

# 汎用事前学習モデルを用いたアルゴリズムの手続き自動生成

大西朔永<sup>1</sup> 椎名広光<sup>2</sup><sup>1</sup> 岡山理科大学大学院総合情報研究科 <sup>2</sup> 岡山理科大学総合情報学部

i20im01os@ous.jp shiina@mis.ous.ac.jp

## 概要

プログラム作成においては、プログラムの目的に必要な処理に分解する論理的思考力が重要となる。しかし、プログラム作成は論理的思考力に加えて、プログラム言語の知識等も同時に必要となる。これまで、アルゴリズムを考える力である論理的思考力を養う手続き学習システムを開発してきた。手続き学習システムの問題作成等では、問題文と必要な処理に分解した手続きを大量に作成する必要がある。論理的思考力には汎用知識が必要である場合も多いが、アルゴリズム手続き生成の大規模な学習データの開発は容易ではない。本研究では、汎用事前学習モデルを用いた問題文からプログラム手続きを生成する手法を提案する。

## 1 はじめに

小学校では2020年からプログラミング教育 [1] が導入されており、その目的は主に論理的思考力の育成である。論理的思考力は、様々な問題を解決するために必要な処理内容や処理順序を考える能力である。論理的思考力を育成する研究では、手作業によるアルゴリズム学習の研究 [2] がある。我々は論理的思考力を養う手続き学習システム [3] を開発してきたが、プログラムの概要を表す問題文と手続きが大量に必要となる。人手によるプログラム手続きの作成は、コストが高く、容易ではない。そこで、我々はこれまでにプログラムのソースコードからプログラム手続きを生成する手法 [4]、ソースコードを用いないプログラムの問題文からプログラム手続きを生成する手法 [5] を提案している。

近年、大規模なデータを用いた大規模な事前学習モデルの研究が盛んに行われている。そして、その学習済みモデルを個々のタスクに fine-tuning する手法が提案され、様々な自然言語処理分野で成果を出している。一方、本研究では学習に利用できるデータは小規模であり、大規模な事前学習モデルの構築

は困難である。そこで、ソースコードを用いない問題文からのプログラム手続きの自動生成において、汎用事前学習モデルを用いた Seq2Seq [6, 7, 8] を小規模なデータに fine-tuning する手法を提案する。プログラム手続きの自動生成には、問題文を必要な手続きに分解する論理的思考力が必要である。入力される問題文から特徴を捉えた分散表現が作成されないと、論理的思考力を発揮できない。汎用事前学習モデルは大規模なコーパスから言語理解の能力に加えて、汎用知識も獲得していると考えられる。プログラムの問題文からプログラム手続きを生成する手法に対して、汎用事前学習モデルを適用することで、生成する手続きの改善を図っている。

また、本研究では学習に小規模なデータを用いるため、手続き生成タスクの学習による汎用事前学習モデルの fine-tuning が十分でないと考えられる。そこで、fine-tuning した汎用事前学習モデルを用いたプログラム手続きの生成手法を提案する。汎用事前学習モデルである BERT [9] 単体の fine-tuning を pre-training と同様の方法で行い、プログラムの問題文というドメインへの適応を先に図る。そして、fine-tuning した BERT を用いてプログラム手続きの生成タスクを学習することで、タスクへの適応を行う二段階の手法である。

## 2 関連研究

プログラミング教育の分野において、エラーを可視化する学習支援ツールの研究 [10] や操作ログを用いた初心者向け教育の研究 [11, 12]、カードの操作を基本とした学習システムの研究 [13] が行われている。プログラミング力と論理的思考力との相関に関する分析 [14] では、論理的思考力と言語能力を必要とする論理的文章作成力とプログラミング力の関係を分析し、プログラミング力は論理的思考力を主に必要とすることが示されている。

汎用事前学習モデルである BERT は、双方向の Transformer [15] を用いたモデルである。トークンの

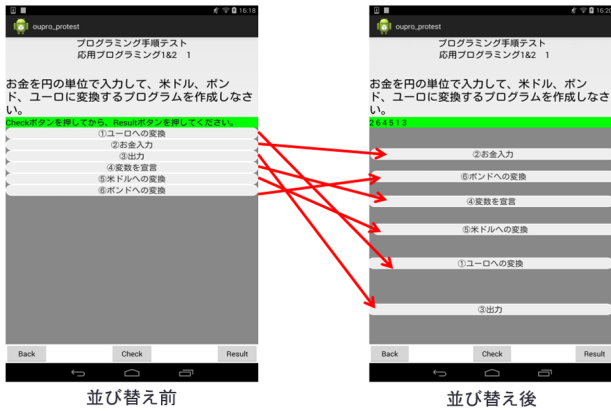


図1 手続き学習システム

分散表現を出力する点は Word2Vec[16] と同様であるが、BERT はコンテキストを考慮した分散表現が得られる。日本語に対応した BERT の事前学習モデルは幾つか公開されており、本研究では東北大学の乾研究室が公開している BERT[17] を利用している。

事前学習済みの BERT をニューラル機械翻訳に利用した研究 [18] では、Transformer ベースの機械翻訳モデルの Encoder 部分を BERT に置き換えるモデルが提案されている。

### 3 手続き学習システム

プログラムには、プログラムの目的を達成するために必要な手続きを正しい順序で記述する必要がある。我々は、手続きをプログラムが動作する順序に並び替える問題を解くことで、論理的思考力を養うシステムを開発してきた。手続き学習システムを図 1 に示す。手続き学習システムでは、最初にプログラムの問題文と必要な手続きを左の画面のように表示する。例の為替のプログラムでは、問題文と 6 個の手続きが表示されている。次にユーザは手続きをプログラムが動作する順序に並び替える。右の画面はユーザが並び替えた後の画面である。最後に手続きの順序の正解、不正解を表示する。

## 4 プログラム手続きの生成

### 4.1 GRU による Seq2Seq を用いたプログラム手続きの生成 (ベースライン)

ベースラインとして、GRU[19] による Seq2Seq を用いた問題文からプログラム手続きを生成する手法について述べる。プログラムの問題文とプログラム手続きを対訳データとして、Seq2Seq を学習する。GRU による Seq2Seq を用いた問題文からプロ

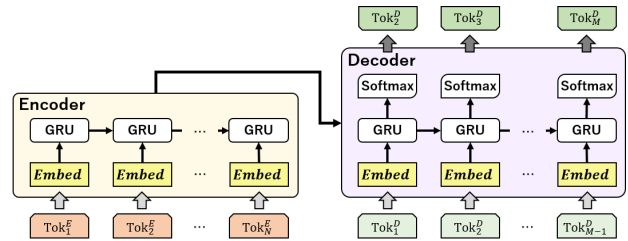


図2 GRU による Seq2Seq を用いた問題文からプログラム手続きを生成するモデルの構造

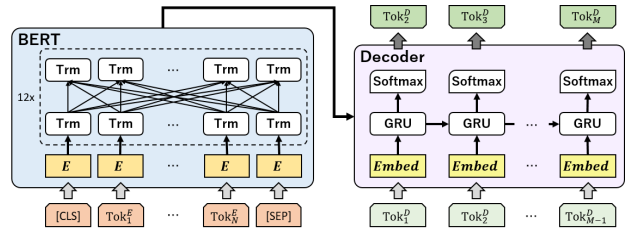


図3 BERT を用いた問題文からプログラム手続きを生成するモデルの構造

グラム手続きを生成するモデルの構造を図 2 に示す。また、モデルの定式化を式 (1) に示す。Encoder では問題文のトークン  $Tok_1^E, \dots, Tok_N^E$  を順に入力し、問題文全体の特徴を捉えたベクトル  $h_N$  を作成する。そして、Decoder がそのベクトル  $h'_0 = h_N$  を用いて、問題文に対応するプログラム手続きのトークン  $Tok_2^D, \dots, Tok_M^D$  を生成する。 $h'_{t-1}$  は GRU の前の隠れ状態である。ただし、隠れ状態の初期値  $h'_0$  は Encoder 側の GRU の最後の隠れ状態  $h_N$  である。

$$\begin{aligned}
 & p(Tok_2^D, \dots, Tok_M^D | Tok_1^E, \dots, Tok_N^E) \\
 &= \prod_{t=2}^M p(Tok_t^D | h'_{t-1}, Tok_1^D, \dots, Tok_{t-1}^D) \quad (1)
 \end{aligned}$$

英語と日本語の翻訳等とは異なり、短文(問題文)から長文(プログラム手続き)を生成するため、抽象的な情報を具体的な情報に分解する必要がある。事前学習をしていないため、大規模なコーパスから得られる汎用知識は期待できず、分散表現に含まれる情報は対訳データに含まれる情報のみとなる。つまり、生成する際に必要な汎用知識等の情報が不足し、モデルは問題文をプログラム手続きに分解する論理的思考力を発揮できないと考えられる。

### 4.2 汎用事前学習モデルを用いたプログラム手続きの自動生成

本研究では、汎用事前学習モデルの BERT を用いた問題文からプログラム手続きを生成する手法を提案する。BERT を用いた問題文からプログラム手続きを生成するモデルの構造を図 3 に示す。モデル

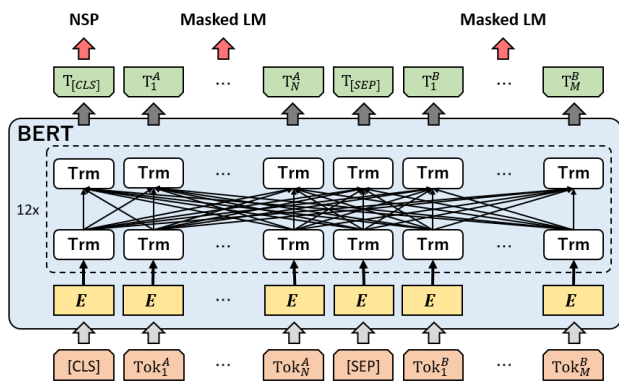


図4 BERTのpre-trainingと同様の方法によるBERT単体のfine-tuning

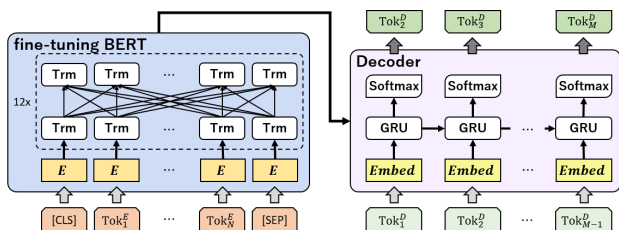


図5 fine-tuningしたBERTを用いた問題文からプログラム手続きを生成するモデルの構造

の入出力はベースラインの手法と変わらず、プログラムの問題文とプログラム手続きを対訳データとして、モデルを学習する。ベースラインのSeq2SeqではGRUを用いたEncoderを使用しているのに対し、このモデルではEncoderをBERTに置き換えている。汎用事前学習モデルであるBERTはコンテキストを考慮した分散表現を作成でき、汎用知識も獲得していると考えられる。BERTを用いることで、Decoderが利用できる分散表現の情報が増え、生成するプログラム手続きが改善されると考えている。

### 4.3 fine-tuningした汎用事前学習モデルを用いたプログラム手続きの自動生成

汎用事前学習モデルのBERTを用いたプログラム手続きの生成手法では、プログラム手続きの生成というタスクを学習する過程でBERTもfine-tuningしている。つまり、プログラム手続きの生成というタスクへの適応に加えて、プログラムの問題文という特定ドメインへの適応も同時に行っていると捉えることができる。本研究では学習に小規模なデータを用いるため、プログラムの問題文というドメインへの適応を先に行うことで更なる改善を図れると考えている。本手法では、BERT単体のfine-tuningをBERTのpre-trainingと同様の方法で行い、特定ドメインへの適応を図る。そして、fine-tuningしたBERT

表1 生成されたプログラム手続きのBLEUによる平均評価

モデル	BLEU
Seq2Seq GRU	0.3113
Seq2Seq BERT	0.3721
Seq2Seq fine-tuning BERT	<b>0.4059</b>

を用いてプログラム手続きの生成というタスクを学習することで、タスクへの適応を行う。

BERTのpre-trainingと同様の方法によるBERT単体のfine-tuningの概要を図4に示す。プログラムの問題文をコーパスとして、fine-tuning用のデータを作成している。fine-tuningの手法はpre-trainingの手法と同様に、Next Sentence Prediction(NSP)とMasked LMを用いている。また、fine-tuningしたBERTを用いた問題文からプログラム手続きを生成するモデルの構造を図5に示す。ベースラインのSeq2SeqのEncoderをプログラムの問題文というドメインにfine-tuningしたBERTに置き換えている。

## 5 各モデルの評価

### 5.1 実験環境

評価するモデルは、ベースラインであるGRUを用いたSeq2Seq、BERTを用いたSeq2Seq、fine-tuningしたBERTを用いたSeq2Seqの3種類である。データセットは、大学の情報科学科の講義で使用されたC言語のプログラムの問題文とプログラム手続きのペア46個を用いた。評価方法には、小規模データである点を考慮し、5-Fold Cross Validationを用いた。各テストの平均を評価値としている。評価指標には自動機械翻訳評価指標であるBLEU[20]を用い、生成したプログラム手続きを評価した。

### 5.2 評価指標BLEUを用いた評価

生成されたプログラム手続きのBLEUによる平均評価を表1に示す。EncoderにBERTを用いたモデルは、GRUのEncoderを用いたモデルと比較して、BLEUが約0.06向上している。汎用事前学習モデルであるBERTの有効性が確認できた。また、fine-tuningしたBERTを用いたモデルでは、BERTを用いたモデルと比べて、BLEUが約0.03向上している。あらかじめプログラムの問題文というドメインにfine-tuningし、タスクの学習と分ける手法が有効であることが分かる。GRUのEncoderを用いたモデ

表 2 各モデルのプログラム手続きの生成例と評価

問題文	Seq2Seq GRU		Seq2Seq BERT		Seq2Seq fine-tuning BERT	
	プログラム手続き	BLEU	プログラム手続き	BLEU	プログラム手続き	BLEU
整数値を2つ読み込んで、1つ目の値が2つ目の値に等しい時は、等しいことを、小さいときは、小さいことを、大きい時は、大きいことを表示するプログラムを作成しなさい。	入出力を行うための宣言 main関数の宣言 整数型変数の宣言 整数型変数の宣言 整数型変数の宣言 整数型変数の宣言 整数型変数の宣言 整数型変数の宣言 整数を入力 入力をもつ文章を表示 整数を入力 入力をもつ文章を表示 整数を入力 None 文章の表示 None 関数を終了させる None {改行 x431}	0.0862	入出力を行うための宣言 main関数の宣言 整数型変数の宣言 入力をもつ文章を表示 整数を入力 変数の初期値を決める 入力をもつ文章を表示 整数を入力 変数の初期値を決める 入力をもつ文章を表示 整数を入力 None 文章の表示 None 関数を終了させる None	0.3683	入出力を行うための宣言 main関数の宣言 整数型変数の宣言 整数型変数の宣言 入力をもつ文章を表示 整数を入力 入力をもつ文章を表示 整数を入力 入力をもつ文章を表示 整数を入力 None 文章を表示 整数を入力 None 文章を表示 関数を終了させる None	0.4164

ルと比較すると、BLEUが約0.09向上し、汎用事前学習モデルの効果は大きい。

### 5.3 プログラム手続きの生成例による評価

各モデルのプログラム手続きの生成例と評価を表2に示す。GRUのEncoderを用いたモデルは、「整数型変数の宣言」を6回繰り返して生成している。整数値を2つ読み込むため、正しくはfine-tuningしたBERTを用いたモデルのように2回である。BERTをプログラムの問題文のドメインにfine-tuningしたことで、問題文の特徴を捉える能力が向上したと考えている。また、GRUのEncoderを用いたモデルは、改行のトークンを繰り返し生成しているため、低いBLEUになっている。2つの整数値を比較して条件分岐をし、表示する部分の手続きは、GRUのEncoderを用いたモデルでは全く生成できていない。BERTを用いた2種類のモデルでは、条件分岐の手続きは生成できていないが、文章の表示は生成できている。全体的にはfine-tuningしたBERTを用いたモデルが良い生成をしているが、過不足ない手続きの生成には改善が必要である。

## 6 おわりに

本研究では汎用事前学習モデルのBERTを用いた問題文からプログラム手続きを生成する手法を提案した。BERTをEncoderとして用いることで、汎用知識も含めて問題文の特徴をより捉えた分散表現が得られ、生成されたプログラム手続きが改善された。さらに、fine-tuningしたBERTを用いたプロ

グラム手続きの生成手法を提案した。本研究では小規模なデータを用いたため、ドメインとタスクのfine-tuningを分割して行い、ベースラインのGRUを用いた手法と比べ、BLEUで約0.9の向上を確認した。生成例においては過不足のない手続きの生成には至っていないため、Encoder側だけでなく、Decoder側の改善が今後の課題であると考えている。

## 参考文献

- [1] Ministry of Education, Culture, Sports, Science and Technology. Elementary school programming education guide (2nd edition). [https://www.mext.go.jp/a\\_menu/shotou/zyouhou/detail/1403162.htm](https://www.mext.go.jp/a_menu/shotou/zyouhou/detail/1403162.htm), 2018. Accessed May. 5, 2020 (in Japanese).
- [2] Junko Shinkai, Yoshikazu Hayase, and Isao Miyaji. A trial of algorithm education emphasizing manual procedures. In Gregory Chamblee and Lee Langub, editors, **Proceedings of Society for Information Technology & Teacher Education International Conference 2016**, pp. 113–118, Savannah, GA, United States, March 2016. Association for the Advancement of Computing in Education (AACE).
- [3] Kohei Sakane, Nobuyuki Kobayashi, Hiromitsu Shiina, and Fumio Kitagawa. Kanji learning and programming support system which conjoined with a lecture. In **IEICE Technical Report**, Vol. 114 of **ET2014–86**, pp. 7–12, 2015.
- [4] Sakuei Onishi, Akiyoshi Takahashi, Hiromitsu Shiina, and Nobuyuki Kobayashi. Automatic comment generation for source code using external information by neural networks for computational thinking. **International Journal of Smart Computing and Artificial Intelligence(IJSCAI)**, Vol. 4, No. 2, pp. 39–61, 12 2020.
- [5] Sakuei Onishi and Hiromitsu Shiina. Two-stage program procedure generation from problem statements us-

- ing the seq2seq combination. In **2021 10th International Congress on Advanced Applied Informatics (IIAI-AAI)**, pp. 406–412, 2021.
- [6] Ilya Sutskever, Oriol Vinyals, and Quoc V Le. Sequence to sequence learning with neural networks. In **Advances in Neural Information Processing Systems 27 (NIPS 2014)**, pp. 3104–3112, 2014.
- [7] Thang Luong, Hieu Pham, and Christopher D. Manning. Effective approaches to attention-based neural machine translation. In **Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing**, pp. 1412–1421, Lisbon, Portugal, September 2015. Association for Computational Linguistics.
- [8] Alexander M. Rush, Sumit Chopra, and Jason Weston. A neural attention model for abstractive sentence summarization. In **Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing**, pp. 379–389, Lisbon, Portugal, September 2015. Association for Computational Linguistics.
- [9] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. **arXiv preprint arXiv:1810.04805**, 2018.
- [10] Haruki Kanamori, Takahito Tomoto, and Takako Akakura. Development of a computer programming learning support system based on reading computer program. In Sakae Yamamoto, editor, **Human Interface and the Management of Information. Information and Interaction for Learning, Culture, Collaboration and Business**, pp. 63–69, Berlin, Heidelberg, 2013. Springer Berlin Heidelberg.
- [11] Koki Okimoto, Shimpei Matsumoto, Shuichi Yamagishi, and Tomoko Kashima. Developing a source code reading tutorial system and analyzing its learning log data with multiple classification analysis. **Artificial Life and Robotics**, Vol. 22, No. 2, pp. 227–237, apr 2017.
- [12] Shimpei Matsumoto, Koki Okimoto, Tomoko Kashima, and Shuichi Yamagishi. Automatic generation of c source code for novice programming education. In Masaaki Kurosu, editor, **Human-Computer Interaction. Theory, Design, Development and Practice**, pp. 65–76, Cham, 2016. Springer International Publishing.
- [13] Shimpei Matsumoto, Yusuke Hayashi, and Tsukasa Hirashima. Development of a card operation-based programming learning system focusing on thinking between the relations of parts. **IEEJ Transactions on Electronics, Information and Systems**, Vol. 138, pp. 999–1010, 08 2018.
- [14] Michiko Oba, Kei Ito, and Akio Shimogoori. Analysis of correlation between programming skills and technical writing skills. In **IPSJ SIG Technical Report**, Vol. 2015-IFAT-118, pp. 1–4, 2015.
- [15] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In **Advances in neural information processing systems**, pp. 5998–6008, 2017.
- [16] Tomás Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. Efficient estimation of word representations in vector space. In Yoshua Bengio and Yann LeCun, editors, **1st International Conference on Learning Representations, ICLR 2013, Scottsdale, Arizona, USA, May 2-4, 2013, Workshop Track Proceedings**, 2013.
- [17] 東北大学 乾研究室. cl-tohoku/bert-base-japanese-whole-word-masking · hugging face. <https://huggingface.co/cl-tohoku/bert-base-japanese-whole-word-masking>, 2020.
- [18] Kenji Imamura and Eiichiro Sumita. Recycling a pre-trained BERT encoder for neural machine translation. In **Proceedings of the 3rd Workshop on Neural Generation and Translation**, pp. 23–31, Hong Kong, November 2019. Association for Computational Linguistics.
- [19] Kyunghyun Cho, Bart van Merriënboer, Caglar Gulcehre, Dzmitry Bahdanau, Fethi Bougares, Holger Schwenk, and Yoshua Bengio. Learning phrase representations using RNN encoder–decoder for statistical machine translation. In **Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)**, pp. 1724–1734, Doha, Qatar, October 2014. Association for Computational Linguistics.
- [20] Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. Bleu: a method for automatic evaluation of machine translation. In **Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics**, pp. 311–318, Philadelphia, Pennsylvania, USA, July 2002. Association for Computational Linguistics.