

様式（2a）学修総まとめ科目 成果の要旨（※3頁以内）

学校名	木更津工業高等専門学校	専攻名	制御・情報システム工学専攻
専攻分野名称	工学	専攻の区分	情報工学
氏 名	長谷川駿一	学籍番号	23-B06
テーマ名	LLMによるプログラミング初学者のスキル分析と問題作成による学習支援		
指導教員名	大枝真一	指導補助教員名	

（1） 「学修総まとめ科目」 で取り組んだ学修・探究の内容

（1-1）学修・探究の背景と目的
<p>プログラミング初学者の能力判定に関する先行研究として、千枝ら[1]や飯棲ら[2]は、決定木やIRMといった教師なし学習手法をソースコードに適用し、能力判定に重要となる特徴量の抽出を試みている。これらの研究は、定期試験や課題提出によらない定量的な学習者理解度の把握可能性を示唆している。しかしながら、これらの機械学習モデルから得られる特徴量のみでは、学習者の問題解決アプローチなど、より包括的な視点からのソースコード評価が困難であるという課題が存在する。</p> <p>この課題に対し、本研究では従来の機械学習モデルをLarge Language Model（LLM）に置き換えることで、より包括的な学習者評価の実現を目指している。近年、LLMは目覚ましい発展を遂げており、適切なプロンプト設計によってその分析能力を最大限に引き出せることが様々な研究で実証されている[3]。本研究では、プロンプトの最適化を通じて、プログラミング初学者のスキル分析の効率化を図るとともに、分析結果に基づいて学習者の能力に適合した問題を自動生成するシステムの構築を目指している。</p> <p>本研究を通じて、プログラミング教育におけるLLMの有効性を実証的に示すことを目指している。これにより、従来の機械学習アプローチでは困難であった学習者の問題解決プロセスの質的評価と、それに基づく個別最適化された学習支援の実現という教育工学上の重要課題に対する新たな解決アプローチを提示することが期待される。</p>
（1-2）学修・探究の手法・方法
<p>本研究では、LLMによるプログラミング初学者のスキル分析と個別最適化問題生成を実現するため、複数のプロンプトエンジニアリング手法を組み合わせたアプローチを採用した。具体的には、Role Prompting[4]を用いてLLMに教育者としての役割を付与し、学習者評価の専門家としての文脈を確立した。また、Chain-of-Thought (CoT) [5]およびFew-Shot Learning[6]を導入し、分析プロセスの明示化と問題作成フォーマットの具体的指定を行った。さらに、Self-consistency[7]の概念に基づき、LLMに複数の問題候補を生成させ、その中から最適な問題を選択するメカニズムを実装した。</p> <p>実験では、異なる特性を持つ3種のLLMを比較評価した。第一のモデルは、OpenAIが開発したGPT-4o miniであり、比較的軽量の構成ながら、高速な推論処理と多様なタスクへの適応性を特徴としている。第二のモデルは、同じくOpenAIが開発したGPT-4oであり、高度な推論能力とマルチモーダル処理機能を備え、GPT-4o miniと比較してより精緻な自然言語処理が可能である。第三のモデルは、Googleが開発したGemini 1.5 Flashであり、大規模なテキストおよびコードベースの文脈理解に優れている。</p> <p>評価データとして、令和6年度木更津工業高等専門学校情報工学科2年生（41名）の後期中間試験における解答ソースコードと試験問題を使用した。システムの有効性評価には、学生を対象とした定量的アンケート調査を実施し、生成された問題の構成妥当性、難易度の適切性など、計6項目について10段階評価を行う形式を採用した。これにより、LLMの分析精度と問題生成能力の両面について、実践的な評価を行った。</p>
倫理的配慮

<p>本研究では、被験者の能力（スキル）を測定するために試験や課題を課すことになる。そのため、木更津工業高等専門学校の研究倫理審査委員会による倫理審査の承認を受けた。また、「木更津工業高等専門学校におけるヒトを対象とする研究の倫理に関する規則」を遵守する。</p>
<p>(1-3) 得られた結果と考察、将来展望</p>
<p>記述統計では、Gemini 1.5 Flashが「(問題文の)理解しやすさ」(平均7.40)で最も高く評価され、問題文の明確さを示唆した。GPT-4o miniは「(問題を解く)楽しさ」(平均6.55)と「良問評価」(平均6.63)で若干優位であり、学習意欲を高める問題生成の可能性を示した。GPT-4oは各項目で中間的な評価であり、バランスの取れた生成傾向を示した。しかし、全モデルで、「(問題を解く)時間の適切さ」(平均4.34~4.56)が低く、設定時間15分が短い可能性が示唆された。また、ソースコード評価も全体的に低く、コード品質の改善が必要である。</p> <p>モデル間の比較（一元配置分散分析、TukeyのHSD検定）では、「理解しやすさ」においてGemini 1.5 FlashとGPT-4o mini間に有意差($p=0.0456$)が認められ、Gemini 1.5 Flashの優位性が統計的にも裏付けられた。他のアンケート指標では有意な差は見られなかった。</p> <p>相関分析では、特にGemini 1.5 Flashにおいて、「難易度」とコード品質の間に強い負の相関($r=-0.657\sim-0.759$)、「時間の適切さ」とコード品質の間に中程度の正の相関($r=0.444\sim0.614$)、「(問題を解いて得られる)新しい学び」とコード品質の間に中程度の負の相関($r=-0.417\sim-0.566$)が認められた。これは、難易度が高いほどコード品質が低下し、適切な時間設定はコード品質向上に繋がり、新しい概念学習時はコード完成度よりも学習に重点が置かれることを示唆している。他のモデルでは同様の顕著な相関は見られなかった。</p> <p>これらの結果から、LLMはプログラミング問題生成の可能性を示すが、教育的視点に基づく問題設計、特に難易度調整、時間配分、コード品質向上が重要な課題である。今後の研究では、これらの課題に取り組むことで、より効果的な学習支援システム開発を目指す。</p>
<p>履修計画書から変更があった場合の理由および履修計画書に対する評価・コメントへの対応</p>
<p>プロンプトエンジニアリングとLLMに関する急速な技術進展と、それらの教育分野への応用可能性の高まりを受け、研究方針を大きく変更した。当初計画していた従来型の機械学習手法によるモチベーション分析から、より包括的なスキル評価とそれに基づく個別最適化問題生成という方向性へと転換を図った。</p> <p>この変更により、単なるモチベーションとソースコードの相関分析から、より実践的な教育支援システムの構築という目標へと発展させることができた。特に、LLMの活用により、従来の特徴量抽出では捉えきれなかった学習者のプログラミングスキルをより多角的に分析することが可能となった。また、分析結果に基づく個別最適化問題の自動生成という新たな機能を追加することで、より直接的な教育支援ツールとしての価値を高めることができた。</p>
<p>参考文献</p>
<p>[1] 千枝睦実, 大枝真一, “プログラミング授業での決定木を用いたドロップアウト原因の可視化,” 2019 年情報科学技術フォーラム, 2019.</p> <p>[2] 飯棲俊介, 大枝真一, “IRM(Infinite Relational Model)と決定木を用いたプログラミング初学者の能力判定のための特徴量の抽出,” IEICE Conferences Archives. TheInstitute of Electronics, Information and Communication Engineers, 2019.</p> <p>[3] Schulhoff, Sander, et al. “The Prompt Report: A Systematic Survey of Prompting Techniques.” arXiv preprint arXiv:2406.06608 (2024).</p> <p>[4] Kong, Aobo, et al. “Better zero-shot reasoning with role-play prompting.” arXiv preprint arXiv:2308.07702 (2023).</p>

- [5] Suzgun, Mirac, et al. "Challenging big-bench tasks and whether chain-of-thought can solve them." *arXiv preprint arXiv:2210.09261* (2022).
- [6] Brown, Tom, et al. "Language models are few-shot learners." *Advances in neural information processing systems* 33 (2020): 1877-1901.
- [7] Wang, Xuezhi, et al. "Self-consistency improves chain of thought reasoning in language models." *arXiv preprint arXiv:2203.11171* (2022).

(2) 「学修総まとめ科目」の学修・探究を支える学修全体について

(2-1) テーマの学修・探究の基盤となる専門科目の学修

本研究のテーマであるLLMを活用したプログラミング教育支援システムの開発には、複数の専門分野の知識が不可欠である。特に、本科4年の「プログラミング言語I・II」および「プログラミング演習III」で学んだプログラミングの基礎知識と教育手法は、初学者のスキル分析において重要な基盤となった。また、本科5年の「データマイニング」で修得した機械学習の理論と実践は、データ分析システムの設計に直接的に活用された。

さらに、専攻科1年の「ヒューマンインターフェース」の知識は、学習者に対する効果的な問題提示方法およびアンケートの設計に活かされている。

これらの専門科目で得られた知識を統合することで、教育工学とLLMを融合した新しい学習支援システムの開発が可能となった。

(2-2) テーマの学修・探究に関係する関連科目の学修

本研究の遂行にあたり、本科4年の「統計学」で修得した知識は特に重要な役割を果たした。この科目では、データ分析の基礎理論やLLMの構造理解に役立ち、さらにアンケート調査の設計と分析手法に関する理解を深めた。この知識は、LLMによって生成された問題の有効性を評価するためのアンケート設計において直接的に活用された。

具体的には、10段階評価による定量的データの収集方法の設計や、得られたデータの統計的分析手法の選定において、「統計学」で学んだ手法を適用した。これにより、研究結果の客観的な評価と、LLMの問題作成能力に対する統計的な検証が可能となった。

(2-3) 専攻に係る科目以外の学修

専攻科1・2年の「技術英語I・II」で培った英語力は、LLMやプロンプトエンジニアリングに関する最新の海外論文や技術文書の理解に大きく貢献した。特に、GPT-4やGeminiなど、急速に発展するAI技術に関する英語文献の読解が、研究の方向性を定める上で重要な役割を果たした。これらの科目で得られた知見は、よりグローバルな視点から研究アプローチを検討することを可能にした。

(2-4) 自分自身の4年間の学修全体の省察

私は本科4年から専攻科2年にかけて、情報処理技術に関する資格取得に積極的に取り組んだ。具体的には、本科4年に基本情報技術者試験、本科5年に統計検定2級、専攻科1年に応用情報技術者試験、そして専攻科2年には統計検定準1級を取得した。これらの資格取得を通じて、情報技術と統計学の両面から実践的な知識を深めることができた。

特に、本科5年の卒業研究では、Knowledge Tracingを活用した教育分野における学習者のスキル推定に取り組み、その研究成果を第85回情報処理学会全国大会で発表する機会を得た。この経験は、自身の研究成果を学術的な場で発表し、専門家との議論を通じて研究の質を高める貴重な機会となった。

今後は、4年間の授業や研究活動で培った、機械学習やLLMにおける学習や推論、プロンプトエンジニアリングの知識と技術を活かし、AIエンジニアとして技術革新に貢献していきたいと考えている。