様　式　Ｃ－１９、Ｆ－１９－１、Ｚ－１９　（共通）

科学研究費助成事業　　研究成果報告書

令和　６ 年　 ６ 月 １０ 日現在

機関番号：１２５０１
研究種目：挑戦的研究（開拓）
研究期間：2018～2023
課題番号：１８Ｈ０５３２１・２０Ｋ２０３４０

研究課題名（和文）トポロジーの形式化における深層学習の適用の研究

研究課題名（英文）Applying deep learning to formalization of Topology

研究代表者
久我　健一（Kuga, Kenichi）

千葉大学・大学院理学研究院・名誉教授

研究者番号：３０１８６３７４

交付決定額（研究期間全体）：（直接経費）　　16,130,000円

研究成果の概要（和文）：証明支援ソフトウェアCoq/SSReflectを用いた証明ファイル(.vファイル）を処理し、証明ステップ(tactic)予想をタスクとする機械学習用のデータファイルを作成するpythonプログラムを作成し、これを用いて既存の.vファイルから400Mb程度の複数のデータセットを作成した。このデータセットを用いてtransformerタイプの深層学習モデルを訓練した。モデルとトークナイザの選択や、ハイパーパラメータの調整等で、予測精度の向上を目指しているが、topology等の数学定理の形式化が進行する精度には現時点ではまだ至っていない。

研究成果の学術的意義や社会的意義
数学の電子化（機械による形式化）やソフトウェアの検証の自動化は科学の基盤を確固とする意味でも、またAI等のソフトウェアの信頼性のある実行のためにも意義がある。このための機械学習は、既存データの絶対数が少なく、またデータセットは証明支援システムと対話的に作成する必要があるので、汎用性が高く、かつ学習が理解しやすいデータセットを作成することが重要である。本研究で得られたデータセットはtopology等の現代数学の分野の定理の形式化に対して実用的な精度の予測の達成には到っていないが、簡明で可読性が高いので、実用的な自動証明・形式化につながる研究である。

研究成果の概要（英文）：We created python programs that execute .v files of proof assistant Coq/SSReflect and produce datasets for machine learning tasks of predicting next proof steps, i.e., tactics in theorem formalization. We apply these programs on existing .v files and produced couple of datasets of size about 400Mb. We trained deep learning models of transformer types using these datasets. Although we aimed at improving prediction accuracy through the choices of deep learning models and tokenizers and through the choices of hyper parameters, we haven't reached a practical accuracy that make the actual formalization of mathematical theorems from topology e.t.c. proceed, at the time of writing this report.

研究分野：topology

キーワード：深層学習　証明支援系　形式化

１．研究開始当初の背景

証明支援ソフトウェアの使用によって、これまで専ら数学者が行なってきた数学定理の証明の検証が、対話的環境で機械的に行えるようになってきた。実際、４色定理、奇数位数定理、ケプラー予想といった、検証が難しいとされてきた定理が、機械的に検証できることが実証されてきた。このような機械的検証（コンピュータを用いた証明の形式化）は、コンピュータが定理証明を完全に記録し検証するということであり、達成されれば意義は大きい。

しかし、たとえ証明支援ソフトウェアを用いても、対話的環境で、定理の証明を形式化することは、簡単な定理であってさえ、多大な労力と時間を要し、数学の多くの定理や分野などを形式化することは、現実的には非常に困難である。

その一方で、深層学習を用いた機械学習によって、囲碁などのゲームや画像認識などで、コンピュータが人間の知的活動を代替できる可能性が広がってきた。

そこで、この深層学習を数学支援ソフトウェアの使用に適用し、数学定理や分野の形式化の労力軽減・自動化の可能性を探り、さらにこれを活用してトポロジーをはじめとする数学の形式化に繋げたいと考えた。

２．研究の目的

（１）数学支援ソフトウェアの一つ Coq を用いた対話的定理証明において、証明の各ステップで発行する命令(tactic)の予測に深層学習（申請時においては LSTM などの RNN、後に Transformer タイプのモデル）を適用する可能性を探る。汎用プログラミング言語の定型的なプログラムに対する AI 支援は、既存の膨大なプログラミングデータをそのまま学習し、次の文章予測としてある程度の予測精度を上げることが実証されつつあるが、数学証明の形式化については、そもそも証明データの絶対数が少ないことに加え、定型的に形式化できない部分が本質であるため、実用的な予測制度を達成するためには､証明支援ソフト Coq から対話的に返される証明項を予測に利用することが最小限必要である。そこでデータ化の段階から Coq を実行し逐次応答を取り出して学習データを作成する。このとき、内部証明項そのものでは、情報が多すぎて実用的訓練データに適さないことに加え、人間に理解しづらいため、精度改善の方向が見えない点などに対処する必要があり、訓練用のデータセットのフォーマットは、できるだけ簡略化し、理解可能なデータを用いて設計する必要がある。

（２）これを利用しトポロジーを手始めに数学定理の電子化（形式化）を進める。これは自然言語から Coq 文への翻訳と、Coq による証明の自動化の２つの側面がある。自然言語から Coq 文への翻訳では、まず自然言語での数学証明の収集が必要である。数学証明においても自然言語は均一でないので、検索・均一化の中間的なデータベースを作成する。このデータベースの各データは証明の全てのステップが自然言語で明示的に表現されているものでなければならない。証明の各ステップが十分小さければ、各ステップの Coq 文への翻訳は基礎的で定型的なものになると考えられるので、（１）のデータセットを用いた学習で、自動的に証明が進む可能性が生じる。

３．研究の方法

（１）証明支援ソフトウェア Coq/SSReflect を用いた数学定理証明ファイルを実行して各ステップで返ってくるレスポンスを記録し、直後に実行された tactic(とその引数）を記録する python プログラムを作る。ここで重要な点は、レスポンスとして Coq の内部の証明項を直接取り出した場合、情報量過多でそのままでは学習データとして有効でないと考えらること、また内部的な証明項は人間が理解しづらいので、学習を改善していく研究方向を探ることが難しいことである。そこで、最初の方針として、最も簡単で人間に状況が把握しやすいデータとして、coqtop -emacs モード あるいは coq-sertop の human モードで返される文字列を、そのまま使用することを考える。例えば、implicit arguments も明示しないことが、深層学習モデルの学習に有効であると考える。また予想ステップとしては最小限の次の tactic とその引数を目標とする学習データを考える。以降の研究において、ライブラリを分野的に制限することや、tactic ごとに引数を学習するデータセットを作成するなど、様々なバリエーションが考えられるので、ここでのデータセット作成プログラムは、最も簡明で、汎化、変更、改変が容易なものを作成する。

（２）インターネット上にある Coq/SSReflect による定理証明（ソフトウェア検証も含む）ファイルやライブラリー群を収集し、方法（１）で作成したプログラムを用いて深層学習用のデータセットを作成する。これと同時に、インターネット上にある自然言語による証明データとして機械学習に繋がりやすいものを収集し、検索、データベース化する。さまざまな情報源があるが、このような多くのデータは pdf ファイルであり、特に数学記号や数式をデータ化することがげ現時点で難しいので、html ファイルとして収集できるものを考える。特に proofwiki は証明ステップも細かいので、収集する。

（３）得られたデータセットをレスポンスから tactic(とその引数）を予想するタスクとして深層学習モデルを訓練する。どのようなモデルを使用するか、トークナイザをどう作成するか、ハイパーパラメータの設定など、またデータセットの構成の再考などを考える。

（４）python プログラムから対話的に Coq と Pytorch を実行し、Coq からのレスポンスを学習済みの深層モデルに送り、tactic を予想し、実行する。これはもちろん(3)の学習の達成度に依存する。

４． 研究成果

証明支援ソフトウェア Coq/SSReflect を用いた数学定理証明ファイルを実行して各ステップで返ってくるレスポンスを記録し、直後に実行された tactic(とその引数）を記録する python プログラム群を作成した。また、インターネット上にある Coq project を収集し、これを用いて種々の訓練データを作成した。

その中の一つは Coq スタンダードライブラリの theories から作成したもので、441Mb の csv ファイルである。各行は 2 項目からなり、第 1 項目は Coq を emacs 上で利用するときに対話的に返される

情報であり、第 2 項 目はそれに対する Tactic とそれに与えられるパラメータである。また第 1 項目には、現れる各項のタイプ情報も付与されているが、これは Show All で得られる情 報であり、特殊なプロトコルに依存しないので、汎用性が高いと同時に、証明ファイル中の証明の避けられない依存性を若干緩和し、各行が比較的独立しているので、訓練結果の検証がある程度可能と考えている。その他の同様の訓練データとして、mathcomp の theories 177Mb, Topology 7.2Mb, CompCert に関するもの 2Mb, odd-order theorem に関するもの 42Mb, GeoCoq 30Mb 等を作成した。

これらの訓練データを用いて、Transformer 型の深層学習モデルを訓練した。一律にある程度の Tactic 予測精度が得られた。例えば、reference として 2017 年に発表された transformer のオリジナルモデル[3]を上述の coq theories の 441Mb の訓練データで学習させると、1800 ステップ/epoch で 7 epoch 訓練すると Loss 関数の値は 25 パーセント程度に減少する。

しかし、調べてみると、これらの予測精度の向上は、自明な intros や auto 等の寄与で実際より良く出ており、Coq/SSReflect による形式化の実質的な部分を占める rewrite や apply といった引数を必要とする tactic の予測は実用的なレベルに達していないことがわかった。このことは、rewrite の引数や、apply の引数としての theorem emmbedding 等の訓練が別に必要であることを示していると考えられるが、その点では現時点では成果が出ておらず、今後の研究課題である。

深層学習を用いて証明支援系による証明の自動化を目指した先行研究に CoqGym [1] と GamePad [2]がある。これらはいずれも LSTM モデルを想定しており、{1}では serapi と前処理によるメタファイルの作成を利用し、また[2]では coq プログラムそのものに情報を取り出すコードを挿入し(tcoq) て coq 内部のデータを可能な限り利用できるようにしてある。 これらの研究の目的は機械学習の環境を用意する点にあり、tactic 予測精度そのものが主目標ではない。その後[3]を契機とする Transformer 型の深層学習モデルの出現によって、特に BERT モデルの事前学習法に見られるように、内部データの詳細を明示することがコンピュータによる内容理解につながる訳ではないことが認識されてきており、この傾向はここ数年のプロンプトプログラミングなどにおいても、特に顕著である。当該研究の目指している方向は、取り出すデータを人間が対話的環境で得られるものをそのまま用い、コンピュータそのものによる内容理解を促進する学習データの作成を目標としており、この点で先行研究と異なっている。

上述の訓練データ作成のための python プログラム群は GitHub [5] に公開してある。_CoqProject ファイルに従って coqtop -emacs からの応答を記録して学習データを作成するもので、serapi 等のプロトコル等の知識は必要ない。作成した PICoq クラスを使うと pytorch のニューラルネットモデルと coqtop を連携させて対話的に実行することが可能である。これとは別にウェブ上の Proofwiki や n Lab 等の証明データベースを作成した[4]。 これは研究目的の（２）の後半で述べた、自然言語による証明と Coq 文との間をつなぐ研究の一環である。

なお、以上の研究でデータベースの作成と Transformer モデルの学習で花輪和孝氏の協力を得ている。

[1] K. Yang and J. Deng, “Learning to Prove Theorems via Interacting with Proof Assistants”, ICML 2019 (https://github.com/princeton-vl/CoqGym)

[2] D. Huang, P. Dhariwal, D. Song, I. Sutskever, “GamePad: A Learning Environment for Theorem Proving”, ICLR 2019 (https://github.com/ml4tp/gamepad)

[3] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, I. Polosukhin, “Attention is All you Need” Advances in Neural Information Processing Systems 30 (NIPS 2017)

[4] http://163.43.192.18:8000/proofs/index3 (**ライセンスの問題から現時点では** digest **認証をかけてある：** proofs/math2019 **でログイン可能**)

[5] https://github.com/kenkuga/picoq

５．主な発表論文等

〔雑誌論文〕　計0件

〔学会発表〕　計4件（うち招待講演　1件／うち国際学会　2件）

１．発表者名 Kenichi Kuga
２．発表標題 Let Transformers speak Coq
３．学会等名 RE:BIT（招待講演）（国際学会）
４．発表年 2022年

１．発表者名 久我健一
２．発表標題 Training Transformers to formalize Topology in Coq
３．学会等名 トポロジーとコンピュータ２０２２
４．発表年 2022年

１．発表者名 久我健一、井上健太
２．発表標題 有限幾何の形式化と深層学習
３．学会等名 第３回情報理論および符号理論とその応用ワークショップ(ICA2019)
４．発表年 2019年

１．発表者名 Ken'ichi Kuga
２．発表標題 Some experiments of formalizing finite/projective geometry using Monte Carlo tree search
３．学会等名 American Mathematical Society, Central and Western Joint Sectional Meeting（国際学会）
４．発表年 2019年

〔図書〕　計0件

〔産業財産権〕

〔その他〕

(1)　https://github.com/kenkuga/picoq　: Formal proof datasets and simple python utilities to interact with coq and create the datasets.
(2) http://163.43.192.18:8000/proofs/index3: 自然言語による数学証明データベース。現時点ではdigest認証がかけてある: proofs / math2019 でログイン可能
(3) https://163.43.192.18:8000/static/jscoq_test/index2.html: 上記データベースに付随するonline形式化サイト（jscoq)

6．研究組織

	氏名 （ローマ字氏名） （研究者番号）	所属研究機関・部局・職 （機関番号）	備考
研究協力者	花輪　和孝 (Hanawa Kazutaka)		数学証明データベースの作成と、Transformerモデルの訓練・検証で協力を得た
研究協力者	井上　健太 (Inoue Kenta)		有限射影幾何のSSReflectによる形式化で協力を得た

7．科研費を使用して開催した国際研究集会

〔国際研究集会〕　計0件

8．本研究に関連して実施した国際共同研究の実施状況

共同研究相手国	相手方研究機関