BeeCrowd. Os problemas selecionados estão distribuídos entre os graus de complexidade Fácil, Intermediário e Difícil. Do total de 100 problemas submetidos, o modelo de linguagem conseguiu responder corretamente 71 problemas ao longo de 3 tentativas, sendo 50 deles da plataforma LeetCode e 21 da plataforma BeeCrowd. Sendo assim, é possível concluir que o ChatGPT pode ser usado para resolver uma gama de problemas, porém seu uso requer muita atenção, uma vez que nem sempre o resultado gerado estará correto.

CASE STUDY: USING CHATGPT FOR PROGRAMMING PROBLEM SOLVING

ABSTRACT

The generation of programs from natural language aims to transform sentences or commands in natural language into programming code. ChatGPT is a general-purpose chatbot based on the GPT-3 language model designed to generate text like a human. It has been trained using reinforcement learning with human feedback in a conversational format. In this scenario, concerns about the reliability of the responses generated by ChatGPT 3.5 have been raised. Therefore, this study evaluated the model's performance in solving 100 randomly selected programming problems from popular platforms such as LeetCode and BeeCrowd. The selected problems were distributed among the Easy, Intermediate, and Difficult complexity levels. Out of the total of 100 submitted problems, the language model was able to correctly answer 71 problems across 3 attempts, with 50 of them from the LeetCode platform and 21 from the BeeCrowd platform. Thus, it can be concluded that ChatGPT can be used to solve a range of problems, but its use requires careful attention as the generated result may not always be correct.

Estudo de caso: uso do ChatGPT para resolução de problemas de programação

Débora Lêda de Lucena Souza

Departamento de Sistemas e Computação
Universidade Federal de Campina Grande
Campina Grande, Paraíba, Brasil
debora.souza@ccc.ufcg.edu.br

Rohit Gheyi

Departamento de Sistemas e Computação
Universidade Federal de Campina Grande
Campina Grande, Paraíba, Brasil
rohit@dsc.ufcg.edu.br

RESUMO

A geração de programas a partir de linguagem natural visa transformar frases ou comandos em linguagem natural em código de programação. O ChatGPT é um chatbot de propósito geral baseado no modelo de linguagem GPT-3 desenvolvido para gerar texto como humano, e treinado em uma forma de conversação usando aprendizagem por reforço com feedback humano. Nesse cenário, questionamentos a respeito da confiabilidade das respostas geradas pelo ChatGPT 3.5 foram levantados. Assim, neste estudo, foi realizada uma avaliação do desempenho do modelo na resolução de 100 problemas de programação selecionados aleatoriamente de plataformas populares como LeetCode e BeeCrowd. Os problemas selecionados estão distribuídos entre os graus de complexidade *Fácil*, *Intermediário* e *Difícil*. Do total de 100 problemas submetidos, o modelo de linguagem conseguiu responder corretamente 71 problemas ao longo de 3 tentativas, sendo 50 deles da plataforma LeetCode e 21 da plataforma BeeCrowd. Sendo assim, é possível concluir que o ChatGPT pode ser usado para resolver uma gama de problemas, porém seu uso requer muita atenção, uma vez que nem sempre o resultado gerado estará correto.

Palavras-Chave

ChatGPT, LeetCode, BeeCrowd, Resolução de problemas, Problemas de Programação.

Dados

[Questões submetidas ao ChatGPT](#)

1. INTRODUÇÃO

A geração de programas a partir de linguagem natural é uma área da inteligência artificial que visa transformar frases ou comandos em linguagem natural em código de programação. Isso é feito através do uso de algoritmos de processamento de linguagem natural e sistemas de aprendizado de máquina para compreender e traduzir as intenções do usuário em uma linguagem de programação apropriada. O objetivo desta tecnologia é tornar o processo de codificação mais acessível e intuitivo para pessoas sem conhecimento de programação, aumentando a eficiência e reduzindo os erros [1].

Um exemplo de ferramenta de geração de código é o GitHub Copilot, lançada em 2021, que visa ajudar os desenvolvedores a escrever código mais rapidamente e com mais eficiência. Copilot

utiliza processamento de linguagem natural e aprendizado de máquina para sugerir linhas de código completas com base no contexto do seu projeto e da sua escrita. É uma ferramenta valiosa para desenvolvedores que procuram acelerar seu processo de codificação e minimizar erros de sintaxe. Em suma, o GitHub Copilot é uma ferramenta importante na área de geração de programas a partir de linguagem natural.

Outra ferramenta é o ChatGPT, um chatbot de propósito geral baseado no modelo de linguagem GPT-3 desenvolvido pela empresa OpenAI. Foi desenvolvido para gerar texto como humano, e treinado em uma forma de conversação usando aprendizagem por reforço com feedback humano. Os seres humanos fornecem feedback ao classificar as respostas do ChatGPT, assim, seguem ajustando o modelo. O ChatGPT já se mostrou capaz de realizar inúmeras tarefas desde escrever artigos [2] até ser aprovado em exames como Master in Business Administration (MBA) [3].

Segundo Matt Welsh, ex-professor de Harvard e ex-engenheiro da Google, esses modelos de inteligência artificial (IA) tornarão a programação obsoleta, já que a ideia convencional de “escrever um programa” será extinta e, para a maioria dos softwares, serão utilizados sistemas de IA treinados em vez de programados. Em especial, situações em que se necessita de um programa “simples”, esses programas serão gerados por uma IA ao invés de serem codificados manualmente [12].

Nesse cenário, surgiram dúvidas sobre a substituição da mão de obra humana em diversos setores, bem como questionamentos acerca da confiabilidade das respostas fornecidas pelo ChatGPT.

No contexto de programação, o ChatGPT é capaz de codificar, decodificar, explicar conceitos, encontrar bugs, otimizar e refatorar código, no entanto, apesar das respostas serem bem escritas e sem erros de sintaxe, não se sabe a confiabilidade dessas respostas geradas pelo chat. Nesse sentido, este trabalho visa avaliar a confiabilidade do ChatGPT na resolução de problemas de programação, para dar uma ideia do quanto pode ser útil para desenvolvedores na prática na geração automática de código.

Neste trabalho, nós avaliamos o desempenho do ChatGPT 3.5 ao resolver problemas de programação das plataformas LeetCode e BeeCrowd, que são duas plataformas populares para prática e aprimoramento de habilidades de programação.

O LeetCode é uma plataforma online que oferece uma ampla variedade de questões de programação, desde algoritmos e estruturas de dados até questões específicas de empresas de tecnologia. É uma plataforma popular entre estudantes e profissionais de programação que desejam se preparar para

entrevistas de emprego em empresas de tecnologia [6]. Já o BeeCrowd é uma plataforma conhecida por seus problemas de alta dificuldade, focada em treinamento e competições de programação voltadas para maratonas de programação, que é uma modalidade de competição de programação em equipe [7].

Para esse estudo, foram selecionados 50 problemas de cada plataforma com nível de dificuldade variável. Para a plataforma LeetCode, todos os problemas foram resolvidos corretamente. No entanto, para a plataforma BeeCrowd, apenas 21 problemas dos 50 foram resolvidos corretamente. Sendo assim, é possível concluir que o ChatGPT pode ser usado para resolver uma gama de problemas, porém seu uso requer muita atenção, uma vez que nem sempre o resultado gerado estará correto. Além disso, o estudo pode dar indício sobre o impacto do ChatGPT no aumento da produtividade que os engenheiros podem ter nas empresas.

Esse artigo está organizado da seguinte forma: a Seção 2 trata da fundamentação teórica, a Seção 3 trata dos objetivos, a Seção 4 trata da metodologia, a Seção 5 trata dos resultados e discussões, e a Seção 6 apresenta as conclusões obtidas.

2. FUNDAMENTAÇÃO TEÓRICA

Esta seção apresenta os conceitos necessários para o entendimento deste trabalho.

2.1 Processamento de Linguagem Natural (NLP)

A inteligência artificial (IA) é um campo de pesquisa em ciência da computação que se concentra em criar sistemas computacionais que podem realizar tarefas que normalmente requerem inteligência humana. O objetivo da IA é criar máquinas que possam pensar, raciocinar, perceber, aprender e adaptar-se a novas situações de forma semelhante aos seres humanos [14].

Processamento de Linguagem Natural (NLP) [15] refere-se ao ramo da inteligência artificial (IA) - preocupado em dar aos computadores a capacidade de entender texto e palavras faladas da mesma forma que os seres humanos [4].

A NLP combina linguística computacional — modelagem baseada em regras da linguagem humana — com modelos estatísticos, de aprendizado de máquina e de aprendizado profundo. Juntas, essas tecnologias permitem que os computadores processem a linguagem humana na forma de dados de texto ou voz e entendam seu significado completo, completo com a intenção e o sentimento do falante ou escritor [4].

2.2 GPT-3

O GPT-3, ou a terceira geração Generative Pre-trained Transformer, é um modelo de aprendizado de máquina de rede neural treinado usando dados da Internet para gerar qualquer tipo de texto. Desenvolvido pela OpenAI, requer uma pequena quantidade de texto de entrada para gerar grandes volumes de texto gerados por máquina [5].

O GPT-3 é treinado primeiro por meio de uma fase de teste supervisionada e depois por uma fase de reforço. Ao treinar o ChatGPT, uma equipe de treinadores faz uma pergunta ao modelo de linguagem com uma saída correta em mente. Se o modelo

responder incorretamente, os treinadores ajustam o modelo para ensiná-lo a resposta certa. O modelo também pode dar várias respostas, que os treinadores classificam da melhor para a pior [5].

O GPT-3 tem mais de 175 bilhões de parâmetros de aprendizado de máquina e é significativamente maior do que seus modelos predecessores de linguagem grandes anteriores, como Bidirectional Encoder Representations from Transformers (BERT) [16] e Turing NLG (Natural Language Generation). Parâmetros são as partes de um grande modelo de linguagem que definem sua habilidade em um problema como geração de texto. O desempenho de um grande modelo de linguagem geralmente escala à medida que mais dados e parâmetros são adicionados ao modelo [5].

3. OBJETIVO

O objetivo deste trabalho é verificar a confiabilidade das respostas geradas pelo ChatGPT para resolução de problemas de programação, bem como levantar implicações do seu uso no contexto do dia a dia de programadores.

4. METODOLOGIA

Para avaliar se o ChatGPT responde corretamente problemas de programação, 100 problemas das plataformas LeetCode e BeeCrowd foram selecionados e submetidos ao ChatGPT. Um exemplo de problema selecionado está na Figura 1 a seguir, o problema contém a descrição, exemplos com valores de entrada e saída e as restrições a respeito das variáveis de entrada.

14. Longest Common Prefix

Write a function to find the longest common prefix string amongst an array of strings.

If there is no common prefix, return an empty string "".

Example 1:

```
Input: strs = ["flower","flow","flight"]
Output: "fl"
```

Example 2:

```
Input: strs = ["dog","racecar","car"]
Output: ""
Explanation: There is no common prefix
among the input strings.
```

Constraints:

- `1 <= strs.length <= 200`
- `0 <= strs[i].length <= 200`
- `strs[i]` consists of only lowercase English letters.

Figura 1. Exemplo de problema selecionado da plataforma LeetCode

Sendo assim, para avaliação do problema pelo ChatGPT, todo o texto (descrição, exemplos e restrições) do problema foi copiado da plataforma e colado no prompt para o chat responder. Este, por sua vez, retorna uma solução que é posta na plataforma, submetida e avaliada como aceita ou rejeitada.

Apesar de no problema usado como exemplo não conter figura explicativa, é comum que os enunciados contenham figuras para elucidar o problema, no entanto, este artifício não foi passado para ser avaliado junto a questão pelo ChatGPT. Como dito anteriormente, apenas o texto completo do enunciado é considerado uma vez que o chat aceita apenas input de texto.

A plataforma LeetCode divide seus problemas em níveis de dificuldade, sendo eles *fácil*, *intermediário* e *difícil*. Sendo assim, 50 problemas foram selecionados onde são 20 do nível fácil, 15 do nível intermediário e 15 do nível difícil.

Já a plataforma BeeCrowd divide seus problemas por nível de dificuldade de 1 a 10, sendo assim, 5 questões de cada nível foram selecionadas, totalizando 50 problemas. Para facilitar a comparação com os problemas do LeetCode, as questões foram redistribuídas em três níveis semânticos, sendo eles:

- Fácil: questões do nível 1 ao 4;
- Intermediário: questões do nível 5 ao 7;
- Difícil: questões do nível 8 ao 10.

É importante ressaltar que no BeeCrowd o nível de dificuldade é estimado usando uma variação do *ELO Rating System* [9] que usa o número de vezes que o problema foi "derrotado" (ou seja, quantos usuários resolveram o problema) para determinar sua dificuldade. Nesse sistema, problemas que são resolvidos por vários usuários com poucas tentativas são classificados como de baixa dificuldade, enquanto problemas com mais tentativas, mas menos resolvidos, são classificados como de alta dificuldade. A dificuldade dos problemas é reajustada semanalmente. Isso significa que os problemas podem mudar de nível e ter sua pontuação alterada. Portanto, sua pontuação total pode variar de semana para semana [8]. Os problemas utilizados para essa análise foram selecionados em 01/04/2023.

Uma vez que o ChatGPT pode retornar respostas diferentes para a mesma pergunta, o ChatGPT teve um total de três chances para resolver cada problema. Se após a terceira submissão a resposta ainda for incorreta, então o problema é identificado como não solucionado. Cada problema foi submetido individualmente ao modelo em um prompt exclusivo, o prompt foi reutilizado apenas quando houve a necessidade da segunda e terceira submissão.

Para cada problema avaliado, foi arquivado em uma planilha para posterior análise seu enunciado, nível de dificuldade, código de solução proposto pelo ChatGPT, o status de correteza da submissão, e o erro lançado pela plataforma no caso da resposta ser incorreta. A versão do ChatGPT usada foi treinada com dados até setembro de 2021 e atualizada em janeiro de 2023 para fornecer respostas precisas e atualizadas.

5. RESULTADOS E DISCUSSÃO

Com o objetivo de realizar a pesquisa sobre a capacidade do ChatGPT em solucionar problemas, foram selecionados 100 problemas das plataformas LeetCode e BeeCrowd. Para avaliar a

eficácia do modelo em resolver tais questões, cada um dos problemas selecionados teve seu enunciado submetido ao ChatGPT por no máximo três vezes. Os resultados dessa pesquisa estão apresentados na Tabela 1 a seguir, na qual é possível verificar o número de acertos e erros classificados de acordo com o nível de complexidade dos problemas. Os resultados específicos para cada plataforma são explorados em sequência.

	Nível Difícil	Nível Fácil	Nível Intermediário
Correta	17	36	18
Incorreta	13	4	12

Tabela 1. Número de acertos e erros classificados de acordo com o nível de complexidade

O portal LeetCode está entre as melhores plataformas para avaliar conhecimento sobre programação, com questões que abordam os mais variados conceitos de computação. Muitas vezes, o enunciado da questão é contextualizado e requer uma etapa de interpretação para saber o que deve ser implementado e qual conceito deve ser aplicado para solução.

Na plataforma LeetCode, o ChatGPT obteve um ótimo desempenho na resolução de problemas. Foram submetidos 50 problemas, onde são 20 do nível fácil, 15 do nível intermediário e 15 do nível difícil. Após a submissão e avaliação das respostas coletadas, obtemos que das 50 questões submetidas ao ChatGPT, todas as 50 questões foram respondidas corretamente, como mostra a Figura 2 abaixo.

Proporção de questões respondidas

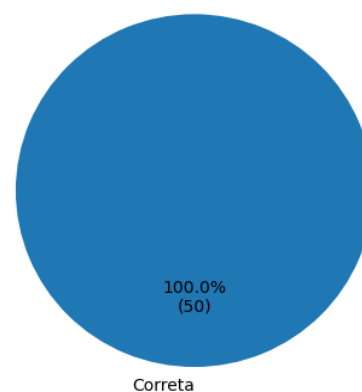


Figura 2. Gráfico com proporção de problemas respondidos na plataforma LeetCode

Dos 50 problemas submetidos à plataforma LeetCode, 16 deles (32%) contém imagens ilustrativas acompanhando o enunciado, sendo 6 problemas do nível fácil, 5 do nível intermediário e 5 do nível difícil. Ao submeter os problemas para o ChatGPT, as imagens foram desconsideradas, e apenas o texto do enunciado foi utilizado. Ainda assim, 14 desses problemas foram solucionados na primeira tentativa, e apenas 2 requereram uma segunda submissão. Nenhum problema com imagem exigiu uma terceira submissão. Isso demonstra que, embora o modelo não possa processar imagens, ele é capaz de lidar efetivamente com o texto

do enunciado quando este é suficiente para compreender o problema.

Como cada um dos problemas foi submetido no máximo três vezes, é possível avaliar como se dá a proporção de acertos por número de tentativas.

De acordo com a Figura 3 apresentada a seguir, dos 50 problemas selecionados, aproximadamente 88% deles foram resolvidos corretamente na primeira submissão ao ChatGPT. Todos os problemas de nível fácil foram solucionados na primeira tentativa, bem como cerca de 93% dos problemas de nível intermediário e aproximadamente 67% dos problemas de nível difícil. Os problemas que não foram resolvidos na primeira tentativa foram submetidos novamente ao ChatGPT. Na segunda rodada de submissões, o ChatGPT conseguiu solucionar 100% dos problemas restantes do nível intermediário, bem como 80% dos problemas restantes do nível difícil. Por fim, na terceira rodada de submissões, o ChatGPT conseguiu resolver 100% dos problemas remanescentes do nível difícil.

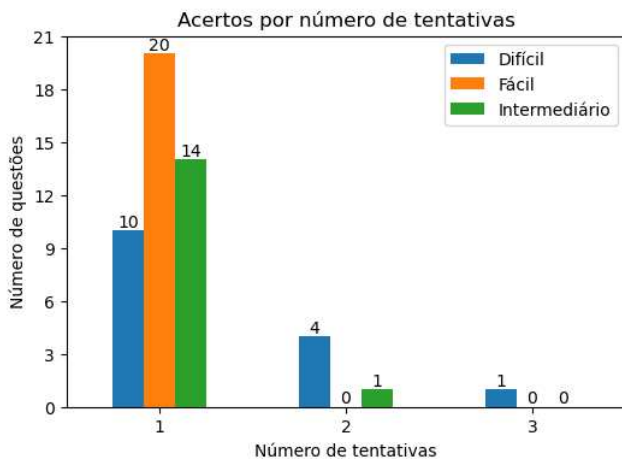


Figura 3. Gráfico com número de acertos por número de tentativas na plataforma LeetCode

Outra plataforma analisada foi o BeeCrowd, uma plataforma conhecida por seus problemas de alta dificuldade, focada em treinamento e competições de programação voltadas para maratonas de programação. A plataforma conta com mais de dois mil e trezentos problemas típicos de competições de programação, em formato muito similar aos problemas disponibilizados na Olimpíada Brasileira de Informática (OBI) e na Maratona de Programação da SBC. Além disso, todos os problemas das edições anteriores da OBI e da Maratona de Programação (Fase eliminatória e Fase final) estão disponíveis no portal. Também estão disponíveis muitos problemas de competições regionais (como por exemplo a Maratona Mineira) [10].

Nessa plataforma, o ChatGPT obteve um desempenho moderado. Assim como na plataforma LeetCode, foram submetidos 50 problemas, sendo 20 do nível fácil, 15 do nível intermediário e 15 do nível difícil. Após a submissão e avaliação das respostas coletadas, obtemos que das 50 questões respondidas pelo ChatGPT, apenas 21 delas, cerca de 42%, foram respondidas corretamente, como mostra a Figura 4 a seguir.



Figura 4. Gráfico com proporção de problemas respondidos na plataforma BeeCrowd

Dos 50 problemas submetidos à plataforma BeeCrowd, apenas 5 deles (10%) possuíam imagens ilustrativas acompanhando o enunciado. Desses, 4 problemas eram de nível fácil, enquanto 1 era de nível difícil. Ao submeter os problemas para o ChatGPT, as imagens foram desconsideradas, e apenas o texto do enunciado foi utilizado. Entre os problemas de nível fácil, 3 foram solucionados corretamente, independentemente das imagens, enquanto 2 problemas não foram resolvidos corretamente, sendo 1 de nível fácil e 1 de nível difícil. Devido à escolha aleatória dos problemas e à baixa quantidade de problemas com imagens (apenas 5), não dispomos de dados suficientes para determinar se as imagens afetaram ou não o desempenho do ChatGPT na resolução dos problemas.

Ao avaliar a distribuição do número de questões corretas e incorretas por grau de dificuldade, é possível verificar como foram os resultados para problemas de nível fácil, intermediário e difícil. Conforme ilustrado na Figura 5 abaixo, dos 20 problemas de nível fácil apresentados, aproximadamente 80% deles foram respondidos corretamente, totalizando 16 acertos. Em contrapartida, as questões de níveis intermediário e difícil tiveram um desempenho inferior, com apenas 3 e 2 problemas corretamente respondidos, o que equivale a cerca de 20% e 13%, respectivamente.

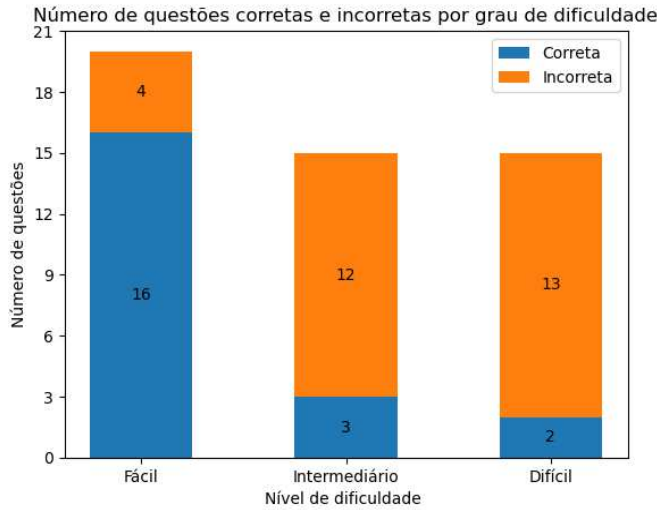


Figura 5. Gráfico com número de questões corretas e incorretas por grau de dificuldade na plataforma BeeCrowd

Uma vez que cada problema foi submetido no máximo três vezes, é possível analisar a proporção de acertos em relação ao número de tentativas.

De acordo com a Figura 6 apresentada a seguir, apenas 36% dos problemas foram resolvidos corretamente na primeira tentativa. Isso significa que 70% dos problemas de nível fácil foram solucionados corretamente na primeira rodada de submissões, bem como 20% dos problemas de nível intermediário e apenas 7% dos problemas de nível difícil. Os problemas que não foram solucionados na primeira tentativa foram submetidos novamente ao ChatGPT. Na segunda rodada de submissões, 33% dos problemas restantes de nível fácil foram solucionados, bem como 7% dos problemas remanescentes de nível difícil. Entretanto, nenhum problema de nível intermediário foi solucionado nesta rodada. Por fim, na terceira rodada de submissões, nenhum problema foi solucionado.

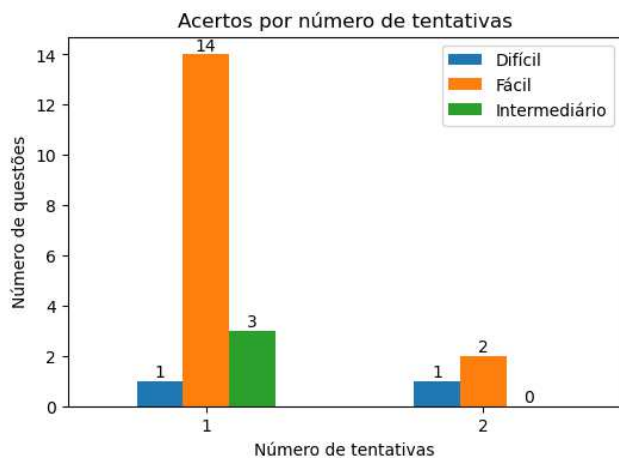


Figura 6. Gráfico com número de acertos por número de tentativas na plataforma BeeCrowd

Os problemas encontrados na plataforma BeeCrowd originam-se de diversas fontes. Uma vez que as questões selecionadas para este estudo foram escolhidas de forma aleatória, suas origens

também são variadas. Foram utilizados problemas de instituições como OBI, IFSULDEMINAS, Maratona de Programação da SBC, UNB, USP, entre outras. Portanto, não foi possível estabelecer uma correlação entre problemas não resolvidos e uma fonte específica.

A plataforma BeeCrowd fornece uma saída contendo o erro associado para cada problema que não é solucionado. Sendo assim, é possível analisar os tipos de erros gerados e sua proporção de ocorrência.

Como mostra a Figura 7 a seguir, quatro erros ao total foram lançados, os principais erros produzidos foram Time Limit Exceeded e Wrong Answer (100%). Time Limit Exceeded é um erro lançado quando a solução enviada leva mais tempo do que o permitido para executar todos os testes de avaliação [13]. Já o Wrong Answer (100%) é lançado quando a solução não apresenta o resultado esperado para todos os casos de teste. O valor exibido ao lado da resposta significa a porcentagem de sua resposta que é diferente da saída esperada. Por exemplo, se o envio retornou Wrong Answer (90%), significa que a saída difere 90% da correta [13].

Além disso, há também o erro Runtime Error, lançado em menor quantidade, que diz respeito a definir um vetor ou array com capacidade inferior à necessária para o problema, ou ao tentar acessar uma posição inválida na memória [13]. Já Memory Limit Exceeded, erro lançado apenas uma vez, é gerado quando o código tenta alocar mais memória do que o máximo permitido para o problema. Isso pode ocorrer porque está sendo utilizado um vetor ou uma estrutura de dados muito grande [13].

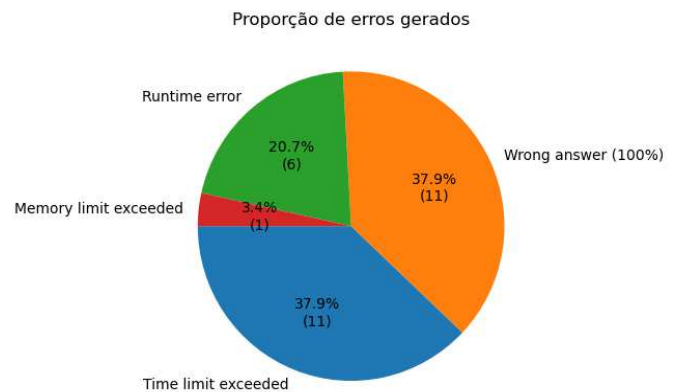


Figura 7. Gráfico com tipos de erros gerados pelo BeeCrowd e suas proporções de ocorrência

Sendo assim, é possível afirmar que, em sua maioria, as respostas incorretas geradas pelo ChatGPT são rejeitadas por não terem sido suficientemente otimizadas para serem aprovadas ou por não terem passado em nenhum caso de teste.

6. CONCLUSÃO

Conforme evidenciado, o ChatGPT tem um desempenho satisfatório, especialmente em problemas de complexidade média e baixa, no entanto, apresenta certa dificuldade em problemas considerados difíceis.

Uma possível explicação para o desempenho insatisfatório do ChatGPT na plataforma BeeCrowd pode estar relacionada ao fato de que seus problemas costumam apresentar histórias fictícias e

lúdicas para ilustrar o problema, envolvendo personagens e exemplos, o que aumenta a complexidade do texto. Por outro lado, na plataforma LeetCode, os enunciados são mais breves, objetivos e diretos ao ponto.

O uso do ChatGPT pode ser vantajoso para programadores, uma vez que o modelo é capaz de fornecer exemplos de códigos bem escritos e oferecer soluções para seus problemas. Mas, é importante que o programador possua uma base de conhecimento para revisar as respostas geradas pelo modelo. É também recomendável que ele evite depender exclusivamente do ChatGPT para solucionar todos os problemas, a fim de evitar a criação de uma dependência prejudicial ao desenvolvimento de suas habilidades. Além disso, é essencial que o programador mantenha sua capacidade de pensar criticamente ativa, a fim de não perder a habilidade de resolver problemas de forma independente.

De todo modo, o uso do modelo pode ser muito útil para agilizar o desenvolvimento de software. No entanto, o grande desafio da programação é saber o que deve ser implementado e isso é obtido a partir do conhecimento a respeito do domínio do problema e da solução, mas o ChatGPT nem sempre terá acesso a detalhes desses domínios e como visto aqui no estudo, mesmo quando o domínio é pequeno (uma questão de plataforma) o modelo pode não compreender completamente e gerar respostas incorretas. Outro problema que pode ser enfrentado é o fato do ChatGPT gerar códigos que podem não seguir a padronização usada pelo time de desenvolvimento, o que pode ocasionar conflito caso o programador não reescreva o código gerado.

Já para estudantes, há a vantagem de ter a sua disposição um auxiliar para explicar conteúdos e esclarecer dúvidas, mas também há desvantagens, uma vez que o ChatGPT pode responder erroneamente e o estudante não estará capacitado para detectar o erro.

Em resumo, o ChatGPT pode ser muito útil para o programador, contanto que ele seja responsável, atencioso e crítico o suficiente para avaliar a resposta gerada e não utilizá-la sem fazer os devidos testes.

6.1 Limitações e Trabalhos Futuros

Como sugestão para trabalhos futuros, é possível explorar o potencial da versão ChatGPT-4 (indisponível gratuitamente no momento desse estudo), que pode oferecer um melhor desempenho na resolução de problemas mais complexos, graças ao seu conhecimento geral mais amplo e habilidades de resolução de problemas [11], além de aceitar o input de imagens. Outra possibilidade seria realizar o estudo considerando outras plataformas de problemas, como HackerRank e Codeforces, com o intuito de identificar um padrão de estrutura de problema no qual o ChatGPT possa ter um melhor desempenho. Uma abordagem similar pode ser realizada em modelos e ver até que ponto o ChatGPT ajuda na síntese e evolução de especificações [17].

REFERÊNCIAS

- [1] ChatGPT. Disponível em: <<https://openai.com/blog/chatgpt>>. Acesso em: 21 abr. 2023.
- [2] Recruitment team unwittingly recommends ChatGPT for job interview. Disponível em: <<https://news.sky.com/story/recruitment-team-unwittingly-recommends-ChatGPT-for-job-interview-12788770>>. Acesso em: 01 mar. 2023.
- [3] ChatGPT passes US MBA, Law, and Medical exams. Disponível em: <<https://www.showmetech.com.br/en/ChatGPT-passed-testin-g-in-the-us/>>. Acesso em: 01 mar. 2023.
- [4] Natural Language Processing. Disponível em: <<https://www.ibm.com/topics/natural-language-processing>>. Acesso em: 03 abr. 2023.
- [5] What is GPT-3? Everything You Need to Know. Disponível em: <<https://www.techtarget.com/searchenterpriseai/definition/GPT-3>>. Acesso em: 03 abr. 2023.
- [6] Plataforma LeetCode. Disponível em: <<https://leetcode.com/>>. Acesso em: 05 abr. 2023.
- [7] Plataforma BeeCrowd. Disponível em: <<https://BeeCrowd.com.br/>>. Acesso em: 05 abr. 2023.
- [8] FAQs Judge - BeeCrowd. Disponível em: <<https://www.becrowd.com.br/judge/en/faqs>>. Acesso em: 05 abr. 2023.
- [9] ELO Rating System. Disponível em: <<https://medium.com/purple-theory/what-is-elo-rating-c4eb7a9061e0>>. Acesso em: 05 abr. 2023.
- [10] Programação Competitiva Beecrowd. Disponível em: <<https://becrowd.io/blog/programacao-competitiva/>>. Acesso em: 10 abr. 2023.
- [11] GPT-4. Disponível em: <<https://openai.com/product/gpt-4>>. Acesso em: 21 abr. 2023.
- [12] The End of Programming. Disponível em: <<https://cacm.acm.org/magazines/2023/1/267976-the-end-of-programming/fulltext>>. Acesso em: 21 abr. 2023.
- [13] FAQs Answer - BeeCrowd. Disponível em: <<https://www.becrowd.com.br/judge/en/answers>>. Acesso em: 21 abr. 2023.
- [14] RUSSELL, S. J.; NORVIG, P. Artificial Intelligence: A Modern Approach. 3. ed. Pearson, 2010. Cap. 1, pp. 1-5.
- [15] RUSSELL, S. J.; NORVIG, P. Artificial Intelligence: A Modern Approach. 3. ed. Pearson, 2010. Cap. 22, pp. 860-882.
- [16] Devlin, Jacob; Chang, Ming-Wei; Lee, Kenton; Toutanova, Kristina. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. arXiv preprint arXiv:1810.04805, 2018. Disponível em: <<https://arxiv.org/abs/1810.04805>>. Acesso em: [26 de maio de 2023].
- [17] Gheyi, Rohit; Borba, Paulo: Refactoring Alloy Specifications. In Cavalcanti, A., Machado, P., eds.: Electronic Notes in Theoretical Computer Science, Proceedings of the Brazilian Workshop on Formal Methods. Volume 95. Elsevier (2004) 227–243.

Sobre a autora:

Débora Lêda de Lucena Souza é graduanda em Ciência da Computação na Universidade Federal de Campina Grande.