XAI for Medicine by ChatGPT Code interpreter

Kenta, Kitamura *
The University of Tokyo, address
kitamura@yamagula.ic.i.u-tokyo.ac.jp

Mhd, Irvan
The University of Tokyo, l address
irvan@yamagula.ic.i.u-tokyo.ac.jp

Rie, Shigetomi Yamaguchi
The University of Tokyo, address
yamaguchi.rie@i.u-tokyo.ac.jp

ABSTRACT

In recent years, with the prevalence of Artificial Intelligence (AI), the interpretability of AI outputs has become a significant issue. Especially the interpretability of large language models (LLMs), including ChatGPT, has emerged as a major challenge. Consequently, there is a growing interest in the research of Explainable Artificial Intelligence (XAI), which seeks to elucidate the decision-making processes of AI in a manner that humans can comprehend. In the medical field, where trust and transparency are important, the use of AI becomes challenging when its decisions are unclear. Therefore, XAI techniques become critically important in the medical field. In this study, we propose the prompt named Code Base Prompt (CBP) to make the ChatGPT's decision-making process on medical texts explainable by using the Python code execution function of Chat GPT Code interpreter. In CBP, the medical decision-making algorithm is rewritten as Python code. Moreover, we propose an explainability evaluation system named Medical Algorithm Presentation Criteria (MAPC) for medical algorithm application tasks to medical text. MAPC is evaluated by five factors to align the human understanding process. To compare CBP with a Text Base Prompt (TBP), we conducted an experiment applying the heart failure classification algorithm to heart failure case report texts in three medical articles. With CBP, the results showed that the ChatGPT Code interpreter executed the Python code in all three cases and met all the five MAPC factors. In contrast, with TBP, no Python code execution was observed in any of the three cases, validating only one factor of MAPC. This study presents a new method for implementing XAI in the use of ChatGPT for medical tasks.

CCS CONCEPTS

• **Computing methodologies** → Artificial intelligence.

KEYWORDS

ChatGPT Code interpreter, ChatGPT, XAI, health care, medicine

ACM Reference Format:
Kenta, Kitamura, Mhd, Irvan, and Rie, Shigetomi Yamaguchi. 2023. XAI for Medicine by ChatGPT Code interpreter. In *2023 5th International Conference on Big-data Service and Intelligent Computation (BDSIC 2023), October 20–22, 2023, Singapore, Singapore.* ACM, New York, NY, USA, 7 pages. https://doi.org/10.1145/3633624.3633629

1 INTRODUCTION

In recent years, the use of Artificial Intelligence (AI) has become very popular [1]. However, the black-box nature of AI outputs can lead to challenges in understanding and interpreting the reasons behind its decisions, which might pose problems in real-world AI applications [2]. To address the opacity of AI, Explainable Artificial Intelligence (XAI) has become important [3].

In particular, Large Language Models (LLMs), including ChatGPT, have been increasingly utilized [4]. While LLM has broad applications, there can be issues related to the interpretability of their outputs and hallucinations [5]. Thus, enhancing the explainability of LLM outputs has become a significant challenge. Efforts to improve interpretability via prompts have also been explored [6] [7]. However, the criterion for explainability is based on the text responses from ChatGPT, which could be merely the most plausible prediction of word sequences, leaving the actual thought process behind the decision ambiguous [8].

In the medical field, the application of AI has been growing [9]. Yet, the black-box nature of AI can create significant issues related to interpretability and trust in medical settings [10]. As a solution, attempts to apply XAI techniques to medical AI have been proposed [11]. Especially in challenges like selecting suitable patients for clinical trials from medical texts, there have been efforts to apply the classification algorithm to medical texts by ChatGPT [12]. However, if LLM's responses are text-based, there's an inherent problem of not being able to objectively verify the thought process [8]. For instance, as depicted in Figure 1, when ChatGPT receives a Text Base Prompt (TBP), ChatGPT interprets the text and produces a text-based response. But even if a human reads this response, the logic behind ChatGPT's answer remains opaque.

Therefore, we propose Code Base Prompt (CBP) to extract programming code-based answers from the ChatGPT Code interpreter that represent decision-making factors essential to human judgment. For example, as shown in Figure 2, CBP commands ChatGPT Code interpreter to execute the Python program. When a medical algorithm is executed by ChatGPT as Python code, humans can verify the decision-making process details. As shown in Figure 2, this includes the extraction of numerical values like LVEF = 11 from the medical text, the algorithm's execution process, and the determination of the result as 'Type: HFrEF'.

For assessing the explanatory power of CBP and TBP, we decompose the human thought process [13] when applying medical algorithms [14] to medical texts into five steps. Then, we propose the Medical Algorithm Presentation Criteria (MAPC) as an indicator for AI explainability based on the explicit representation of each human understanding step.

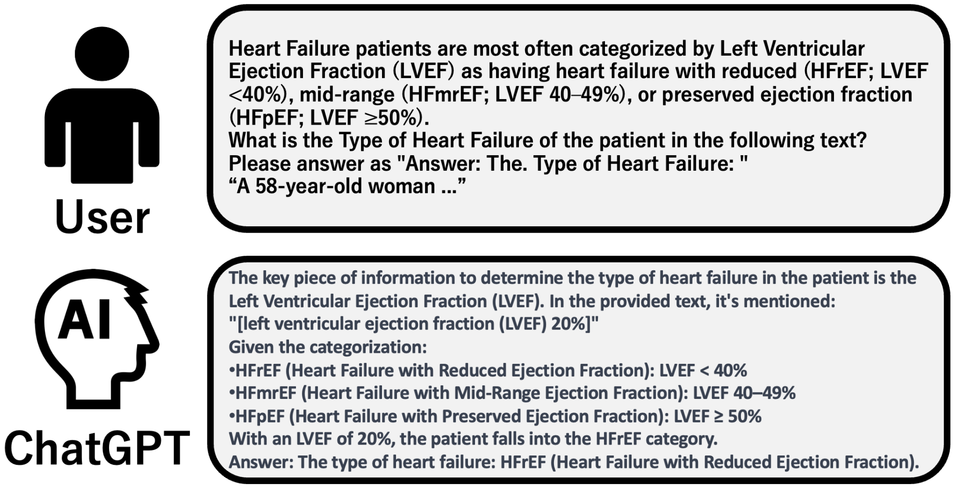
In this paper, we specifically test the outputs of ChatGPT Code interpreter [15] for CBP and TBP, focusing on a heart failure classification algorithm applied to three publicly available heart failure

---

*Place the footnote text for the author (if applicable) here.

This work is licensed under a Creative Commons Attribution International 4.0 License.

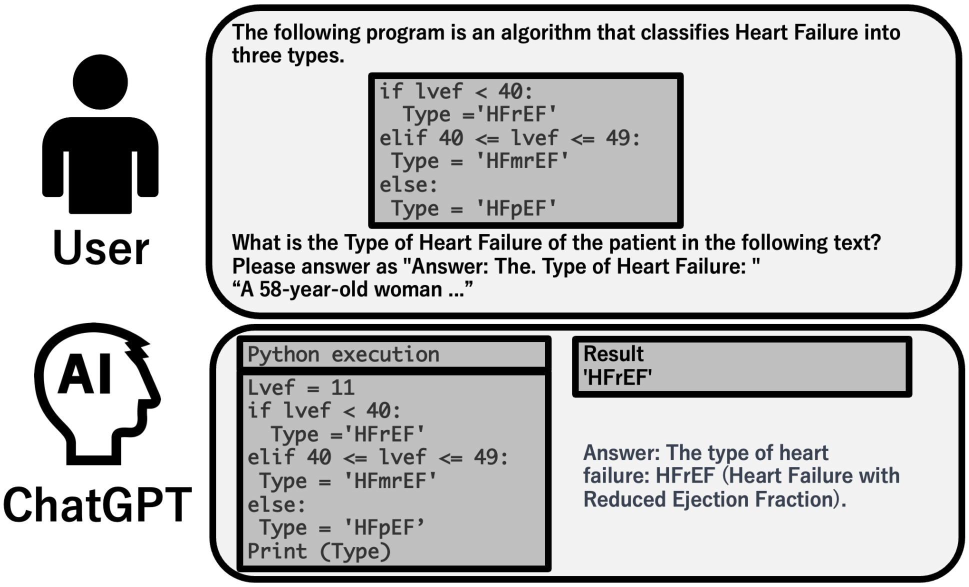
BDSIC 2023, October 20–22, 2023, Singapore, Singapore
© 2023 Copyright held by the owner/author(s).
ACM ISBN 979-8-4007-0892-3/23/10.
https://doi.org/10.1145/3633624.3633629



Heart Failure patients are most often categorized by Left Ventricular Ejection Fraction (LVEF) as having heart failure with reduced (HFrEF; LVEF <40%), mid-range (HFmrEF; LVEF 40–49%), or preserved ejection fraction (HFpEF; LVEF ≥50%).
What is the Type of Heart Failure of the patient in the following text?
Please answer as "Answer: The. Type of Heart Failure: "
“A 58-year-old woman ...”

The key piece of information to determine the type of heart failure in the patient is the Left Ventricular Ejection Fraction (LVEF). In the provided text, it's mentioned:
"[left ventricular ejection fraction (LVEF) 20%]"
Given the categorization:
•HFrEF (Heart Failure with Reduced Ejection Fraction): LVEF < 40%
•HFmrEF (Heart Failure with Mid-Range Ejection Fraction): LVEF 40–49%
•HFpEF (Heart Failure with Preserved Ejection Fraction): LVEF ≥ 50%
With an LVEF of 20%, the patient falls into the HFrEF category.
Answer: The type of heart failure: HFrEF (Heart Failure with Reduced Ejection Fraction).

Figure 1: Example of the Text Base Prompt (TBP) and the ChatGPT Code interpreter answer.



The following program is an algorithm that classifies Heart Failure into three types.

```
if lvef < 40:
  Type ='HFrEF'
elif 40 <= lvef <= 49:
 Type = 'HFmrEF'
else:
 Type = 'HFpEF'
```

What is the Type of Heart Failure of the patient in the following text?
Please answer as "Answer: The. Type of Heart Failure: "
“A 58-year-old woman ...”

Python execution

```
Lvef = 11
if lvef < 40:
  Type ='HFrEF'
elif 40 <= lvef <= 49:
 Type = 'HFmrEF'
else:
 Type = 'HFpEF’
Print (Type)
```

Result

```
'HFrEF'
```

Answer: The type of heart failure: HFrEF (Heart Failure with Reduced Ejection Fraction).

Figure 2: Example of the Code Base Prompt (CBP) and the ChatGPT Code interpreter answer.

case studies. The results showed that with CBP, for all three cases, the application of the heart failure classification algorithm could be verified through ChatGPT executing Python programs, fulfilling all the five MAPC factors. In contrast, with TBP, no Python code execution by ChatGPT was observed, and only one of the five factors was met.

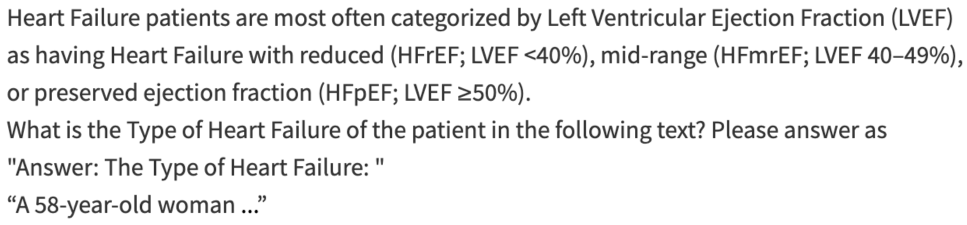
In summary, the main contributions of this study are:

- We proposed an XAI method using CBP that induces the ChatGPT Code interpreter to execute Python code to apply a medical algorithm to medical text data.
- We introduced MAPC as an evaluation method for the explainability of ChatGPT’s decision-making process for both CBP and TBP.
- We evaluated the outputs of ChatGPT Code interpreter for CBP and TBP using MAPC in the context of the heart failure classification algorithm applied to heart failure case reports, confirming that CBP outperforms TBP in terms of explainability.

This study shows the potential of XAI realization in the medical field by ChatGPT Code interpreter.

2 BACKGROUND

In this section, we explain ChatGPT Code interpreter and the heart failure classification criteria by Left Ventricular Ejection Fraction (LVEF).



Heart Failure patients are most often categorized by Left Ventricular Ejection Fraction (LVEF) as having Heart Failure with reduced (HFrEF; LVEF <40%), mid-range (HFmrEF; LVEF 40–49%), or preserved ejection fraction (HFpEF; LVEF ≥50%).
What is the Type of Heart Failure of the patient in the following text? Please answer as "Answer: The Type of Heart Failure: "
“A 58-year-old woman ...”

Figure 3: Text Base Prompt (TBP) for Heart Failure classification by Left Ventricular Ejection Fraction (LVEF).

2.1 ChatGPT Code interpreter

ChatGPT Code Interpreter [15] is designed by OpenAI and uses LLM model that supports text-based interactions. ChatGPT Code Interpreter facilitates code execution and problem-solving through user interactions. ChatGPT is not limited to answering questions; it can also provide code examples. The ChatGPT Code Interpreter is utilized for various tasks, including code debugging, optimization advice, explaining programming concepts, and data analysis.

2.2 Heart failure classification criteria by Left Ventricular Ejection Fraction (LVEF)

Heart failure [16] refers to a condition where the heart is unable to supply enough blood to the body. Heart failure is classified into three types based on an indicator called Left Ventricular Ejection Fraction (LVEF) [17], which measures the amount of blood ejected from the left ventricle of the heart in a single contraction. The first type, known as Heart Failure with Preserved Ejection Fraction (HFpEF), occurs when the LVEF is 50% or higher. The second type, termed Heart Failure with Reduced Ejection Fraction (HFrEF), is diagnosed when the LVEF is less than 40%. The third type, called Heart Failure with Mid-Range Ejection Fraction (HFmrEF), is identified when the LVEF ranges between 40% and 49%. Accurate classification of heart failure is essential for effective medical intervention, as each type of heart failure has different characteristics and treatments.

3 CODE BASE PROMPT (CBP) CREATION FROM TEXT BASE PROMPT(TBP)

In this section, we discuss the method of transforming TBP, as shown in Figure 3, to CBP, as depicted in Figure 4, for the task of applying medical algorithms to medical texts. We first explain how to convert a medical algorithm into a Python code format. For instance, the classification criteria for heart failure states that heart failure is algorithmically categorized into three types based on LVEF: HFrEF for LVEF ≤ 40, HFmrEF for $41 \leq LVF \leq 49$, and HFpEF for LVEF ≥ 50. This criterion, when expressed in test form, is depicted in Figure 3. To transform this text-based criterion into Python code, we made a request to the ChatGPT Code Interpreter as follows: *“Heart Failure patients are most often categorized by* Left Ventricular Ejection Fraction (LVEF) *as having Heart Failure with reduced (HFrEF; LVEF ≤ 40 %), mid-range (HFmrEF; LVEF 40 − 49 %), or preserved ejection fraction (HFpEF; LVEF ≥ 50 %). Can you make Python code to distinguish Heart Failure?”* As shown in Figure 4, in response, we received the Python code. Before the obtained Python code, we include the sentence, *“The following program is an algorithm that classifies Heart Failure into three types,”* to explain the purpose of the medical algorithm. After the Python code, similar to the TBP, we insert the sentence, *“What is the Type of Heart Failure of the patient in the following text? Please answer as "Answer: The Type of Heart Failure:””* to present the task. Figure 5 shows the ChatGPT Code interpreter response result example to the CBP. The Python code is executed, and a human can understand that ChatGPT uses the value of LVEF = 18, executes the LVEF classification algorithm, and the Python code result is HFrEF.

4 MEDICAL ALGORITHM PRESENTATION CRITERIA(MAPC)

In this section, we propose an explainability evaluation system for ChatGPT Code interpreter outputs for medical algorithm application tasks to the medical text.

4.1 Explainability factors

For human understandability [13], we decompose the human thought process of classification and diagnosis for medical textual information by medical algorithms [14] into five steps. We denote these five factors as F1 through F5:

F1: Understanding the logic of the medical algorithm.

F2: Grasping the applicability of the medical algorithm to the medical text.

F3: Extracting appropriate input data for the medical algorithm from the medical text.

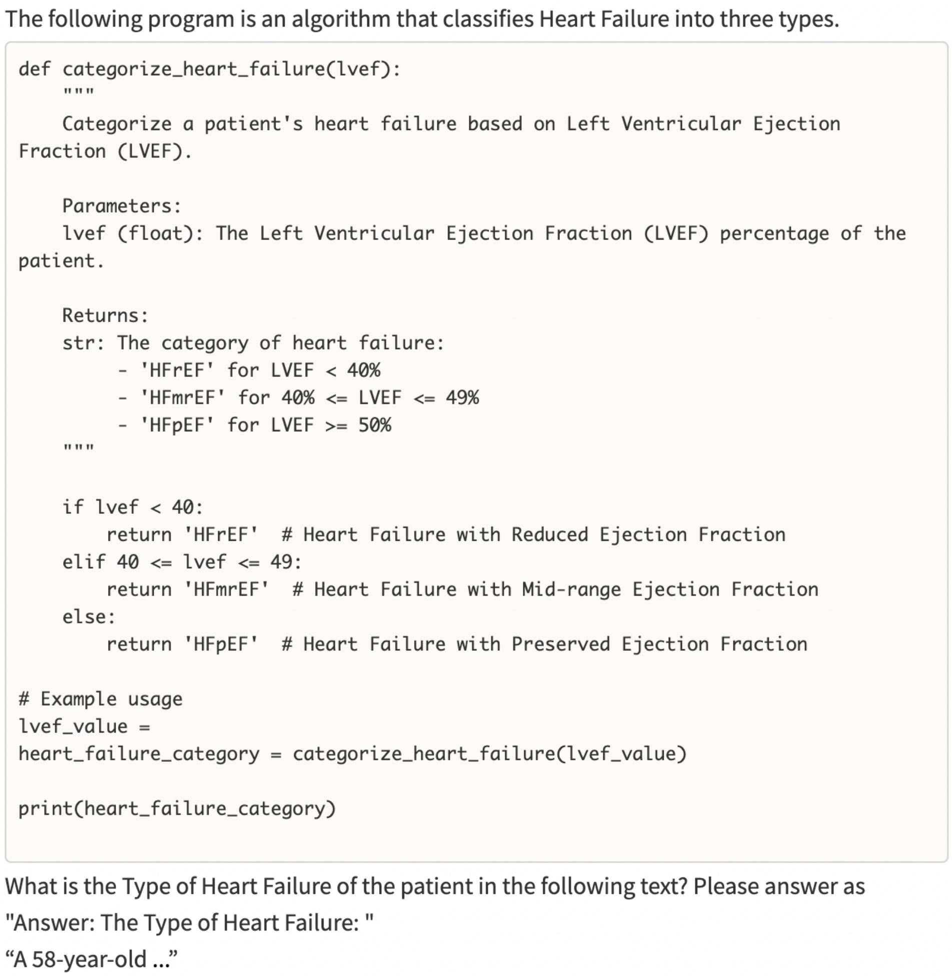
F4: Correct execution of the medical algorithm.

F5: Output of the correct answer.

We propose a set of explainability metrics, which we call Medical Algorithm Presentation Criteria (MAPC), to measure whether ChatGPT’s decisions are explicit for each of the five factors. Each of these factors serves as a criterion for evaluating the explainability of ChatGPT’s output.

4.2 MAPC evaluation for ChatGPT Code interpreter answer on TBP and CBP

In the case of TBP, it has been demonstrated that metrics such as accuracy can be measured through ChatGPT’s text-based answers [13]. Therefore, we believe that F5 in MAPC can be verified by looking at the text if the answer format is provided to ChatGPT. For instance, if instructed to output in the format “Answer: ”, and it outputs “Answer: HFrEF”, which matches the correct answer, then F5 can be verified. However, F2 is circumstantial evidence, and it



The following program is an algorithm that classifies Heart Failure into three types.

```
def categorize_heart_failure(lvef):
    """
    Categorize a patient's heart failure based on Left Ventricular Ejection
Fraction (LVEF).

    Parameters:
    lvef (float): The Left Ventricular Ejection Fraction (LVEF) percentage of the
patient.

    Returns:
    str: The category of heart failure:
         - 'HFrEF' for LVEF < 40%
         - 'HFmrEF' for 40% <= LVEF <= 49%
         - 'HFpEF' for LVEF >= 50%
    """

    if lvef < 40:
        return 'HFrEF'  # Heart Failure with Reduced Ejection Fraction
    elif 40 <= lvef <= 49:
        return 'HFmrEF'  # Heart Failure with Mid-range Ejection Fraction
    else:
        return 'HFpEF'  # Heart Failure with Preserved Ejection Fraction

# Example usage
lvef_value =
heart_failure_category = categorize_heart_failure(lvef_value)

print(heart_failure_category)
```

What is the Type of Heart Failure of the patient in the following text? Please answer as "Answer: The Type of Heart Failure: "
“A 58-year-old ...”

Figure 4: Code Base Prompt (CBP) for Heart Failure classification by Left Ventricular Ejection Fraction (LVEF).

is unclear whether ChatGPT actually applied the algorithm in its thought process or simply stitched together appropriate sentences [8]. Furthermore, even if F1, F3, and F4 are indicated in ChatGPT’s text response, it is unclear whether they were actually used in the decision-making process.

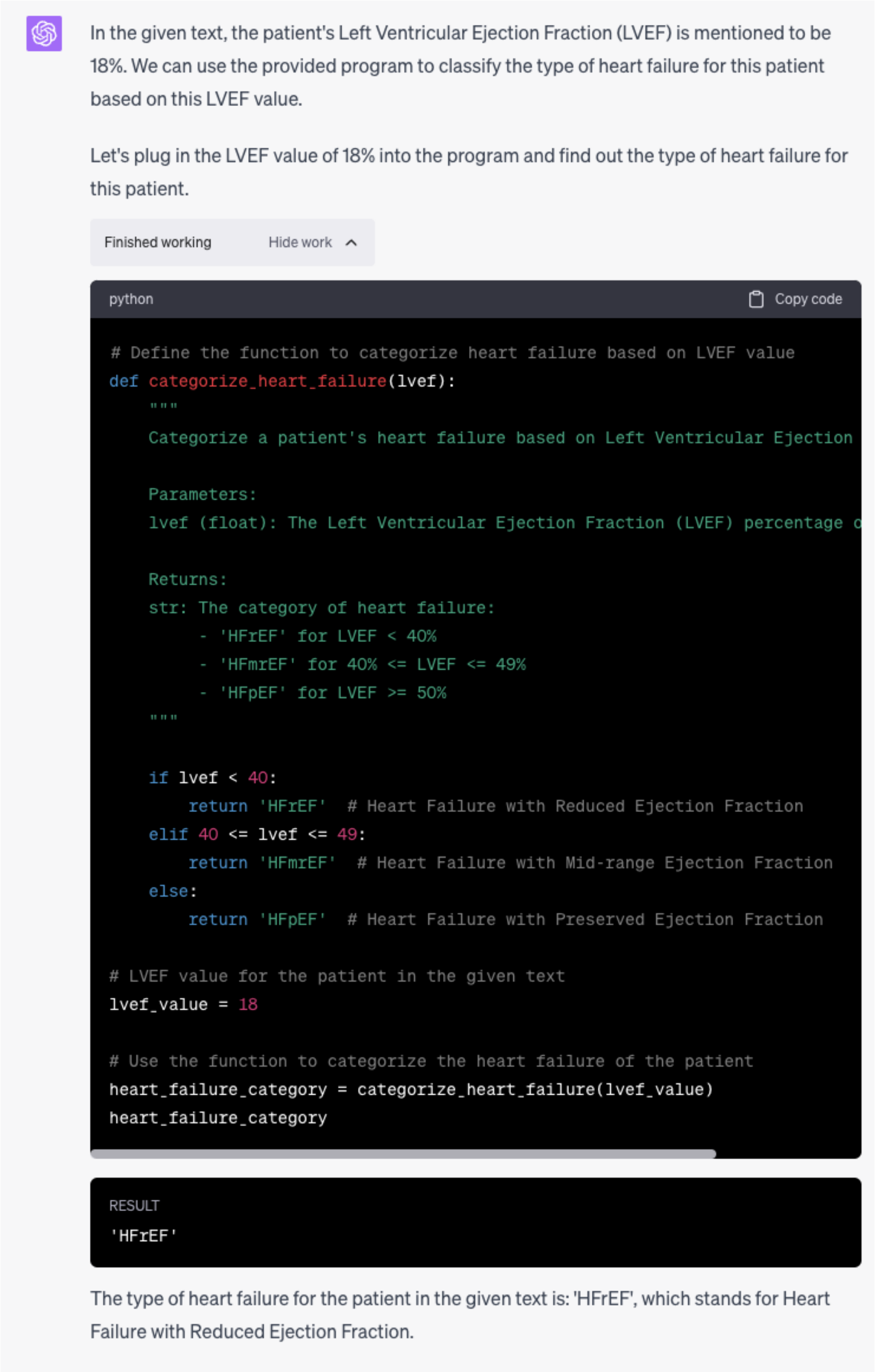
In the case of CBP, because the ChatGPT Code interpreter represents the medical algorithm in Python code, if the Python code accurately represents the human algorithmic thought process, the human can explicitly verify MAPC factor F1 [13]. Next, as shown in Figure 5, input and output values are displayed on the ChatGPT Code interpreter’s execution screen if the Python code is executed. The execution of the Python algorithm allows for the verification of F2 in MAPC, the input values allow for the verification of F3, and the execution result allows for the verification of F4 [13]. Similar to TBP, even in CBP, F5 in MAPC can also be verified by looking at ChatGPT’s text-based answer. Note that if Python code execution does not take place, F1, F2, F3, and F4 cannot be verified, similar to what happens with TBP.

5 EXPERIMENT

In this section, we evaluate the MAPC in both CBP and TBP when applied to three case reports of heart failure.

The cases used for this experiment are shown in Table 1. Case 1 utilizes the full text from the Case Presentation section of [18]. In this text, the LVEF is measured twice, noted as 11% and 20%, and in both instances, the patient is classified as HFrEF in the paper. Case 2 uses the full text from the Case Report section of [19]. The LVEF is also measured twice, recorded as 67% and 70%, and the patient is classified as HFpEF in the paper. Case 3 uses the full text from the Case Report section of [20], where the LVEF is noted as 50%, and the patient is classified as HFpEF in the paper.

The results of applying ChatGPT Code Interpreter to these three cases based on both TBP and CBP are presented in Table 2. In the case of CBP, the Python code was executed for all cases, and the input values, output values, and classifications were all correctly displayed. In the case of TBP, no program execution took place, and thus, F1, F2, F3, and F4 in MAPC could not be verified. However, the classification was correctly performed, and F5 was verified.



In the given text, the patient's Left Ventricular Ejection Fraction (LVEF) is mentioned to be 18%. We can use the provided program to classify the type of heart failure for this patient based on this LVEF value.

Let's plug in the LVEF value of 18% into the program and find out the type of heart failure for this patient.

```python
# Define the function to categorize heart failure based on LVEF value
def categorize_heart_failure(lvef):
    """
    Categorize a patient's heart failure based on Left Ventricular Ejection

    Parameters:
    lvef (float): The Left Ventricular Ejection Fraction (LVEF) percentage o

    Returns:
    str: The category of heart failure:
         - 'HFrEF' for LVEF < 40%
         - 'HFmrEF' for 40% <= LVEF <= 49%
         - 'HFpEF' for LVEF >= 50%
    """

    if lvef < 40:
        return 'HFrEF'  # Heart Failure with Reduced Ejection Fraction
    elif 40 <= lvef <= 49:
        return 'HFmrEF'  # Heart Failure with Mid-range Ejection Fraction
    else:
        return 'HFpEF'  # Heart Failure with Preserved Ejection Fraction

# LVEF value for the patient in the given text
lvef_value = 18

# Use the function to categorize the heart failure of the patient
heart_failure_category = categorize_heart_failure(lvef_value)
heart_failure_category
```

```
'HFrEF'
```

The type of heart failure for the patient in the given text is: 'HFrEF', which stands for Heart Failure with Reduced Ejection Fraction.

Figure 5: Python code generation result from ChatGPT Code interpreter by Code Base Prompt (CBP) for Heart Failure classification by Left Ventricular Ejection Fraction (LVEF).

6 DISCUSSION

First, we discuss the advantages and disadvantages of our proposed method. Our proposed method is unique in that it uses the ChatGPT Code interpreter to make ChatGPT XAI in the application of medical algorithms to medical texts and checks its XAI capability with new criteria called MAPC. The advantage of using the ChatGPT Code interpreter is objectivity, as the Python code is executed mechanically, and humans can easily know MAPC by simply checking the code execution. The first disadvantage is that ChatGPT does not work reliably for prompts, so the case when MAPC is satisfied or

Table 1: Cases Information [18] [19] [20].

	Case 1 [18]	Case 2 [19]	Case 3 [20]
Text Source	Case Presentation Section	Case Report Section	Case Report Section
LVEF	11% or 20%	67% or 70%	50%
Correct Answer	HFrEF	HFpEF	HFpEF

Table 2: Results of Code Base Prompt (CBP) and Text Base Prompt (TBP) evaluated by Medical Algorism Presentation Criteria (MAPC).

	Case 1 [18]	Case 2 [19]	Case 3 [20]
F1 of CBP Result	✓	✓	✓
F2 of CBP Result	✓	✓	✓
F3 of CBP Result	✓	✓	✓
F4 of CBP Result	✓	✓	✓
F5 of CBP Result	✓	✓	✓
F1 of TBP Result	×	×	×
F2 of TBP Result	×	×	×
F3 of TBP Result	×	×	×
F4 of TBP Result	×	×	×
F5 of TBP Result	✓	✓	✓

not depends on the task and prompt, and the ability of the prompt cannot be determined in advance. The second disadvantage is that when creating CBP from TBP, we use ChatGPT to generate and utilize Python code. This requires careful evaluation of the Python code by the techniques to prevent inappropriate code [21] [22].

Second, we discuss the results. In the results of this study, when medical text from case report papers was given for the heart failure classification algorithm [17], it was confirmed that the ChatGPT Code interpreter executed Python code for all case reports when using CBP, thus satisfying all criteria from F1 to F5 in MAPC. In contrast, in the case of TBP, the ChatGPT Code interpreter did not execute Python code, and therefore, F1, F2, F3, and F4 in MAPC could not be verified. This is likely because the ChatGPT Code Interpreter generally executes programs when required by programming code, and medical text prompts usually do not need program execution. However, in CBP, Python program execution was implicitly required in the form of Python code, causing the ChatGPT Code interpreter to execute the code.

Third, we discuss the accuracy of the ChatGPT response. The primary objective of our research is to verify the explainability of CBP through MAPC, contributing to the field of XAI. However, it is challenging to statistically evaluate the accuracy of the execution due to the limited number of test cases. A possible direction for future research is to evaluate the accuracy along with the performance of XAI based on MAPC, but this would require validation on a larger data set.

At last, we note the limitations. In this study, we focused on tasks that classify heart failure according to the LVEF classification algorithm. However, it is unclear whether the ChatGPT Code interpreter will provide Python code that meets MAPC for other medical algorithms.

7 RELATED WORKS

In this section, we compare this study with other works.

7.1 XAI techniques

XAI is AI that has been designed to describe its objectives, rationale, and decision-making process in a way that can be understood. The purpose of XAI is to make the inferences of AI algorithms, which are often black boxes, understandable to human users and to increase trust. It has been proposed that XAI techniques can be classified into four categories: (i) data explainability, (ii) model explainability, (iii) post-hoc explainability, and (iv) assessment of explanations [23].

Our research differs from existing XAI techniques in that it specifically focuses on improving post-hoc explainability in ChatGPT by ChatGPT Code interpreter. Furthermore, regarding the assessment of explanations, we propose criteria for judging XAI capability in the task of applying medical algorithms in medical text information.

7.2 XAI by LLM

XAI techniques application for interpretation and understanding of LLM operations and outputs have been proposed [6] [7]. These techniques are expected to enhance the transparency and reliability of AI. Efforts have also been proposed in the context of ChatGP [24].

On the other hand, we leverage ChatGPT Code interpreter to achieve XAI.

7.3 LLM for medicine

The use of LLM in the medical field is expanding across various domains, including clinical application, research application, and education [25]. In clinical application, LLM plays a role in assisting the physician's work. In medical research, LLM can make research activities more efficient by helping to create abstracts, provide explanations, and even suggest new research directions. Furthermore, in medical education, LLM can serve as an auxiliary tools for medical students searching for information or learning. While the use of LLM in the medical field holds much promise, there are also risks and challenges, such as issues of information accuracy, overreliance, hallucination, and ethical considerations.

In the medical LLM study field, our research especially focuses on ChatGPT Code interpreter.

7.4 ChatGPT Code interpreter base analysis

ChatGPT Code Interpreter [15] became available as a beta feature on ChatGPT in 2023. There have been reported applications in some fields, such as mathematics [26] and bioinformatics [27]. These studies mainly focus on improving accuracy.

On the other hand, studies of XAI by ChatGPT Code Interpreter are scarce.

8 CONCLUSION

In this study, we aimed to make the decision-making process of ChatGPT transparent by ChatGPT Code interpreter for the medical algorithm application on a medical text. For this purpose, we proposed using CBP to encourage the execution of Python code by ChatGPT Code Interpreter. Furthermore, we introduced MAPC as an evaluation metric to assess the explainability of the ChatGPT's decision-making process. We tested our approach on the task of applying the heart failure classification algorithm to medical texts. Three case studies on heart failure were assessed using both CBP and TBP. The results revealed that, with CBP, Python code was executed 100% of the time, and all the five factors of MAPC were met. On the other hand, with TBP, Python code was not executed, and only one criterion (F5) of MAPC was met, leaving F1 to F4 unfulfilled. This study presents an XAI technique that makes ChatGPT responses understandable to humans when a medical algorithm is applied to medical text.

For future work, we aim to explore the trade-off between MAPC and accuracy by using larger datasets. In addition, we plan to investigate other medical algorithms besides heart failure classification.

REFERENCES

[1] Spyros Makridakis. 2017. The forthcoming Artificial Intelligence (AI) revolution: Its impact on society and firms. Futures, 90 (July 2017), 46-60. https://doi.org/10.1016/j.futures.2017.03.006

[2] Andres Morales-Forero, Samuel Bassetto, and Eric Coatanea. 2023. Toward safe AI. AI & SOCIETY 38, 2 (November 2023), 685-696. https://doi.org/10.1007/s00146-022-01591-z

[3] Rudresh Dwivedi, Devam Dave, Het Naik, Smiti Singhal, Rana Omer, Pankesh Patel, Bin Qian, Zhenyu Wen, Tejal Shah, Graham Morgan, and Rajiv Ranjan. 2023. Explainable AI (XAI): Core ideas, techniques, and solutions. ACM Comput. Surv. 55, 9 (January 2023), 1-33. https://doi.org/10.1145/3561048

[4] Jesse G. Meyer, Ryan J. Urbanowicz, Patrick C. N. Martin, Karen O'Connor, Ruowang Li, Pei-Chen Peng, Tiffani J. Bright, Nicholas Tatonetti, Kyoung Jae Won, Graciela Gonzalez-Hernandez, and Jason H. Moore. 2023. ChatGPT and large language models in academia: opportunities and challenges. BioData Min. 16, 20 (July 2023). https://doi.org/10.1186/s13040-023-00339-9

[5] Teresa Datta and John P. Dickerson. 2023. Who's Thinking? A Push for Human-Centered Evaluation of LLMs using the XAI Playbook. arXiv preprint arXiv:2303.06223. https://doi.org/10.48550/arXiv.2303.06223

[6] Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Brian Ichter, Fei Xia, Ed Chi, Quoc Le, and Denny Zhou. 2022. Chain-of-thought prompting elicits reasoning in large language models. arXiv preprint. arXiv:2201.11903. https://doi.org/10.48550/arXiv.2201.11903

[7] Hunter Lightman, Vineet Kosaraju, Yura Burda, Harri Edwards, Bowen Baker, Teddy Lee, Jan Leike, John Schulman, Ilya Sutskever, and Karl Cobbe. 2023. Let's Verify Step by Step. arXiv preprint arXiv:2305.20050. https://doi.org/10.48550/arXiv.2305.20050

[8] Ruixiang Tang, Yu-Neng Chuang, and Xia Hu. 2023. The science of detecting llm-generated texts. arXiv preprint arXiv:2303.07205. https://doi.org/10.48550/arXiv.2303.07205

[9] Pavel Hamet and Johanne Tremblay. 2017. Artificial intelligence in medicine. Metab. 69, Supplement (April 2017), S36-S40. https://doi.org/10.1016/j.metabol.2017.01.011

[10] Juan Manuel Durán and Karin Rolanda Jongsma. 2021. Who is afraid of black box algorithms? On the epistemological and ethical basis of trust in medical AI. J. Med. Ethics 47, 5 (March 2021), 329-335. http://dx.doi.org/10.1136/medethics-2020-106820

[11] Ruey-Kai Sheu and Mayuresh Sunil Pardeshi. 2022. A Survey on Medical Explainable AI (XAI): Recent Progress, Explainability Approach, Human Interaction and Scoring System. Sensors 22, 20 (October 2022), 8068. https://doi.org/10.3390/s22208068

[12] Zihan Guan, Zihao Wu, Zhengliang Liu, Dufan Wu, Hui Ren, Quanzheng Li, Xiang Li, and Ninghao Liu. 2023. CohortGPT: An Enhanced GPT for Participant Recruitment in Clinical Study. arXiv preprint arXiv:2307.11346. https://doi.org/10.48550/arXiv.2307.11346

[13] Pedro Lopes, Eduardo Silva, Cristiana Braga, Tiago Oliveira, and Luís Rosado. 2022. XAI Systems Evaluation: A Review of Human and Computer-Centred Methods. Appl. Sci. 12, 19 (September 2022), 9423. https://doi.org/10.3390/app12199423

[14] Dongwen Wang, Mor Peleg, Samson W Tu, Aziz A Boxwala, Robert A Greenes, Vimla L Patel, and Edward H Shortliffe. 2002. Representation primitives, process models and patient data in computer-interpretable clinical practice guidelines: A literature review of guideline representation models. Int. J. Med. Inform. 68, 1-3 (December 2002), 59-70. https://doi.org/10.1016/S1386-5056(02)00065-5

[15] ChatGPT plugins. OpenAI. 2023. Retrieved Aug 24, 2023 from https://openai.com/blog/chatgpt-plugins

[16] Véronique L. Roger. 2013. Epidemiology of heart failure. Circ. Res. 113, 6 (August 2013), 646-659. https://doi.org/10.1161/CIRCRESAHA.113.300268

[17] Amy Groenewegen, Frans H. Rutten, Arend Mosterd, and Arno W. Hoes. 2020. Epidemiology of heart failure. Eur. J. Heart Fail. 22, 8 (June 2020), 1342-1356. https://doi.org/10.1002/ejhf.1858

[18] Loai Almazroa, Vesna Mihajlovic, Patrick R. Lawler, and Adriana Luk. 2020. Crossing the chasm: caution for use of angiotensin receptor-neprilysin inhibition in patients with cardiogenic shock–a case report. EHJ – Case Reports 4, 6 (October 2020), 1-4. https://doi.org/10.1093/ehjcr/ytaa233

[19] Nikhil Narang, Diego Medvedofsky, Kathryn Dryer, Sanjiv J. Shah, Charles J. Davidson, Amit R. Patel, and John E.A. Blair. 2017. Microvascular dysfunction and cardiac fibrosis in heart failure with preserved ejection fraction: a case report. ESC heart Fail. 4, 4 (June 2017), 645-648. https://doi.org/10.1002/ehf2.12170

[20] Elena Leonidou, Maria Ioannou, Petros Mavrommatis, and Angeliki Mouzarou. 2023. A case report of a patient with heart failure with preserved ejection fraction presented as dysphagia. ESC heart Fail. 10, 4 (May 2023), 2707-2710. https://doi.org/10.1002/ehf2.14415

[21] Lamia Ketari, and Mohammadi Akheela Khanum. 2012. A Review of Malicious Code Detection Techniques for Mobile Devices. IJCTE. 4, 2 (April 2012), 212-216. https://doi.org/10.7763/IJCTE.2012.V4.453

[22] Upul Bandara, and Gamini Wijayrathna. 2012. Detection of Source Code Plagiarism Using Machine Learning Approach. IJCTE. 4, 5 (October 2012), 674-678. https://doi.org/10.7763/IJCTE.2012.V4.555

[23] Sajid Ali, Tamer Abuhmed, Shaker El-Sappagh, Khan Muhammad, Jose M. Alonso-Moral, Roberto Confalonieri, Riccardo Guidotti, Javier Del Ser, Natalia Díaz-Rodríguez, and Francisco Herrera. 2023. Explainable Artificial Intelligence (XAI): What we know and what is left to attain Trustworthy Artificial Intelligence. Inf. Fusion. 99 (November 2023), 101805. https://doi.org/10.1016/j.inffus.2023.101805

[24] Lichang Chen, Jiuhai Chen, Tom Goldstein, Heng Huang, and Tianyi Zhou. 2023. InstructZero: Efficient Instruction Optimization for Black-Box Large Language Models. arXiv preprint arXiv:2306.03082. https://doi.org/10.48550/arXiv.2306.03082

[25] Arun James Thirunavukarasu, Darren Shu Jeng Ting, Kabilan Elangovan, Laura Gutierrez, Ting Fang Tan, and Daniel Shu Wei Ting. 2023. Large language models in medicine. Nat. Med. 29 (July 2023), 1930–1940. https://doi.org/10.1038/s41591-023-02448-8

[26] Aojun Zhou, Ke Wang, Zimu Lu, Weikang Shi, Sichun Luo, Zipeng Qin, Shaoqing Lu, Anya Jia, Linqi Song, Mingjie Zhan, and Hongsheng Li. 2023. Solving Challenging Math Word Problems Using GPT-4 Code Interpreter with Code-based Self-Verification. arXiv preprint arXiv:2308.07921. https://doi.org/10.48550/arXiv.2308.07921

[27] Lei Wang, Xijin Ge, Li Liu, and Gangqing Hu. 2023. Code Interpreter for Bioinformatics: Are We There Yet?. Ann. Biomed. Eng., 2023 (July 2023), 1-3. https://doi.org/10.1007/s10439-023-03324-9