

利用上の注意事項:

ここに掲載した著作物の利用に関する注意 本著作物の著作権は情報処理学会に帰属します。本著作物は著作権者である情報処理学会の許可のもとに掲載するものです。ご利用に当たっては「著作権法」ならびに「情報処理学会倫理綱領」に従うことをお願いいたします。

Notice for the use of this material The copyright of this material is retained by the Information Processing Society of Japan (IP SJ). This material is published on this web site with the agreement of the author (s) and the IP SJ. Please be complied with Copyright Law of Japan and the Code of Ethics of the IP SJ if any users wish to reproduce, make derivative work, distribute or

プログラミング言語教育のための自然言語における 第2言語教育法の援用に関する調査研究

石倉 滉大[†]
早稲田大学[†]

雲居 玄道[†]
早稲田大学[†]

梅澤 克之[‡]
湘南工科大学[‡]

小林 学[†]
早稲田大学[†]

中野 美知子[†]
早稲田大学[†]

平澤 茂一[†]
早稲田大学[†]

1. 研究背景・目的

近年、プログラミング言語教育は小学校での導入も始まり、重要性が高まってきている。そこで、次世代のプログラミング言語教育システムの開発にあたり、プログラミング言語教育に自然言語教育の手法を援用可能であると仮説を立てた。これにより、自然言語教育の豊富な知見をプログラミング言語教育に取り込むことで、より効果的な学習を提供可能となる。しかし、プログラミング言語は何かしらの自然言語を元に解説されることから、母語としてプログラミング教育は存在しえない。そこで、母語を教育済みの状態に対して、他の言語を教育する第2言語教育法に着目した。この第2言語教育法の研究は日本第二言語習得学会やEuroSLAなどの学会で盛んに行われており分野として確立されている。

そこで本研究では、第2言語教育とプログラミング言語教育の関連性やプログラミング教育に第2言語教育の知見を援用した研究についての調査を行い、それらの内容を踏まえた統合的な知見を考察する。

2. 調査内容

2.1 N. B. Pandza の研究

N. B. Pandza は、プログラミング言語と自然言語の類似点と相違点を整理した[1]。プログラミング言語と自然言語の類似点として、語彙、会話のルール、再帰性などを挙げている。語彙項目とは、if などのプログラミング言語の語彙は自然言語と同義の意味を持つことが多いことを指す。会話のルールとは、1つの呼びかけ（入力）に対して1つの返答（出力）があることを指す。再帰性とは、自然言語における”I have hurt myself.”のような主語と目的語が同一のものを指す文章を許容する特徴とプログラミングにおける関数の再帰構造が共通していることを示している。一方、相違点として単語の総数や会話方法を挙げている。自然言語は聴覚と視覚の両方でコミュニケーションをとり、何千もの語彙を使用しているが、プログラミング言語は数百の語彙のみを使用しており、視覚でのみ会話される特徴がある。

さらに学習過程の観点からは、プログラミング学習の際に、教育済みの言語の知識が学習に影響を与えることがよく見られることから類似性がある可能性が高いということが考えられる。

N. B. Pandza らは、プログラミング言語は自然言語と類似している部分は多くあるが、自然言語として捉えるのは難しいと結論付けた。しかし、未教育のプログラミング言語を教育済みプログラミング言語の知識を活かして学習することは比較的容易で今後の研究に役立つ可能性があるとしている。さらに、プログラミング言語と自然言語の学習過程には多くの点で共通しているため研究の余地があるとしている。

2.2 R. Rohmeyer らの研究

R. Rohmeyer らはプログラミング言語の教育には第2言語教育の知見を活用できるという仮説を立て、特に有用と考えられる第2言語教育の知見を整理した[2]。有用と考えられる知見は行動主義理論に基づく知見、生得主義の視点に基づく知見、相互作用論的な視点に基づく知見の3つである。行動主義理論に基づく知見は、構文や語彙の学習には模倣、反復、暗記が有用であるというものである。これは日本の高等学校における英語教育などでよく用いられている手法である。生得主義の視点に基づく知見は、学生は教師からではなく、友人や家族、本などから言語を教育しやすいというものである。これは授業化が困難という問題点が存在する。相互作用論的な視点に基づく知見は、会話の中での様々な相互作用に着目し学習を促進させるものである。これは解明されていないことが多い分野であることが知られている。

2.3 C. Frederick らの研究

C. Frederick らは、第2言語教育分野のベストプラクティスを大学のプログラミング講義に援用した教育法である Second Language Acquisition in a Blended Learning (SLAa-Ble) を提案し、学生の試験結果から提案手法を評価した[3]。SLAa-Ble の教育方針は学生の習熟度に応じて教育方針を変えることと受動的でない授業を行うことの2点が大きな特徴である。C. Frederick らは提案手法の有用性を示すために86名のSLAa-Bleで授業を行った群と80名

A Study on the Application of Second Language Education in Natural Language for Programming Language Education
[†] Kodai Ishikura, Gendo Kumoi, Manabu Kobayashi,
 Michiko Nakano, Shigeichi Hirasawa: Waseda University
[‡] Katsuyuki Umezawa, Shonan Institute of Technology

の従来通りの授業を行った群の2つに分け評価を行った。評価指標は2つの群の期末試験の点数を用いる。また、SLAa-BLeと従来の講義方法では同様にオンライン講義と対面での演習というハイブリッド講義の形式で授業を行った。SLAa-BLeの講義を受けた学生は従来の授業法に比べると優秀な成績評価を受ける学生が増加したものの、期末試験の結果からは統計的に有意な差は見られなかった。

2.4 R. P. Kena の研究

R. P. Kena は、欧州言語共通参照枠 (CEFR) に着目している[4]。このCEFRはヨーロッパ全体で外国語の学習・教授・評価のためのガイドラインとして策定されたものである。このガイドラインを参考とし、プログラミングの習熟度を自己診断可能なガイドライン Programming skills self-assessment matrix (PSSM) として提示している。PSSMは、コーディングスキルやコードの解読スキルなどシステム開発に必要とされる8つの項目に対して、CEFRと同様に習熟度を大きく3段階に分類している。このようなガイドラインは教育において自身の習熟度を把握することはもちろんのこと、就職活動などにおいて自身の習熟度を明確に示すことが可能である。

3. 統合的な知見に関する考察

先行研究の調査により、プログラミング言語教育は多方面から第2言語教育との明確な類似性があることが分かった。また、その類似性に着目し、自然言語教育分野である第2言語教育の知見を援用し学習効率を向上させた実証実験事例を示し、前節で有用な知見を整理した。これらの研究により、学習過程においてプログラミング言語と自然言語の類似している部分があり、自然言語教育で得られた知見をうまく援用することによってプログラミング言語における学習効率を高めることができる可能性は大いにあると考える。自然言語教育のベストプラクティスの中でも特にC. Frederickらのように学生を理解度によってポジショニングし、理解度に応じた指導を行うことは非常に興味深い。C. Frederickらの実証実験では、この習熟度の段階を授業の進捗とリンクさせており、全ての学生が同じ習熟度レベルに属していた。しかし、習熟度は学生によって個人差が当然あるため、習熟度を個別に判別し、それに応じた指導を行う必要がある。さらには、指導法だけでなく試験内容、教材、講義速度などの指導方針を習熟度別にカスタマイズすることでより一層の学習効果の向上が期待される。これらの第2言語教育の知見の援用をはじめとするプログラミング言語の教育方法については研究の余地が十分にあると考える。また、これらの学習効果の向上が期待される知見やこれ以外の知見も次世代のプログラミング教育システムへ導入を検討する必要がある。

これらの知見からプログラミング教育において受講者のレベルを把握することは必要不可欠であると

考えられる。このため、CEFRの知見をプログラミング教育に取り入れることは重要であると考えられる。CEFRは表1に示すように学習者全体を3段階に大別している。また、このCEFRは各種試験と対応付けられ、TOEFLの点数によって客観的な判別が可能となっている。これをプログラミング教育に援用した場合の考え方はR. P. Kenaが示している。一方で、受講者がどの段階にあるかという点においては、Kanaは自己評価としている。そのため、客観的な評価を行う上では、CEFRのように各種試験にレベルを対応させることが必要であると考えられる。

表1. CEFRとPSSM

レベル群	CEFR	PSSM
A	基礎段階の言語使用者	指導の下でプログラミングが可能なる者
B	自立した言語使用者	最低限の助言でプログラミングが可能なる者
C	熟達した言語使用者	自立してプログラミング可能で指導も可能なる者

4. まとめと今後の課題

本研究では第2言語教育とプログラミング言語教育関連性や第2言語教育の知見のプログラミング教育への援用についての調査を行った。また、それらの内容を踏まえた統合的な知見を考察した。今後の課題としては、CEFRと各種試験のように、既存のプログラミングにおける試験と受講者レベルの対応付けの検討が挙げられる。その上で、一般的にはレベル別になっているプログラミング各種試験をTOEFLのように全段階の学習者を1つの試験で判定可能な定量化手法の検討が望まれる。

謝辞

本研究はJSPS科研費JP19H01721の助成による。

参考文献

- [1] Pandza, N. B., "Computer Programming as a Second Language," *Advances in Human Factors in Cybersecurity*, pp439-445, 2016.
- [2] Rohmeyer, R., et al., "A human factors perspective on learning programming languages using a second language acquisition approach," *2017 ASEE Zone II Conference*, 2017.
- [3] Frederick, C., et al., "Work in Progress: Using Second Language Acquisition Techniques to Teach Programming—Results from a Two-Year Project," *ASEE Annual Conference*, 2017.
- [4] Kena, R. P., "How good are you at programming? A CEFR-like approach to measure programming proficiency," <https://drknz.net/programminglevels.html>, 2018.