

機械学習を用いた Coq 上の命題論理の自動証明

金原 雅典^{a)} 佐藤 亮介^{b)} 鵜林 尚靖^{c)} 亀井 靖高^{d)}

概要: プログラムの正しさを形式的に保証することは、そのプログラムに誤りがないことを確認する点で開発者にとって有用である。形式的に保証することは大変であり、自動化することでその手間を減らすことができる。そこで本研究では、機械学習を用いた命題の自動証明を試みる。その第一段階として、対象の論理を命題論理に限定して既存の機械学習によって学習・自動証明が可能かどうかを検証する。本研究での“証明”は、自然言語により書かれたものではなく、定理証明支援システム Coq 上での証明である。Coq とは、プログラムの正しさの証明や安全なプログラムを書くことに使用される証明支援システムである。学習は既存の機械学習モデルであるシーケンス変換モデルを用いて行う。これにより、学習に用いるデータセットが十分な量あれば既存の機械学習の方法でも学習可能かどうかを検証する。

1. はじめに

プログラムの正しさを証明することは、プログラムに誤りがないということを保証し、膨大なテストケースを検査したり余計なデバッグ作業を無くす、あるいは減らすことに貢献する。形式的に保証することは大変であり、その証明作業を自動化することができれば、多くのプログラムの正しさを証明する手間を大幅に減らすことができる。プログラムの正しさを証明することで、開発の効率化及びプログラムの信頼性の向上が見込める。

現在機械学習は、機械翻訳を用いた自然言語間の翻訳や、強化学習による囲碁プログラムなど、様々な分野で研究・応用され、多様な手法が考案されている。しかし、機械学習による証明の自動生成の研究はまだない。

本研究では、学習に使用するデータが十分に存在すれば既存の機械学習の方法でも学習可能かどうかということを検証するために、命題論理に限定した論理の機械学習及び自動証明を行う。有限の範囲の命題論理の証明のデータセットを網羅的にプログラムで作成し、シーケンス変換モデル^{*1}を使用して学習させる。

本研究での“証明”は自然言語により書かれたものではなく、定理証明支援システム Coq 上で書かれたものである。ここでは、Coq での証明を既存の機械学習のシーケンス変換を用いて自動証明を試みる。

2. 学習対象とする証明

本研究では定理証明支援システム Coq 上での証明を対象として学習を行う。Coq とは定理証明支援システムであり、プログラムの正しさを証明することを手助けするツールである。これは Curry-Howard 同型対応の、論理とプログラムの対応を利用してそのプログラムが論理的に正しいことを証明するものである。Coq では証明すべき命題を定義し、それに Coq に対する計算命令である Tactic を順に適用させていくことで、証明の副目標であるゴールや仮定を変化させ、命題の正しさを保証することができる。例えば、“ $(P \wedge Q) \rightarrow (P \vee R)$ ” という命題を証明するときは表 1 のように適切な Tactic を順に適用させて証明を行う。“No more subgoals.” と表示されれば証明終了である。

しかし、どの Tactic を使用するかは証明者が選択する必要があり、Tactic の選択は命題が複雑なものになるに連れて難しく手間のかかるものになる。

本研究では、機械学習を用いてゴールと仮定から適用できる Tactic を学習させることで、その状況で適用できる Tactic を推論することを試みる。

Tactic が推論できれば、自動証明したい命題に推論した Tactic を適用し、変化後のゴールと仮定に対して再び適用できる Tactic を推論し適用する。証明が終了するまで、あるいは証明できないことがわかるまで Tactic の推論と適用を繰り返すことで Coq における命題の自動証明を行うことができる。

a) kanahara@posl.ait.kyushu-u.ac.jp

b) sato@ait.kyushu-u.ac.jp

c) ubayashi@ait.kyushu-u.ac.jp

d) kamei@ait.kyushu-u.ac.jp

*1 <https://www.tensorflow.org/versions/master/tutorials/seq2seq#sequence-to-sequence-models>

表 1 Tactic 適用例

ステップ	適用前 仮定	ゴール	適用 Tactic	操作	適用後 仮定	ゴール
1		$(P \wedge Q) \rightarrow (P \vee R)$	intro	“ $\rightarrow$ ”の前半を仮定に追加, 後半を新たなゴールにする	$H : P \wedge Q$	$P \vee R$
2	$H : P \wedge Q$	$P \vee Q$	left	“ $\vee$ ”の左側を新たなゴールにする	$H : P \wedge Q$	P
3	$H : P \wedge Q$	P	destruct H	仮定 H の“ $\wedge$ ”の右側と左側を それぞれ仮定に追加する	$H1 : P, H2 : Q$	P
4	$H1 : P, H2 : Q$	P	apply H1	仮定 H1 をゴールに適用する	<i>No more subgoals.</i>	

3. Coq 上の命題論理に対する自動証明可能かの検証

本研究では、証明全体を機械学習させる前準備として、一般的な証明も学習データが十分な量あれば既存の機械学習で学習かどうかを検証するために、まずは対象の論理を単純で扱いやすい命題論理に限定して自動証明が可能かの検証を行う。対象の命題論理は以下のような BNF 記法で表せる。 x は *True*, *False* を値に持つ自由変数, f は命題論理の式である,

$$f ::= x \mid \text{True} \mid \text{False} \mid f \wedge f \mid f \vee f \mid f \rightarrow f$$

学習には教師あり学習のシーケンス変換モデルを用いる。シーケンス変換モデルとは、入力シーケンスと出力シーケンスの組を学習させ、入力を受け取ったとき、その入力を元に推論した出力を返す学習モデルである。ライブラリは TensorFlow^{*2}の seq2seq を使用する。

本研究では仮定とゴールを順に並べたものを入力として、その仮定とゴールに適用できる Tactic の集合を出力として用意し、それを用いて学習させる。データはプログラムで自由変数の数とその名前、深さを指定して命題論理の恒真式を網羅的に生成し、その仮定とゴールに対してある Tactic が適用できるかを Coq によって一つずつ確認し、適用できる Tactic の集合を作成する。これによって仮定とゴールを入力シーケンス、Tactic の集合を出力のシーケンスとする。深さとは、自由変数・“*True*”・“*False*”を葉, “ $\rightarrow$ ”, “ $\vee$ ”, “ $\wedge$ ”を子を持つ2つ節点とした構文木の深さである。例として, “*True*”などの葉のみであれば深さ1であり, “ $P \rightarrow P$ ”など節点が1つでその子が葉であれば深さ2, “ $P \rightarrow (True \vee Q)$ ”のように深さ2の子を持つ節点が根であれば深さ3である。自由変数や深さを大きくするにつれて生成される命題論理の数は爆発的に大きくなるため、今回の検証では自由変数の個数は3、深さを3として生成を行う。網羅的に生成・確認することにより、偏りなくデータセットを用意できると考えている。

学習結果は、同じ自由変数・深さの命題論理だけでなく、学習に使用したものとは別の自由変数やより深い深さの命題論理に対しても確認する。

データセットの自由変数数と深さが適切であれば、ある程度の量を学習させることで、学習結果が得られると思われる。しかし、自由変数の数とその変数名を固定するとその変数名にしか適応できない可能性も考えられる。そのため、変数名にかかわらず学習結果を適用できるような方法を考える必要がある。

4. まとめと今後の予定

本研究では、命題論理に対象を限定した命題の自動証明が可能かの検証を行うために、Coq 上の証明に対して機械学習モデルのシーケンス変換を用いて学習を行う。現在はシーケンス変換に学習させるためのデータセットの準備を行っている段階である。

命題論理の自動証明ができると確認できた場合、一般的な命題へと学習範囲を拡張する。学習に用いるデータセットは Coq の標準ライブラリや、Coq 開発陣以外が証明し一般に公開されている Coq のライブラリ^{*3}を用いて作成する。証明の中にその証明の外部で証明された定理が使用されている場合、その定理は仮定として与えるようにする。

教師あり学習のシーケンス変換による命題論理の自動証明ができなかった場合、強化学習や深層学習を用いて学習させることを考えている。強化学習を用いて学習させる場合、Seq2SQL [1] の手法が応用できないかと考えている。Seq2SQL は強化学習によって自然言語から SQL のクエリ式へ変換するよう学習させたものである。モデルを訓練するためにデータセットを用意して強化学習のモデルの訓練を行い、シーケンスからある種の命令へと変換している点において参考になると考えている。

謝辞

本研究は JSPS 科研費 JP26240007 の助成を受けたものである。

参考文献

- [1] Zhong, V., Xiong, C. and Socher, R.: Seq2SQL: Generating Structured Queries from Natural Language using Reinforcement Learning, *CoRR*, Vol. abs/1709.00103 (2017).

^{*2} <https://www.tensorflow.org>

^{*3} <https://coq.inria.fr/opam/www/>