4K-01

オブジェクト指向プログラムの要素関係グラフを用いたクラス名のend-to-end学習

萬場 大登* 早瀬 康裕† 天笠 俊之‡ 北川 博之‡

*筑波大学情報学群情報科学類 †筑波大学システム情報系
‡筑波大学計算科学研究センター

1 序論

ソフトウェア開発において，プログラム中の識別子への命名作業は重要である．なぜならば，ローカル変数やクラス，メソッドなどの識別子名をもとに，開発者はそのプログラム要素の役割を推測するためである．識別子名が端的であるほど，開発者のプログラム理解に貢献する．

しかし，プログラム要素の役割を推測できるような適切な識別子名を与えることは開発者にとって困難な作業となることがある．それは，適切な命名に，ソフトウェア開発への熟練とドメイン知識の両方が必要なためである．

本稿では，識別子の命名支援を目的として，オブジェクト指向プログラムのクラス・メソッド・フィールドの間の関係と識別子名との関係を end-to-end 学習する．これらの要素間の関係から識別子名を推定できることは栗本らの研究 [1] によって示されており，本研究は同じ関係を用いて推定精度を高めることを目指す．

2 関連手法

識別子への命名作業を支援するため，プログラムの内容から識別子名を推薦する研究が行われている．

栗本ら [1] は，プログラムの要素関係グラフでグラフ埋め込み手法を用いることで，クラス名推薦を行った．このグラフ埋め込みでは，グラフの各ノードの分散表現とその隣接ノードの分散表現の重み付き平均が，近傍となるような学習を行う．これにより各ノードの分散表現は，「どんな要素を参照するか」，「どんな要素に参照されるか」という情報，すなわちその要素の機能や役割を表現すると考えられる．

Allamanis ら [2] は抽象構文木と変数の読み書き関係などからグラフを構築し，グラフニューラルネットワーク (GNN) によるノード間の特徴量伝播を行うことで，ローカル変数の名前の推薦や誤り検出を行った．GNN は，グラフ構造を入力にとり，その構造情報を学習に活用するネットワークである．

3 提案手法

クラスの命名作業を支援するため，プログラム要素間の関係グラフと周辺要素の名前をもとに，対象要素の識別子名 end-to-end 学習により推定する．提案手法の概要を図 1 に示す．まず，プログラム要素間の関係を表すグラフを構築する．このグラフの推薦対象以外のノードにその要素名の単語列に応じた特徴量を割り当てる．次に，各ノードに割り当てた特徴量を周辺ノードへ伝播させ，対象ノードへ伝播された特徴量からその要素名の単語の列を推定する．

本手法では，栗本ら [1] と同様の要素関係グラフを用いて，より精度良くクラス名を推定できることを目指す．まず，グラフの各ノード間で特徴量伝播を行うことで，栗本らの手法と同様に，各要素の機能や役割を表現できるようになると考えられる．一方栗本らの手法は，あくまで機能や役割が似た要素同士の分散表現を近似させており，これは必ずしも識別子名の推定に特化するものではない．栗本らの手法に対し本手法では，グラフ構造から識別子名の関係までを end-to-end 学習することで，より識別子名の推定に特化させる．

3.1 要素関係グラフの構築

プログラム要素の関係グラフを，栗本らの手法と同様の方法で構築する．図 2 の例のように，Java プログラムのクラス・メソッド・フィールドの 3 種類のプログラム要素をノードとし，それらの間の特定の関係についてエッジを張ったグラフを構築する．エッジが張

End-to-end learning of class names using graph of elements in object-oriented programs
Hiroto MAMBA*, Yasuhiro HAYASE†,
Toshiyuki AMAGASA‡and Hiroyuki KITAGAWA‡
*College of Information Science, University of Tsukuba
†Faculty of Engineering, Information and Systems, University of Tsukuba
‡Center for Computational Sciences, University of Tsukuba

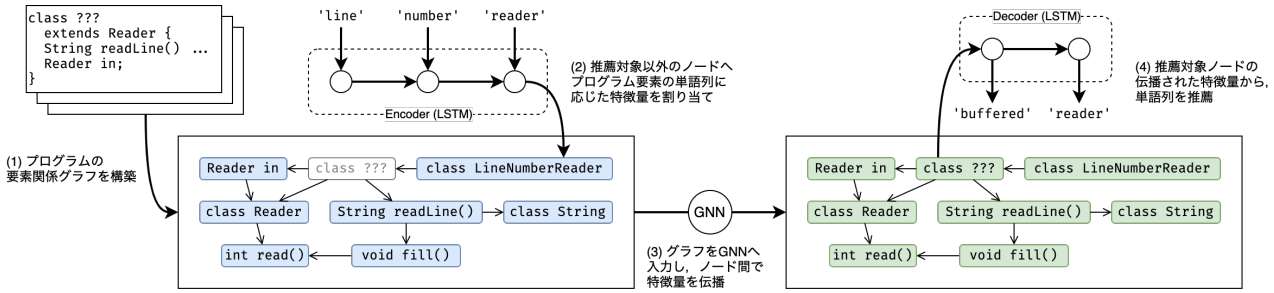

図 1: プログラムの要素関係グラフと対象ノード周辺の名前情報を入力し，対象ノード‘class ???’のクラス名を推定する例．この例での対象ノードは，図 2 のグラフにおける‘class BufferedReader’である．

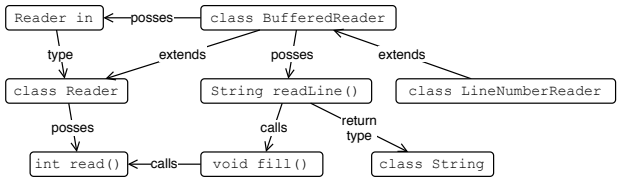

図 2: プログラム要素の関係グラフの例

られる関係には，クラス間の継承関係，クラスからメソッド・フィールドへの所有関係，メソッド間の呼び出し関係，メソッドからフィールドへのアクセス関係，メソッドからその戻り値の型として使われているクラスへの関係，フィールドからその型として使われているクラスへの関係がある．

3.2 名前の単語列を推薦するニューラルネットワーク

構築したグラフから推薦する単語列を出力するニューラルネットワークについて述べる．

構築したグラフの各ノードに，まず要素名に応じた特徴量を割り当てる．各要素について，Java プログラムに現れる名前を命名規則（キャメルケース，アンダースコア区切り）に従って単語列に分解する．例えば‘LineNumberReader’は，‘Line’, ‘Number’, ‘Reader’の 3 単語に分解する．各単語の埋め込みベクトルを LSTM (Long short-term memory) へ逐次的に入力し，LSTM の隠れ状態をそのノードの特徴量とする．

割り当てた特徴量を，GNN によって各グラフノード間で伝播させる．Allamanis らと同様の特徴量伝播を行うことで，要素関係グラフの各ノードが周辺ノードの要素に関する情報を併せ持つことが期待される．GNN は，演算方法に応じて Gated graph neural networks[3], Graph convolutional networks[4] などがある．今後，それぞれでの性能評価を行い，適当なものを検討する．

推薦対象ノードへ伝播された特徴量から，LSTM を用いて推薦する単語列を生成する．LSTM から逐次的に出力されるベクトルのうち最も値が大きい要素のインデックスが，推薦する単語の番号である．ニューラルネットワークの訓練では，LSTM が出力するベクトルと真の単語番号との交差エントロピー損失を最小化することを目的とする．

4 評価実験の計画

クラス名の推定精度を，真の名前の一部単語を推定できた割合，単語ごとの適合率・再現率・F 値によって栗本らの手法と比較する．データセットには，栗本らの手法の評価で用いられた 20 の Java プロジェクトによるもの，および GitHub でのスター数上位 1000 位までの Java プロジェクトによるものを用いる．

予備実験として，上述したデータセットのうち後者を用いて提案手法の大まかな精度を確認しており，適合率 0.2，再現率 0.06，F 値 0.1 でクラス名の単語を推定できている．

5 結論

本稿では，プログラムの要素関係グラフで end-to-end 学習を行い，クラス名を単語列で推薦する手法を提案した．今後，栗本らの手法との性能比較を行い，GNN や単語列の入出力を取り入れて end-to-end 学習を行うことの有効性を考察する予定である．

参考文献

[1] S. Kurimoto, Y. Hayase, et al. Class Name Recommendation Based on Graph Embedding of Program Elements. In APSEC 2019.

[2] M. Allamanis, M. Brockschmidt, et al. Learning to represent programs with graphs. In ICLR 2018.

[3] Y. Li, R. Zemel, et al. Gated graph sequence neural networks. In ICLR 2016.

[4] T. N. Kipf, M. Welling. Semi-Supervised Classification with Graph Convolutional Networks. In ICLR 2017.