

前提知識を考慮した数学の確率問題自動解答の精度向上

谷村華梨 平博順
大阪工業大学 情報科学部
{e1c20061,hirotoshi.taira}@oit.ac.jp

概要

大規模言語モデルによる文章生成の品質が大幅に向上し、さまざまな分野への応用が期待されている。しかし、数学などの教科における個別学習での使用には、各教科の問題に対する解答精度が低いなどの課題がある。本研究では、大規模言語モデルの一つである GPT-4 を用いて高校数学における確率分野の問題に対して自動解答を行い、不正解となる場合の原因の分析を行った。分析の結果、GPT-4 に不足していると考えられる分野の前提知識を追加して解答生成を行い、解答精度が向上するケースがあることが分かった。題材として、従来研究で高い精度を出すことが難しいとされている確率分野の問題を取り扱うこととした。その結果、前提知識を考慮することで、解答精度の向上が見られる事例があった。

1 はじめに

2022 年 11 月に公開された対話型文章生成 AI である ChatGPT は、その高品質な回答により世間からも大きく注目されている。一般的な事柄に対する質問への回答精度が高く、教育分野への活用も多く提案されている。文章生成 AI が個別学習に使用されるようになれば、各人の理解不足の部分に絞った学習も行いやすくなり、効率的に学習が進められることも期待できる。しかし、現時点での ChatGPT などの文章生成 AI は、計算や数学問題を解くのが苦手であると言われており、数学の分野に関してはそのまま個別学習に文章生成 AI を使うのにはまだ課題がある。一方、ChatGPT 以前の数学の問題に対する自動解答の研究では、最終的な計算結果のみで解答精度を出しているものや計算問題の解答精度についての研究が多く、数学の個別学習に重要な問題の解法過程の正否まで細かく評価している研究や文章問題についての研究は少ない。数学の個別学習にこれらの技術を用いるには、ChatGPT などの対話型文章生

成 AI のような高い文章生成能力も重要な要素となる。高い文章生成能力を保ちつつ、数学の問題解答精度についても高める技術が求められている。そこで本研究では、ChatGPT の基盤となっている大規模言語モデル GPT-4 の数学問題を自動解答する際にどのような部分が解答精度の低下につながっているのかを分析した。現在の大規模言語モデル (LLM) に不足した部分を補うことにより数学問題の自動解答の精度向上を目指した。題材としては、数学の問題の中でも問題内容を文章で表現できることが多く、また従来の数学問題自動解答の研究でも高い精度を出すことが難しいと言われている確率分野の問題を取り上げる。

2 関連研究

Dries ら [1] は、自然言語で表現された数学の確率問題を、Problog 言語による中間表現に変換した上で、2 段階で問題を解いている。ここで、Problog 言語とは、命題に確率が付与できるように Prolog 言語が拡張された論理型言語である。また、Simon ら [2] は、自然言語で書かれた問題を Problog 言語の表現に高精度で変換するためにニューラルネットを用いている。Zheng ら [3] は、GPT-3.5 を用いて、前の解答をヒントにしながら対話を繰り返すことで、正解率を上げる手法を提案している。Aman ら [4] は、LLM に数学の問題を解く Python コードを生成させ、それを実行することで、計算誤りを減らす手法を提案している。Zhao ら [5] は、Aman らの手法と LLM による解答のどちらが良い答えかを推論することで、高精度な問題解答を行っている。GPT-4 は、数学の問題を解くための学習は特に行われていない LLM のため、計算間違いを起こしやすいといわれている。この欠点に対処するため、岡谷ら [6] は、筆算と同様の計算手順で計算を行わせるようなプロンプトにより、計算を行わせたが、精度向上に寄与しなかったことを報告している。本研究では、数学の確率問題に対して問題を解く上で、LLM に解くの

表 1 実験結果 (%)

	完答	解答のみ正解	過程のみ正解	両方不正解
「チャート式」	59.65	1.75	5.26	33.33
「チャート式」(英訳)	52.63	0.00	5.26	42.11
nlp4plp	83.92	0.00	10.05	6.03
nlp4plp (和訳)	75.88	0.00	11.56	12.56

表 2 「チャート式」の各テーマ別の解答精度 (%)

問題テーマ	完答	解答のみ正解	過程のみ正解	両方不正解
集合の個数	80.00	0.00	6.67	13.33
場合の数	77.78	0.00	0.00	22.22
順列	57.14	14.29	0.00	28.57
円順列・じゅず順列	50.00	0.00	0.00	50.00
重複順列	28.57	0.00	0.00	71.43
組合せ	53.33	0.00	6.67	40.00
確率と種々の定理	55.56	0.00	0.00	44.44
独立試行	100.00	0.00	0.00	0.00
反復試行	50.00	0.00	10.00	40.00
確率の乗法定理	50.00	0.00	25.00	25.00

に必要な前提知識が欠けていることが問題解答精度の低下を招いているのではないかと仮定して、精度向上方法について分析・検討を行った。

3 GPT-4 による確率問題の自動解答

まず、高品質な文章生成が可能な LLM として高い評価を得ている GPT-4 を用いて、数学の確率問題を解き、その解答精度について評価を行った。

3.1 評価項目

GPT-4 を用いた解答に対して、以下の 2 項目で人手で評価した。

- 最終的な計算結果の正しさ
- 解法の過程の正しさ

なお、計算誤りのみの理由で最終的な計算結果が不正解であった場合は、解法の過程は正しいと評価し、その他の理由で最終的な結果が不正解であったときは、解法の過程が誤っているとして評価した。また、最終的な計算結果が正しいが、偶然正解したと考えられる場合は、解法の過程が間違っていると評価した。

3.2 評価用データ

本研究では、以下の 2 種類のデータセットに対して評価を行った。

- 「新課程 チャート式 解法と演習 数学 I+A」 [7]
- nlp4plp データセット [1]

それぞれの数学の確率問題のデータセットは、「チャート式」が日本語の問題セット、nlp4plp が英語の問題セットである。「チャート式」は場合の数と確率の例題 114 問、nlp4plp は確率問題 199 問から構成されている。GPT-4 で自動解答させる際、問題の言語により解答精度に違いがあるかを検証するために、それぞれのデータを他の言語（英語または日本語）に DeepL を用いた翻訳したものについても評価することとした。

3.3 結果

「チャート式」と nlp4plp および、それらを翻訳したものについて、GPT-4 により自動解答を行った結果を表 1 に示す。

実験の結果、「チャート式」の日本語の問題では、最終的な計算結果と解法の過程のどちらもあっているものが、59.65 % あった。また、「チャート式」の問題を英語に翻訳し GPT-4 に解かせたものでは、52.53 % と、オリジナルの日本語の問題に対する解答精度の方が高かった。また、nlp4plp でも同様に、英語のデータセットでは最終的な計算結果と解法の過程のどちらもあっているものが 83.92 % であったのに対し、日本語に翻訳したものでは 75.88 % と、

翻訳前のオリジナルなデータに対する解答精度の方が高かった。

「チャート式」と nlp4plp の解答精度を比較すると、nlp4plp の方が、最終的な計算結果が間違っており、解法過程が合っている問題の割合が高く、nlp4plp の方が計算誤りが多い。

「チャート式」の各テーマ別の解答精度を表 2 に示す。実験の結果、問題テーマにより解答精度に大きな違いが見られた。特に、円順列・じゅず順列、重複順列のテーマでは、最終的な計算結果と計算過程のどちらも不正解である問題の割合が 50%以上と高かった。

3.4 考察

どちらのデータセットも、翻訳したものとしていないものを比較すると、翻訳していないオリジナルのデータセットの方が、解答精度が高かった。その理由として、翻訳された文が完全には元の文の内容を表したものになっておらず、情報の欠如が起こったためだと考えられる。GPT-4 は、日本語より英語の入力に対する解答精度が高いといわれているが、この数学の確率問題データセットに関しては英語に翻訳せず、オリジナルの日本語では GPT-4 の入力として与えた方がよいことが分かった。

解法の過程はあっているが最終的な計算結果が間違っている問題は、「チャート式」の問題では 5.56 %、nlp4plp の問題では 10.05 %であった。最終的な解答が誤った原因は、途中の計算誤りによるものがほとんどであった。特に、桁数が多い数の割り算での誤りが目立った。nlp4plp の方が「チャート式」の問題より計算過程で桁数が大きくなる問題が多くあったため、結果として、nlp4plp の方が最終的な解答誤りが多くなったと考えられる。また、計算途中で分数を小数に変換してしまう解答も多く、丸め誤差により最終的な解答が誤っているものも多かった。そのため、途中の計算は分数のままで計算させる方が計算誤りは防げるのではないかと考えられる。

表 2 に示した通り、「チャート式」の問題では、円順列・じゅず順列、重複順列の問題で特に解答精度が低かった。GPT-4 で事前学習されているデータは、英語のテキストが多いと考えられるが、英語で書かれた円順列やじゅず順列、重複順列の問題が少ない可能性がある。そのため、これらのテーマの問題を解かせる際に、これらのテーマに関する前提知

識を与えることで、精度が向上する可能性がある。

最終的な計算結果と解法の過程のどちらも不正解である問題は、「チャート式」の問題では 35.19 %もあった。これらの問題は、問題設定の認識が不正確で、考慮すべき条件の欠落や、問題に記載された数量に関する認識誤りなどによるものである。問題設定に関する認識誤りは、各テーマの問題を解く際に必要な前提知識を追加することで精度向上が図れる可能性がある。

4 前提知識の考慮

前節で考察したように、GPT-4 による確率問題の自動解答では、計算誤りや問題設定に対する認識誤りなど、様々な原因により解答を誤ってしまう。従来研究にもあったように、GPT-4 による計算誤りの精度を上げる方法は、現在のところ明らかになっていない。しかし、前提知識の不足については、プロンプトで知識を追加して解答精度を上げられる可能性があるため、解答精度向上に有効な前提知識について検討を行った。前提知識を考慮することで、正解となった問題を以下に示す。

4.1 考え方の考慮

問題の考え方を示すことで正解となったものを以下に示す。

<問題>

白色カードが 5 枚、赤色カードが 2 枚、黒色カードが 1 枚ある。同じ色のカードは区別できないものとして、この 8 枚のカードを左から 1 列に並べるとき、次のような並べ方は、それぞれ何通りあるか。赤色カードが隣り合う

< GPT-4 による解答 >

2 枚の赤色のカードを 1 つのグループと見なし、白色のカード 5 枚、赤色のカードのグループ 1 つ、黒色のカード 1 枚の 7 つの要素を並べる。 $7 \times 6 \times 5 \times 4 \times 3 \times 2 \times 1 = 5040$ 通り。

<前提知識を追加したときの解答>

赤色カードが隣り合う場合、赤色カードを 1 つのグループと考え、白色カード 5 枚、黒色カード 1 枚、赤色カードのグループ 1 つの 7 つの要素を並べる。並べ方は 7 要素の順列で、同じものがある順列の公式を用いると、 $7! / (5! \times 1! \times 1!) = 42$ 通り。

<追加した前提知識>

同じ色のカードは区別できないという部分を理解できていなかったため、次の前提知識を追加した。

「ただし、区別ができないため、例えば (1,2), (2,1) は同じ場合と考える。」

区別できないものについてどのように考えればよいか前提知識を与えた結果、正解に導くことができた。確率問題ではさいころの大小や球の色など、区別するときとしないときで解法が異なり、特に、GPT-4 は区別しない問題を区別するものとして解くことが多かった。そこで前提知識として、区別ができない場合の具体例を示すことで解決することができた。このことから、問題の考え方は、具体例を示すことで解けるようになると思う。

4.2 解き方の考慮

問題の解き方を考慮することで正解となったものを以下に示す。

<問題>

6 人の生徒 A, B, C, D, E, F が丸いテーブルに座る。このとき、A, B が隣り合う並び方は何通りあるか。

<GPT-4 による解答>

A と B を 1 つのグループと見るとこのグループと C, D, E, F の 5 つの要素を丸いテーブルに座らせるので、 $5! = 120$ 通り。グループ内の A と B の座る位置は 2 通り。A と B が隣り合う並び方は、 $120 \times 2 = 240$ 通り。

<前提知識を追加したときの解答>

5 つの要素 (グループ AB と C, D, E, F) を丸いテーブルに座らせる場合の数は $(5-1)! = 4! = 24$ 。グループ AB 内で A と B が座る場合の数は $2!$ 。A, B が隣り合う並び方は $24 \times 2 = 48$ 通り。

<追加した前提知識>

GPT-4 による解答では、円ではなく 1 列であるときの解き方をしていたので、次の前提知識を追加した。「なお、異なる n 個の円順列の総数は $(n-1)!$ 」

円に並んでいるにもかかわらず、1 列に並ぶときの公式を用いて解いていた。そのため、円順列の公式を前提知識として追加することで正解に導くことができた。このことから、GPT-4 には円順列についての前提知識が少なかったため、解き方として公式を与えることで解くことができたと思う。

5 その他の工夫

前提知識の考慮以外で、プロンプトの追加により正解となった問題を以下に示す。

5.1 言葉の曖昧性の考慮

問題文の曖昧性の解消により、正解となったものを以下に示す。

<問題>

50 人の生徒の中でバス通学の者は 28 人、自転車通学の者は 25 人、どちらでもない者は 10 人である。バスと自転車の両方で通学している者の人数を答えよ。

<GPT-4 による解答>

バス通学の者と自転車通学の者は $28+25=53$ 人。生徒の総数は 50 人なのでバスと自転車の両方で通学している者は $53-50=3$ 人。

<提案手法による解答>

バスや自転車を使って通学している者は $50-10=40$ 人。バスと自転車の両方で通学している者は $28+25-40=13$ 人。

<追加した前提知識>

どちらでもない者を考慮できていなかったため、「どちらでもない者」を「バスも自転車も使わずに通学している者」という具体的な言葉で説明した。

言葉の曖昧性の考慮は、特に日本語の問題で有効だと考える。理由として、日本語には英語よりも曖昧な言葉が含まれていることが多いからである。そのため、曖昧な言葉は具体的に説明することで解けるようになると思う。

6 おわりに

本研究では、大規模言語モデル GPT-4 の数学問題の自動解答における解答精度の低下の原因を明らかにし、不足した部分を補うことで数学問題の自動解答の精度向上を目指した。その結果、各問題に必要な前提知識を付与することは、解答精度の向上につながるということが分かった。しかし、追加すべき前提知識が問題によって異なることから、今後の課題として、個別学習に利用するには自動的に問題文から追加すべき前提知識を把握し、考慮する必要があることだといえる。

参考文献

- [1] Anton Dries, Angelika Kimmig, Jesse Davis, Vaishak Belle, Luc De Raedt. Solving probability problems in natural language. In Proceedings of the Twenty-Sixth International Joint Conference on Artificial Intelligence, pp. 3981-3987, 2017.
- [2] Simon Suster, Pieter Fizev, Pietro Totis, Angelika Kimmig, Jesse Davis, Luc De Raedt, Walter Daelemans. Mapping

- probability word problems to executable representations. In Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing, pp. 3627-3640, 2021.
- [3] Chuanyang Zheng, Zhengying Liu, Enze Xie, Zhenguo Li, Yu Li. Progressive-hint prompting improves reasoning in large language models. arXiv preprint arXiv:2304.09797, 2023.
 - [4] Luyu Gao, Aman Madaan, Shuyan Zhou, Uri Alon, Pengfei Liu, Yiming Yang, Jamie Callan, Graham Neubig. Pal: Program-aided language models. In International Conference on Machine Learning, pp. 10764-10799, 2023.
 - [5] Xu Zhao, Yuxi Xie, Kenji Kawaguchi, Junxian He, Qizhe Xie. Automatic Model Selection with Large Language Models for Reasoning. arXiv preprint arXiv:2305.14333, 2023.
 - [6] 岡谷基弘. GPT-4 による足し算実験から示唆される Large Language Models の課題. 人工知能学会, 汎用人工知能研究会, SIG-AGI-024-02, 2023.
 - [7] 星野 泰也. 新課程 チャート式 解法と演習 数学 I+A. 第 23 刷, 数研出版, 2015.