

【研究ノート】

生成 AI を活用した未知の領域の学習への試み**—量子コンピューティングを例にして—**

廣峯優希・富山哲男・宮崎淳*

**Attempts to learn the unknown by using generative AI
: For example, Quantum Computing**

Yuki Hiromine, Tetsuo Tomiyama, and Jun Miyazaki

Abstract: This paper concentrates on the application of generative AI in learning quantum computing. It discusses guidelines, usage examples, and precautions for beginners in quantum computing by utilizing generative AI. Moreover, it presents instances of utilizing generative AI in learning quantum computing, offering objective perspectives on its potential uses in various research and educational fields.

Keywords: Quantum Computing, generative AI, QAOA

1. はじめに

2022 年に OpenAI が発表した ChatGPT[1] をきっかけに、世界的な生成 AI ブームが巻き起こっている。生成 AI は、個人的な使用を目的とするユーザだけでなく、教育や研究などにも活用されている。文部科学省では、2023 年の 7 月 4 日に“初等中等教育段階における生成 AI の利用に関する暫定的なガイドライン”[2] を公開しており、国内でも政府が主導して生成 AI を活用しようとする動きが見られる。

2. 生成 AI を使って未知の領域の学習・研究にチャレンジする—量子コンピューティングを例として—

本論文では、2023 年 11 月現在公開されている生成 AI のうち、OpenAI の ChatGPT[1] と、Microsoft の BingAI[3]、Google の Google Bard[4] を使用して、量子コンピューティングの学問に触れたことがない初学者を支援し、生成 AI の活用方法についてのガイドラインを提供するとともに、今回の実験に使用する各々の生成 AI の性能の比較について考察を行うことを目的としている。

2011 年にカナダの D-Wave System 社が世界初の商用量子コンピュータを販売 [5] して以

* 東京国際工科専門職大学客員教授、株式会社オレンジテクラボ

降、量子コンピューティングは世界中でめざましい研究開発が行われている。さらに、量子コンピュータは量子力学の重ね合わせを利用 [6] し、一部の問題で計算量が減り、計算が高速化するマシンとして期待されている。[7] しかしながら、現在一般的に使用されている従来のコンピュータ（以降「古典コンピュータ」とよぶ）と量子コンピュータを比較すると、量子コンピュータはハードウェアの制約や研究開発段階といったこともあり、古典コンピュータを超える計算速度を誇るほどの実用化には至っていない。また、量子コンピューティングは、量子力学の理論をもとに研究されているため、量子力学の理論が完全でないことから未知の領域を含んでいることに加え、人間が直感で理解できる古典力学とは理論が大きく異なってくる。今回は、このような未知の領域の学習・研究にチャレンジし、そこで行った実験や学習の成果について考察をしていく。

3. 量子計算の基礎

量子コンピュータを使った量子計算の基礎について、一般的に使用されている古典コンピュータと比較して、2つの項目に分けて説明する。

3.1 情報の最小単位（ビット）

古典コンピュータは、電圧の高低を利用して、電圧が低いときは0、電圧が高いときは1として古典ビットを表現してきた。量子コンピュータの場合は、電子スピンの状態 [8] によって量子ビットを表現する。古典ビットと量子ビットの表現について、以下の図1に示している。






| 古典ビット | 量子ビット |
|---|--|
| <p>【定義】 電圧の高低によって0,1の状態が表される。</p> <div style="display: flex; justify-content: space-around; align-items: center;"> <div style="text-align: center;">  低電圧“0” </div> <div style="text-align: center;">  高電圧“1” </div> </div> | <p>【定義】 電子スピン(角運動量)の状態によって、重ね合わせの状態が表される。</p> <div style="display: flex; justify-content: space-around; align-items: center;"> <div style="text-align: center;">  アップスピン“ 0>” </div> <div style="text-align: center;">  ダウンスピン“ 1>” </div> <div style="text-align: center;">  重ね合わせ“ ψ>” </div> </div> |

図1. 古典ビットと量子ビットの表現

以上の図1より、古典コンピュータでの古典ビット“0”,”1”という表現は、量子ビットでは“ $|0\rangle$ ”,” $|1\rangle$ ”に対応する。また、量子ビットでは量子力学の原理により重ね合わせの状態をとることができる。重ね合わせによって表された量子ビットの状態 $|\psi\rangle$ は以下のよう表される。

$$|\psi\rangle = \alpha|0\rangle + \beta|1\rangle$$

ここで、 α, β は複素数で $|\alpha|^2 + |\beta|^2 = 1$ である。また $|\psi\rangle$ は量子状態を表す複素ベクトルである。

ここでいう“原理”とは、他の基本原理から導出できないもので、それに反する実験事実もないことから、それを疑っても仕方がないという意味であるため、量子力学の原理を受け入れて理解を進める必要があることに注意しなければならない。

さらに、それぞれの量子ビット $|0\rangle, |1\rangle$ について、以下で定義される。

$$|0\rangle := \begin{pmatrix} 1 \\ 0 \end{pmatrix}, \quad |1\rangle := \begin{pmatrix} 0 \\ 1 \end{pmatrix}$$

3.2 論理回路とその出力パターン

古典コンピュータには古典回路、量子コンピュータには量子回路と呼ばれるものがある。量子計算では次のように定義されるテンソル積が使用される。

$$U \otimes V := \begin{pmatrix} u_{00}V & u_{01}V \\ u_{10}V & u_{11}V \end{pmatrix} = \begin{pmatrix} u_{00}v_{00} & u_{00}v_{01} & u_{01}v_{00} & u_{01}v_{01} \\ u_{00}v_{10} & u_{00}v_{11} & u_{01}v_{10} & u_{01}v_{11} \\ u_{10}v_{00} & u_{10}v_{01} & u_{11}v_{00} & u_{11}v_{01} \\ u_{10}v_{10} & u_{10}v_{11} & u_{11}v_{10} & u_{11}v_{11} \end{pmatrix}$$

また、テンソル積は以下の性質を満たす。

$$\alpha(U \otimes V) = (\alpha U) \otimes V = U \otimes (\alpha V)$$

$$(U_0 + U_1) \otimes V = U_0 \otimes V + U_1 \otimes V$$

$$U_0 \otimes (V_0 + V_1) = U_0 \otimes V_0 + U_0 \otimes V_1$$

これらのテンソル積の定義と性質を踏まえた上で、古典回路と量子回路の違いについて以下の図2で示す。



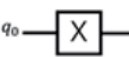
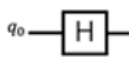

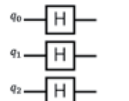
| 古典回路 | 量子回路 |
|---|---|
| 【回路記号】 論理積(AND)  論理否定(NOT)  ...など | 【量子ゲート】 Xゲート(NOTゲート)  Hゲート(アダマールゲート)  ※ただし、 $ 0\rangle = \begin{pmatrix} 1 \\ 0 \end{pmatrix}$, $ 1\rangle = \begin{pmatrix} 0 \\ 1 \end{pmatrix}$...など |
| 【論理式】 論理積(AND) $X = A \cdot B$ 論理否定(NOT) $X = \bar{A}$ | 【行列表現】 Xゲート(NOTゲート) $q_0 \otimes X = q_0\rangle \otimes \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}$ Hゲート(アダマールゲート) $q_0 \otimes H = q_0\rangle \otimes \frac{1}{\sqrt{2}} \begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix}$ |
| ・古典回路の例(3 bit) $A=0, B=0, C=0$ のとき  $X = AB + \bar{A}\bar{B} + C$ $X = 0 \cdot 0 + 1 \cdot 1 + 0$ $X = 1$ よって、出力は“X=1” | ・量子回路の例(3 qubit) $q_0 = 0\rangle, q_1 = 0\rangle, q_2 = 0\rangle$  $(q_0 \otimes H) \otimes (q_1 \otimes H) \otimes (q_2 \otimes H)$ $= (0\rangle \otimes H) \otimes (0\rangle \otimes H) \otimes (0\rangle \otimes H)$ $= \frac{1}{\sqrt{8}} (000\rangle + 001\rangle + 010\rangle + 011\rangle + 100\rangle + 101\rangle + 110\rangle + 111\rangle)$ よって、量子ビットの状態は“8つの状態の重ね合わせとなる” |

図2. 古典回路と量子回路の違い

以上の図2では、古典回路と量子回路における回路の表し方や数学的表現、回路の例について表している。古典回路と量子回路の決定的な違いとして、古典回路はある入力に対して出力は1つに定まるが、量子回路ではある入力に対し、量子ビットの状態は確率的表現で表されることである。

4. 量子アルゴリズム

量子計算におけるアルゴリズムの意義について、Quantum Native Dojo[9] に解説がある。

【第2章 量子アルゴリズム入門】

量子コンピュータは、量子力学的な重ね合わせによって、 n 個の量子ビットを用いて 2^n 個の状態を同時に処理できる。しかし、これだけでは「計算が速い」ということにはならない。なぜなら、計算終了後に結果を観測する際に、 2^n 個の状態の内どれか一つがランダムに得られるのみだからである。したがって、欲しい答えが高確率で得られるように設計された、量子コンピュータ専用のアルゴリズムが不可欠である。そのようなアルゴリズムを量子アルゴリズムと呼ぶ。

このことから、組み合わせ最適化問題の解を求めるためには、量子アルゴリズムを使用する必要がある。

量子計算で使用されるアルゴリズムについて、Quantum Algorithm Zoo[10] で 2022 年 6 月 26 日時点において約 60 種類ほど公開されている。[10] で公開されている量子アルゴリズムに関して、以下の 4 つの種類に分類されている。

1. 代数的整数論アルゴリズム (Algebraic & Number Theoric)
2. オラクルアルゴリズム (Oracular)
3. 近似アルゴリズム (Approximation and Simulation)
4. 機械学習と数値最適化 (Optimization, Numerics & Machine Learning)

ここで、量子計算で使用されるアルゴリズムについて、3 に分類される QAOA(Quantum Approximation Optimization Algorithm)[11] を例に解説していく。QAOA は量子ゲート方式において、組み合わせ最適化問題を解くためのアルゴリズムの一つである。このアルゴリズムの理論を、“IBM Quantum で学ぶ量子コンピュータ [12]” から引用したものを以下に示す。

【QAOA の手順】

- (i) 初期状態 $|s\rangle = |+\rangle^{\otimes n}$ を用意する
- (ii) パラメータ $\{\beta_1, \dots, \beta_l\}$ と $\{\gamma_1, \dots, \gamma_l\}$ で指定される 2 つの入力 $U_c(\gamma_i), U_x(\beta_i)$ を作用させ状態 $|s\rangle \rightarrow |\beta, \gamma\rangle$ を得る。
- (iii) 量子コンピュータで期待値 $\langle \beta, \gamma | C(Z) | \beta, \gamma \rangle$ を評価 (測定) する。
- (iv) $\langle \beta, \gamma | C(Z) | \beta, \gamma \rangle$ を小さくするように古典コンピュータでパラメータ β, γ を更新する。
- (v) 最適解 β^*, γ^* を得るまで (i) ~ (iv) を繰り返す。
- (vi) 何回か状態 $|\beta^*, \gamma^*\rangle$ を Z 基底で測定し、良さそうな測定結果 (ビット列 z_1, \dots, z_n を解として出力する。

QAOA のアルゴリズムの手順は以上のようにになっているが、これでは理解が難しいので、先ほどの QAOA の手順の一部を以下の図 3[13] に対応させる。QAOA の手順における (i) ~ (v) を以下の図 3 の "1 ~ 5" に対応させ、QAOA の手順 (vi) を最終的な出力結果として表す。

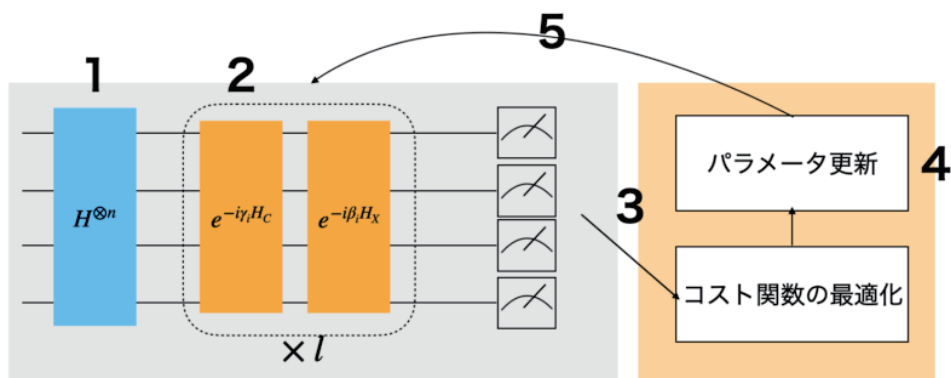


図 3.QAOA アルゴリズムの手順 [13]

この図 3[13] を用いて、量子アルゴリズムについて例題を以下で取り扱う。例題として、ここでは最大カット問題を取り扱う。最大カット問題は、重みつきグラフの辺を 2 つのグループに分割する問題で、グループ分けをした際に分割される辺の数を最大化するものである [14]。今回は、以下の図 4 ~ 図 7 を例に説明する。

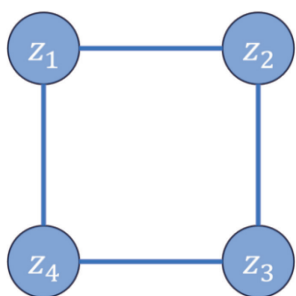


図 4. 無向グラフ

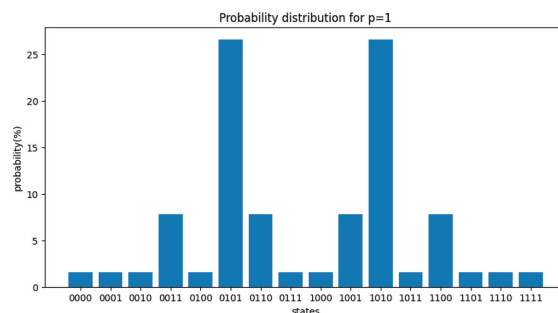


図 5. 2 つの入力 $(\gamma_i), U_X(\beta_i)$ を 1 度作用させたもの

この図 4 のグラフは、初期状態 (i) を表している。ここで、2 つの入力 $(\gamma_i), U_X(\beta_i)$ を作用 (ii) させると、図 5 のような結果が観測される (iii)。さらに、2 つの入力 $(\gamma_i), U_X(\beta_i)$ を作用させる回数を増やす (iv)(v) と、図 6 のように最適解を得ることができる (vi)。

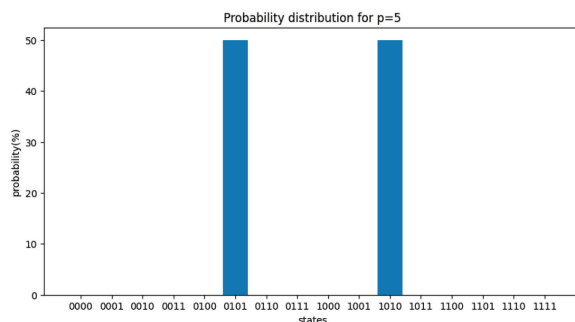
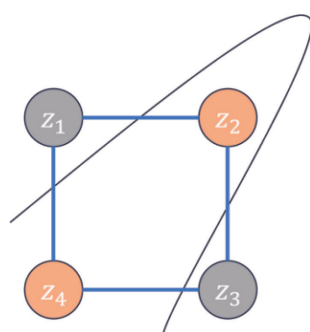
図 6. 2つの入力 (γ_i), $U_X(\beta_i)$ を複数回作用させたもの

図 7. 得られた最適解

ここで図 6 と図 7 をみると、図 6 で $|0101\rangle$ と $|1010\rangle$ が得られた。この 2 つの最適解のうち $|0101\rangle$ の解を用いると、 z_1 と z_3 のグループと z_2 と z_4 のグループに分けて、図 7 のように辺を 4 つに分割すると、辺の分割を最大化することができる。

このように、量子計算特有の重ね合わせを利用することで、身近な問題の最適解を求めることができる。

5. 生成 AI の自己学習への活用

筆者の一人（廣峯）は、2023 年 10 月 2 日から同年 11 月 14 日まで「生成 AI を活用して量子コンピューティングを自己学習する」ことをテーマにインターンシップの体験をした。その中で、全 6 週間におよぶインターンシップの中で生成 AI を活用した場面は、以下のとおりである。

1. 6 週間の実習計画表の作成
2. 生成 AI を使った Python コードの生成

上記の 1,2 について、どのように生成 AI を活用してどのような結果が得られたかについて、以下で説明する。

5.1 6 週間の実習計画表の作成

まず、6 週間分の実習計画を立てるために、ChatGPT(GPT-4.0)[1] を活用して、実習計画

表の作成を行なった。本インターンシップでは以下のプロンプト (図 8) に対する出力結果 (図 9) をベースとして、実習計画書の進捗チェックリストを作成し、図 10 の実習計画表に従って実習を進めていった。



6週間のインターンシップを実施します。
その6週間分のインターン実施計画を作成してください。表や図を使っても良いです。
内容は主に量子コンピュータを扱い、その分野の中でQAOAのような量子ゲートにおける最適化の学習や、ChatGPT(GPT-3.5)やBingAI(GPT-4.0)を用いた量子ゲートにおけるプログラミングの支援に関する事柄について学習します。

図 8. 実習計画書の作成にあたって使用したプロンプト

| | |
|---|--|
| <p>週1-2: 量子コンピューティング基礎</p> <ul style="list-style-type: none"> 日程: 週1-2 内容: <ul style="list-style-type: none"> 量子コンピューティングの基本概念の学習 量子ビットと量子ゲートの理解 Qiskitなどの量子コンピューティングフレームワークの導入 <p>週3-4: 量子ゲート最適化 (QAOA)</p> <ul style="list-style-type: none"> 日程: 週3-4 内容: <ul style="list-style-type: none"> Quantum Approximate Optimization Algorithm (QAOA) の理論と実装の学習 QAOAを使用した最適化問題の解決方法の習得 QAOAの実際の問題への適用 <p>週5-6: ChatGPTとBingAIの活用</p> <ul style="list-style-type: none"> 日程: 週5-6 内容: <ul style="list-style-type: none"> ChatGPT (GPT-3.5) およびBingAI (GPT-4.0) の理解と機能の調査 量子ゲートプログラミング支援用のGPTモデルのカスタマイズ方法の学習 プロジェクトにおけるChatGPTまたはBingAIの活用 | <p>週7-8: プロジェクトの準備と設計</p> <ul style="list-style-type: none"> 日程: 週7-8 内容: <ul style="list-style-type: none"> インターンシッププロジェクトの具体的な目標の設定 必要なリソースとデータの収集 プロジェクトの計画と設計の作成 <p>週9-10: プロジェクト実施と実験</p> <ul style="list-style-type: none"> 日程: 週9-10 内容: <ul style="list-style-type: none"> プロジェクトの実施と実験の開始 ChatGPTまたはBingAIを使用した量子ゲートプログラミング支援の実際の適用 問題の特定と解決 <p>週11-12: プロジェクト評価と成果報告</p> <ul style="list-style-type: none"> 日程: 週11-12 内容: <ul style="list-style-type: none"> プロジェクトの評価と結果の分析 結果をドキュメント化し、成果報告の準備 インターンシップの成果を発表するためのプレゼンテーションの準備 |
|---|--|

図 9. 図 8 のプロンプトに対する出力

| | |
|--|---|
| <p>進捗チェックシート</p> <p>週1: 量子コンピューティング入門</p> <ul style="list-style-type: none"> Day 1-3: 量子コンピューティングの基本概念を学ぶ <ul style="list-style-type: none"> <input type="checkbox"/> 基本的な量子力学の概念の理解 <input type="checkbox"/> 量子計算モデルの基礎を把握 Day 4-7: 量子ゲートと量子ビットの操作に関する理解を深める <ul style="list-style-type: none"> <input type="checkbox"/> 主要な量子ゲートを学ぶ <input type="checkbox"/> 量子回路の構築と実験 <p>週2: QAOAの理論</p> <ul style="list-style-type: none"> Day 8-10: QAOAの理論を学ぶ <ul style="list-style-type: none"> <input type="checkbox"/> QAOAの基本的な仕組みの理解 <input type="checkbox"/> QAOAの利点と制限を把握 Day 11-14: QAOAアルゴリズムの詳細な理解と数学的背景の探求 <ul style="list-style-type: none"> <input type="checkbox"/> アルゴリズムのステップごとの動作の理解 <input type="checkbox"/> 重要な数学的な式や定理の理解 <p>週3: QAOAの実装</p> <ul style="list-style-type: none"> Day 15-17: QiskitやCirqを使用してQAOAの基本的な実装方法を学ぶ <ul style="list-style-type: none"> <input type="checkbox"/> QiskitやCirqの基本的な操作を学ぶ <input type="checkbox"/> QAOAのシンプルな実装を試す Day 18-21: サンプル問題を使用してQAOAの実装と最適化に取り組む <ul style="list-style-type: none"> <input type="checkbox"/> サンプル問題の選定と定義 <input type="checkbox"/> QAOAの実装と結果の評価 | <p>週4: ChatGPT (GPT-3.5) の活用</p> <ul style="list-style-type: none"> Day 22-24: ChatGPTの基本的な使用法を学び、自然言語処理の概念を理解する <ul style="list-style-type: none"> <input type="checkbox"/> ChatGPTのインターフェースの使い方を学ぶ <input type="checkbox"/> 自然言語処理の基本的な理解 Day 25-28: ChatGPTを活用して量子コンピューティングに関する質問や課題を解決する <ul style="list-style-type: none"> <input type="checkbox"/> 質問のリストアップ <input type="checkbox"/> ChatGPTを使用して質問を解決 <p>週5: BingAI (GPT-4.0) の活用と量子最適化への応用</p> <ul style="list-style-type: none"> Day 29-31: BingAIの高度な機能と自然言語生成の学習 <ul style="list-style-type: none"> <input type="checkbox"/> BingAIの主な特長を学ぶ <input type="checkbox"/> 自然言語生成の応用例を試す Day 32-35: BingAIを使用してQAOAや量子最適化に関する複雑な問題に取り組む <ul style="list-style-type: none"> <input type="checkbox"/> 問題のリストアップ <input type="checkbox"/> BingAIを使用して問題を解決 <p>週6: プロジェクトの締めくくり</p> <ul style="list-style-type: none"> Day 36-38: 6週間の学習と実装を振り返り、進捗報告の準備 <ul style="list-style-type: none"> <input type="checkbox"/> 学習ポイントのリストアップ <input type="checkbox"/> 進捗報告のドキュメント作成 Day 39-42: インターンシッププロジェクトの成果物のまとめとプレゼンテーションの準備 <ul style="list-style-type: none"> <input type="checkbox"/> 成果物のリストアップと整理 <input type="checkbox"/> プレゼンテーションのスライド作成 |
|--|---|

図 10. ChatGPT(GPT-4.0) が作成した実習計画書

筆者はこの実習計画表に従い、インターンシップ活動を行った。

5.2 生成 AI を使った Python コードの生成

この項目については、生成 AI で Python コードを生成したりエラーコードの修正を試みたりする実験で活用した。その活用事例に関して、以降「6. 生成 AI と量子回路の生成」の中で詳しく説明していく。

6. 生成 AI と量子回路の生成

6.1 量子回路生成の実験の概要

ここでは、生成 AI を活用して、Python の外部ライブラリ Qiskit[15] を使った量子回路の生成の実験について記述する。この実験では「生成 AI を使って、量子回路の生成と数式の表示および、量子計算の質問のプロンプトに対する的確な回答を目指す」ことを目的としている。また、ChatGPT には「カスタム指示」といった機能があり、それを設定すると ChatGPT をパーソナライズ化することができる。これは ChatGPT のみがもっている機能だが、他の生成 AI サービスでも同様の指示を与えることで、カスタム指示を再現する。今回、この実験で使った生成 AI は以下である。

- ChatGPT[1]
- BingAI[3]
- Google Bard[4]

今回の実験の概要は以下のとおりである。また、今回の実験で使用した各生成 AI の種類と機能について、以下の表 1 に示す。

表 1. 今回の実験で使用した各生成 AI の種類と機能

| 生成AIの種類 | インターネット接続 | 使用料 | Pythonの実行 | Dataのアップロード |
|---|-----------|-----|-----------|-------------|
| ChatGPT(GPT-3.5) Default | × | なし | 不可 | 不可 |
| ChatGPT(GPT-3.5) Custom Instructions | × | なし | 不可 | 不可 |
| ChatGPT(GPT-4.0) With DALL-E, browsing and analysis | ○ | あり | 可 | 可 |
| ChatGPT(GPT-4.0) Plugins | ○ | あり | 不可 | 不可 |
| Bing AI(GPT-4.0) | ○ | なし | 不可 | △ 画像のみ |
| Google Bard(GPT-4.0) | ○ | なし | 不可 | △ 画像のみ |

【今回の実験の概要と成功とみなす条件】

- Python のバージョンは 3.10.12(Google Colab ※ 2023/11/13 時点)
- Qiskit のバージョンは 0.45.0(※ 2023/11/13 時点)
- ChatGPT でサーバ混雑による NetworkError が多発したため、一部のプロンプトでは実験プロンプトの下に「ただし、回答は 700 文字ごとに分割し、私が「続き」と入力したら続きの回答を投稿してください。」と記述する。
- プロンプトを送信し GPT から Python コードが返ってきたあと、次のプロンプトには原則エラー文のみを送信する。
- 生成 AI が“ ネットなどを参照して ...etc” と返答し、Python コードを記述しなくなった場合は、コードは動かなかったものとみなし、やり取りの回数を 10 回とする。
- 生成 AI が答えまで記述せず、中途半端な返答をしてきた場合は、その返答内にある文を使って次のプロンプトを決める。「例：「測定結果から得られた値を古典的なアルゴリズムで解析し、15 の素因数を求めます。...」のとき、「測定結果から得られた値を古典的なアルゴリズムで解析してください。...」」
- カスタム指示を使用しないのは「ChatGPT(GPT-3.5) -Default-」のみで、その他の生成 AI はすべてカスタム指示を使用する。
- 9 回目のプロンプトを送信し、その返答にあるコードでエラーが発生した場合、そのコードは動かなかったものとみなす。
- 生成 AI の回答が間違っている場合、次のプロンプトを「答えが違います。」とする。

また、量子回路生成の実験の最適化のためにカスタム指示を導入した。カスタム指示は以下のとおりである。

【カスタム指示】

ChatGPT にあなたについて何を知らせれば、より良い応答を提供できると思いますか？

>> 量子計算および量子コンピューティングのスペシャリスト

ChatGPT にどのように応答してほしいですか？

>> 回答は主に **qiskit** を使う

実行環境は、**Python3.10.12** で **qiskit** のバージョンは **0.45.0** である。

全ての数式、文字式を **LaTeX** 形式で表示する。

量子アルゴリズムや量子回路について解説を行う際は、必ず **Python** コード (量子回路は '**mpl**' 表示) とアルゴリズムについて表す数式 (必ず **LATEX** 形式) を使ってわかりやすく教える。

Python コードには、1 行ごとにコメントアウトを残し、その回路やコードが何を示しているか詳しく記述する。

量子回路を実行結果で示すとともに、測定結果を保存する "**measure**" を必ず記述する。

そして、今回実験で使用するプロンプトの一覧を以下に示す。

【今回指示するプロンプト一覧】

K_1 : 2 量子ビットの量子重ね合わせ状態について、量子回路を実装して詳しく説明してください。

K_2 : Shor のアルゴリズムについて 15 の素因数を求める量子回路を実装し、詳しく説明してください

K_3 : 2 量子ビットにおける量子フーリエ変換を実装し、詳しく説明してください。

K_4 : QAOA を使って量子回路を実装し、MaxCut 問題を解いてください

K_5 : 量子振幅推定を用いて量子数値積分を実装してください。求める解は $\sin^2(x)$ の 0 から 1 までの定積分の値です。

ここで注意点として、今回 ChatGPT(GPT-3.5) -Default- と ChatGPT(GPT-3.5) -Custom Instructions- について、比較実験のために赤字にしてある ChatGPT(GPT-3.5) -Default- では、カスタム指示を導入していない状態で実験を行う。

表 1 に示している ChatGPT(GPT-4.0) -Plugin- について、以下の 3 つを使用する。

【ChatGPT(GPT-4.0) -Plugin-】

- BrowserOp

→ウェブページの詳細情報を取得し、リアルタイムの検索エンジンの結果を取得するためのツール。

- Data Interpreter

→ SandBox 環境を使って Python コードのデータ解析を可能にするツール

- AskTheCode

→リポジトリの構造を分析し、関連するファイルの内容を取得して質問に答えるツール

さらに、今回の実験で“生成 AI のコード生成における 1 つのプロンプトに対してエラーを修正する能力”を表す指標であるエラー修正率 E_m と、“生成 AI のコード生成における実験するすべてのプロンプトのエラーを修正する能力の平均値”を表す指標である平均エラー修正率 E_{avg} の 2 つを導入した。これらの E_m と E_{avg} について、以下のルールを設け、数式とその例についても示す。

【指標のルール】

- エラーに対してどれくらい修正できるかを表す指標 (0 に近いほど修正力がある)
- 最初のプロンプトに対する回答でコードが動いた場合、 $E_m = 0$ とする
- 9 回目のやり取りの後の回答でコードが動かなければ、そのコードは動かないものとして、 $E_m = 1$ とする

【指標の数式】

“生成 AI のコード生成における 1 つのプロンプトに対してエラーを修正する能力”を表す指標であるエラー修正率 E_m を以下の数式で表す。

～ m 個目のプロンプトにおけるエラー修正率 E_m ～

$$E_m = \sin\left(\frac{\pi}{2} \cdot \frac{K_m - 1}{r_{max} - 1}\right)$$

生成 AI のコード生成における実験するすべてのプロンプトのエラーを修正する能力の平均値を以下の数式で表す。

～平均エラー修正率 E_{avg} ～

$$E_{avg} = \frac{E_1 + E_2 + \dots + E_{m=n}}{n}$$

n : 総プロンプト数, r_{max} 生成 AI とのやり取りの総回数 (ただし, $1 < r_{max}$)

K_m : m 個目のプロンプトに関して何回目のやり取りでコードが動いたか ($1 \leq K_m \leq r_{max}$)

【指標の使用例】

例) $n = 5, r_{max} = 5, E_1 = 0, E_2 = 0.38, E_3 = 1, E_4 = 1, E_5 = 0.92$ のとき

$$E_{avg} = \frac{0 + 0.38 + 1 + 1 + 0.92}{5}$$

$$E_{avg} = 0.66$$

6.2 実験結果

今回の実験において、検証を行った $K_1 \sim K_5$ と E_{avg} を以下の表 2 に示す。

表 2. 今回の実験に使用した各生成 AI の結果

| 生成AIの種類 | インター ネット接続 | 使用料 | Pythonの 実行 | Dataのアップ ロード | →実験値 | K_1 | K_2 | K_3 | K_4 | K_5 | E_{avg} |
|---|---------------|-----|---------------|-----------------|------|-------|-------|-------|-------|-------|-----------|
| ChatGPT(GPT-3.5) Default | × | なし | 不可 | 不可 | | 10 | 10 | 2 | 10 | 10 | 0.83 |
| ChatGPT(GPT-3.5) Custom Instructions | × | なし | 不可 | 不可 | | 1 | 10 | 10 | 10 | 10 | 0.80 |
| ChatGPT(GPT-4.0) With DALL-E, browsing and analysis | ○ | あり | 可 | 可 | | 2 | 2 | 2 | 10 | 10 | 0.504 |
| ChatGPT(GPT-4.0) Plugins | ○ | あり | 不可 | 不可 | | 1 | 3 | 2 | 10 | 10 | 0.503 |
| Bing AI(GPT-4.0) | ○ | なし | 不可 | △ 画像のみ | | 1 | 10 | 1 | 10 | 10 | 0.60 |
| Google Bard(GPT-4.0) | ○ | なし | 不可 | △ 画像のみ | | 1 | 10 | 1 | 10 | 10 | 0.60 |

この実験結果から「ChatGPT(GPT-4.0) -Plugins-」の E_{avg} が最も小さいことから最も性能が高く、「ChatGPT(GPT-3.5) -Default-」の E_{avg} が最も大きいことから最も性能が低いといった結果になった。今回の実験のポイントとして、ChatGPT のうち GPT-4.0 を搭載している生成 AI のサービスのみが、唯一 K_2 のプロンプトの解を導くことができた点についてだ。

ゆえに総合的に判断すると、インターネット接続が可能で、画像のみならず pdf や pptx などの拡張子のファイルの解析が可能で、「ChatGPT(GPT-4.0) -Plugins-」とほぼ同水準の

性能である「ChatGPT(GPT-4.0) -With DALL-E,browsing and analysis-」が量子計算の学習において最も良いと判断する。

その一方、 K_4 と K_5 のプロンプトに対して実験に成功した生成 AI サービスはなかったため、それを改善するためにカスタム指示の内容を変えたり生成 AI に量子アルゴリズムの手法についてファインチューニングを行ったりする必要があると考える。さらに、すべての生成 AI サービスに見られた特徴として、現在の Qiskit の新しいバージョンが刷新される以前のデータのモデルであるため、現在の Qiskit のバージョンでは使えないクラスや関数が存在した。

6.3 生成 AI を活用した新しい学習法

今回の実験から、生成 AI を活用して量子計算の学習に活かすことは、今回実験で利用したプロンプトにおいて有効であるといった結果に至った。また、今回の実験で利用した各々の生成 AI の性能を、定量的な値で比較することができた。その一方、実験の結果を見ると、平均エラー修正率の数値が最も良くても 0.503 であることや、利用したプロンプト数が 5 つであるため、あらゆるプロンプトに対する検証が行われていないという問題点がある。そのため現時点では、量子計算に関するすべての質問に対して、正しい答えを出力する保証がないため、量子計算を知らない初学者が生成 AI のみを活用して量子計算を学習することは難しい。そこで生成 AI のみでの学習では不十分なことから、今回筆者が量子計算を学習するにあたり「IBM Quantum で学ぶ量子コンピュータ [12]」を採用した。この書籍の対象として、Python プログラミングを通じて、量子コンピュータのアルゴリズムやアプリケーションを学びたいエンジニアに向けた内容となっているため、この対象に合致していると量子計算の学習に大いに役立てることができると考える。

ただ、今回の実験のように、生成 AI に対してあるアルゴリズムの Python プログラミングの生成とそのコードの修正には一定の成果がみられたため、量子計算の初学者が [12] の書籍に関して十分理解することができれば、生成 AI を活用して Python を活用した量子計算の学習を効率的に進められることができると考える。

7. 考察と結論

本稿では、量子コンピューティングにおける生成 AI の活用方法について説明した。今回のインターンシップにおいて一定の成果を得ることができたが、その一方で課題も見えた。量子計算に触れたことがない初学者に関して、生成 AI のみを使って学習することは現時点では難しいという結果に至った。量子計算に触れたことがない初学者が学習を進めていくためには、生成 AI だけでなく量子計算に関する論文や記事、動画、[12] の書籍を参考にすることが不可欠であると考え。今回のインターンシップでの生成 AI の活用を含めた量子計算の学習の手順、まとめについて以下に示す。

1. 生成 AI による実習計画表の作成

| 進捗チェックシート | 週4: ChatGPT (GPT-3.5) の活用 |
|--|--|
| <p>週1: 量子コンピューティング入門</p> <ul style="list-style-type: none"> Day 1-3: 量子コンピューティングの基本概念を学ぶ <ul style="list-style-type: none"> 基本的な量子力学の概念を理解 量子計算モデルの基礎を把握 Day 4-7: 量子ゲートと量子ビットの操作に関する理解を深める <ul style="list-style-type: none"> 主要な量子ゲートを学ぶ 量子回路の構築と実装 <p>週2: QAOAの理論</p> <ul style="list-style-type: none"> Day 8-10: QAOAの理論を学ぶ <ul style="list-style-type: none"> QAOAの基本的な仕組みの理解 QAOAの利点と制限を把握 Day 11-14: QAOAアルゴリズムの詳細な理解と数学的背景の探求 <ul style="list-style-type: none"> アルゴリズムのステップごとの動作の理解 重要な数学的な式や定理の理解 <p>週3: QAOAの実装</p> <ul style="list-style-type: none"> Day 15-17: QiskitやCirqを使用してQAOAの基本的な実装方法を学ぶ <ul style="list-style-type: none"> QiskitやCirqの基本的な操作を学ぶ QAOAのシンプルな実装を試す Day 18-21: サンプル問題を使用してQAOAの実装と最適化に取り組む <ul style="list-style-type: none"> サンプル問題の選定と定義 QAOAの実装と結果の評価 | <ul style="list-style-type: none"> Day 22-24: ChatGPTの基本的な使用法を学び、自然言語処理の概念を理解する <ul style="list-style-type: none"> ChatGPTのインターフェースの使い方を学ぶ 自然言語処理の基本を理解 Day 25-28: ChatGPTを活用して量子コンピューティングに関する質問や課題を解決する <ul style="list-style-type: none"> 質問のリストアップ ChatGPTを使用して質問を解決 <p>週5: BingAI (GPT-4.0) の活用と量子最適化への応用</p> <ul style="list-style-type: none"> Day 29-31: BingAIの高度な機能と自然言語生成の学習 <ul style="list-style-type: none"> BingAIの主な特長を学ぶ 自然言語生成の応用例を試す Day 32-35: BingAIを使用してQAOAや量子最適化に関する複雑な問題に取り組む <ul style="list-style-type: none"> 問題のリストアップ BingAIを使用して問題を解決 <p>週6: プロジェクトの締めくくり</p> <ul style="list-style-type: none"> Day 36-38: 6週間の学習と実装を振り返り、進捗報告の準備 <ul style="list-style-type: none"> 学習ポイントのリストアップ 進捗報告のドキュメント作成 Day 39-42: インターンシッププロジェクトの成果物のまとめとプレゼンテーションの準備 <ul style="list-style-type: none"> 成果物のリストアップと整理 プレゼンテーションのスライド作成 |

図 11. ChatGPT(GPT-4.0) が作成した実習計画表

- [12] の書籍や記事を活用した量子コンピュータの概念および量子計算の学習
 - ・[12] では主に、「Chapter1 量子コンピュータとは?」「Chapter2 量子コンピュータの数理」「Chapter3 IBM Quantum を使った量子計算」を学んだ。
 - ・インターネット上の記事では、以下の記事を使って学習した。
 - 日本総研が公開している「量子コンピュータの概説と動向」[16] 中の、第一章
 - NTT データが公開している量子コンピューティングガイドライン [17] 中の、第一章および第二章
 - IBM が公開している「Qiskit を使った量子計算の学習」[15] 中の、「1. 量子状態と量子ビット」「2. 複数量子ビットともつれ状態」
- [12] の書籍や Youtube の動画、記事を参考にした量子アルゴリズムおよび QAOA の学習
 - ・[12] では主に、「Chapter4 Qiskit を使った汎用量子計算」「Chapter5 量子古典ハイブリッドアルゴリズム」
 - ・動画では主に以下のものを使って学習した。
 - 【誰でも量子コンピュータ！量子アルゴリズム編】Quantum Computing for You【第2回・9/16 実施】[19]
 - 【誰でも量子コンピュータ！QAOA 編】Quantum Computing for You【追加公演第4回・9/30 実施】[20]
 - ・インターネット上の記事では、以下の記事を使って学習した。
 - IBM が公開している「Qiskit を使った量子計算の学習」[15] 中の、「3. 量子プロトコルと量子アルゴリズム」「4. 量子アプリケーションのアルゴリズム」

もちろん上記の手順が全てではないが、量子コンピュータを初めて学ぶ際は、この手順に沿って、量子計算について学習を深めていくとよい。

今回は、量子計算について生成 AI を活用した研究を立ち上げた。今回行った実験より、生成 AI が量子計算の学習において、Python プログラミングが可能で量子計算を初めて学習する人の完全なサポートを行うことができる可能性があると考える。一部の限られた問題のみではあるが、生成 AI を活用することで、量子計算の分野にも応用できることを示した。

この活用法は他の領域にも活用できると考える。その例として、ある研究を立ち上げたいときに、その内容を詳しく記述して生成 AI に研究計画表を作成してもらう。次に、生成 AI を活用してその研究に必要な文献の調査をしてもらう。ここで注意すべきこととして、生成 AI は事実に基づかない情報を生成する現象があり、これはハルシネーション [21] と呼ばれている。これを判別するために、研究を行う際は生成 AI のみに頼るのではなく、生成 AI が出力する内容について事実かそうではないかについて判断ができ、事実でない場合はそれを自力で修正できるという前提のもと、アイデアを広げる方向で活用することを推奨する。具体的にアイデアを広げた例として、今回のインターンシップで以下の図 12 のプロンプトを実行した。



図 12. “QAOA で解ける問題の例” について ChatGPT(GPT-4.0) に質問したもの

この図 12 は、筆者が QAOA で解くことのできる問題の例として、生成 AI に列挙してもらったものである。このように生成 AI の利点として、プロンプトに対して簡潔な回答かつさまざまなアイデアを提供ができるという点がある。ただ、課題としてハルシネーション [21] の現象が起こる可能性があるが、生成 AI の回答をうのみにせず、新しいアイデ

アを生み出すことに活用することは、研究において活用可能であると考え。

この研究を通して、生成 AI の新しい活用法として、学習の計画表を作成したり文字のプロンプトからプログラミングコードを生成したり、エラーコードの修正を支援したりすることができると思った。しかしながら、今回の実験では十分な精度が出せていないため、さらにプロンプト数を増やしたり指示を増やしたりするなどの検証が必要である。そして、生成 AI がプロンプトに対して正しい回答と正しい構文で Python コードを出力することが、これからの課題となってくる。この生成 AI の活用は、未知数である量子計算の分野のみならず、他の未知、既知の学問でも学習の支援に役立たせることができると考えている。ゆえに、教育領域においてこのような生成 AI の活用を取り入れることで、さらなる教育環境の向上を期待している。

8. 謝辞

本稿を執筆するにあたって、インターンシップとしての活動にご協力いただいた株式オレンジテクラボの、西田健次 (CTO) さん、山下一郎 (技術顧問) さん、山田敏哉 (シニア UX デザイナー) さんに感謝いたします。ありがとうございました。

参考文献

- [1] OpenAI, ChatGPT, <https://openai.com/chatgpt> (accessed on 26 December 2023)
- [2] 文部科学省、初等中等教育段階における生成 AI の利用に関する暫定的なガイドライン、https://www.mext.go.jp/content/20230710-mxt_shuukyo02-000030823_003.pdf (accessed on 26 December 2023)
- [3] Microsoft, Bing, <https://www.bing.com/search?form=NTPCHB&q=Bing+AI&showconv=1> (accessed on 26 December 2023)
- [4] Google, Google Bard, <https://bard.google.com/chat> (accessed on 26 December 2023)
- [5] Zeeya M., "First sale for quantum computing", nature, Vol. 474, p. 18, (2011)
- [6] Steane, A., "Quantum computing." Reports on Progress in Physics 61.2 (1998): 117.
- [7] 株式会社 QunaSys, 量子コンピュータの基礎から応用まで, <https://speakerdeck.com/qunasys/quantum-summit-2019?slide=102> (accessed on 29 December 2023)
- [8] Ladd, T., Jelezko, F., Laflamme, R. et al. Quantum computers. Nature, Vol. 464, p.45–53 (2010).
- [9] Quantum Native Dojo, 5-3. Quantum Approximate Optimazation Algorithm (QAOA): 量子近似最適化アルゴリズム, https://dojo.qulacs.org/ja/latest/notebooks/5.3_quantum_approximate_optimizazation_algorithm.html (accessed on 8 January 2024)
- [10] Quantum Algorithm Zoo, Navigation, <https://quantumalgorithmzoo.org/> (accessed on 8 January 2024)
- [11] Farhi, E., "Quantum supremacy through the quantum approximate optimization algorithm." Bulletin of the American Physical Society 62 (2017).
- [12] 湊雄一郎, 加藤拓己, 比嘉恵一朗, 永井隆太郎『IBM Quantum で学ぶ量子コンピュータ』、秀和システム、2021 年 3 月 10 日
- [13] Investor, QAOA アルゴリズムの量子回路の仕組みを理解する, <https://www.investor-daiki.com/qaoa-circuit> (accessed on 8 January 2024)

- [14] Majumdar, Ritajit, et al. "Optimizing Ansatz Design in QAOA for Max-cut." arXiv preprint at arXiv: 2106.02812, <https://arxiv.org/pdf/2106.02812.pdf>. (accessed on 8 January 2024)
- [15] Qiskit, Qiskit community, <https://qiskit.org/> (accessed on 8 January 2024)
- [16] 日本総研、量子コンピュータの概説と動向、
<https://www.jri.co.jp/MediaLibrary/file/column/opinion/pdf/11942.pdf> (accessed on 8 January 2024)
- [17] NTT データ 量子コンピューティングガイドライン、NTT DATA、
https://www.nttdata.com/jp/ja/-/media/nttdatajapan/files/news/services_info/2021/012800/012800-01.pdf (accessed on 8 January 2024)
- [18] Qiskit, 量子とは、
<https://qiskit.org/textbook/ja/what-is-quantum.html> (accessed on 8 January 2024)
- [19] 大関真之, (2022, September 16), 【誰でも量子コンピュータ！量子アルゴリズム編】 Quantum Computing for You 【第 2 回・9/16 実施】 [video], Youtube,
https://www.youtube.com/watch?v=ZSqYDf_vSAo&t=0s (accessed on 8 January 2024)
- [20] 大関真之, (2022, September 30), 【誰でも量子コンピュータ！QAOA 編】 Quantum Computing for You 【追加公演第 4 回・9/30 実施】 [video], Youtube,
<https://www.youtube.com/watch?v=7h2mticpcFc&t=0s> (accessed on 8 January 2024)
- [21] Vipula Rawte, Amit Sheth, and Amitava Das. "A survey of hallucination in large foundation models." arXiv preprint arXiv:2309.05922 (2023).

| | | | | |
|------|-------------|------------------|-------|-----|
| 廣峯優希 | 東京国際工科専門職大学 | 工科学部 | 情報工学科 | 3 年 |
| 富山哲男 | 東京国際工科専門職大学 | 工科学部 | 情報工学科 | 教授 |
| 宮崎 淳 | 東京国際工科専門職大学 | 客員教授、株式会社オレンジテラボ | | |