トップエスイー修了制作

静的コード解析メトリクスを用いたバグ予測モデルの評価

NTTデータ　新井 広之　araihrb@nttdata.co.jp

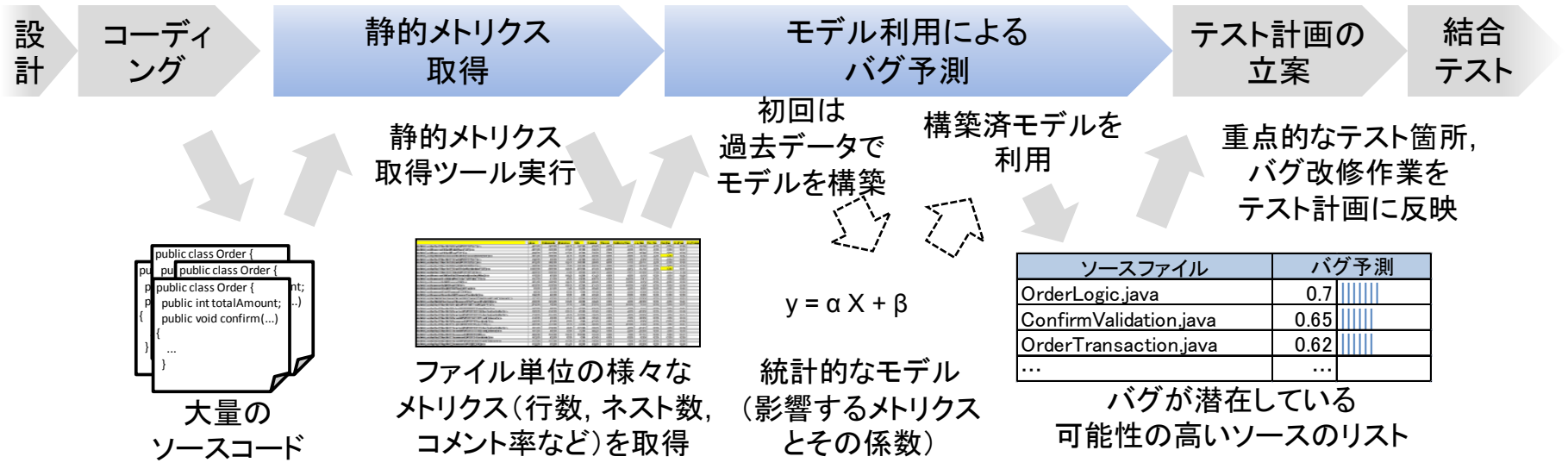
開発における問題点

ソースコードは必ずバグが潜在しているものの，そのバグがどこにあるかはわからない．そのためバグが潜在している可能性の高いモジュールを特定し，効果的に品質保証の作業を行いたい．しかし実務で統計を利用したモデルは適用経験がなく，どの手法をどのように適用すればよいかわからない．

本評価による解決

研究分野で提案されている数種の既存手法を，社内で実際に取得されているプロジェクトデータに適用し，バグの潜在している可能性の高いソースコードの検出の可能であることが検証できた．また手法の選択においても，より精度の高いものを特定できたため，効果的な品質保証活動が可能になる．

バグ予測モデル利用の流れ



評価対象の手法

本取組では、「モデル利用によるバグ予測」時に用いるモデルとして，以下の複数の手法を評価した．

手法	特徴
ロジスティック回帰モデル	応答変数が0から1の範囲に収まる。下の4つはソースファイル中のバグ数を予測するが，これはバグが含まれているかどうかの判定に利用する
ポアソン回帰モデル	1件、2件と数えられるデータの発生分布を分析する際によく用いられる
負の二項回帰モデル	用途はポアソン回帰モデルに似ているが，データの分散が大きいときに用いられる
ハードルモデル	ゼロが多い（本件ならバグがないソースファイルが多い）場合の分析用に提案されているモデル．内部的にゼロ用のモデルと1件以上の場合のモデルの2つを分けて持つ
ゼロ過剰モデル	ハードルモデルと似ているが、内部的に持つモデルのうち，「1件以上の場合のモデル」ではなく「0件以上のモデル」とし，応答変数のゼロと1以上を連続的に扱う

結果と課題

結果

評価用のプロジェクトデータで検証したところ，以下の結果だった．

- バグがあるかないかの判定（F値の高さで評価）
  - ✓ ゼロ過剰モデルが最も性能が良い
- バグ数の精度（残差平方和の小ささで評価）
  - ✓ ハードルモデル，ゼロ過剰モデルが高精度

評価データでは約3/4のファイルにバグが含まれている．「バグはソースコードに偏在する」ため，ゼロが多いケースのモデルを利用することで高い精度が得られることが分かった．

課題

今回の取組で統計的モデルの実用性，精度の良い手法がわかった．より手軽に利用するため，「手順の自動化」や，「開発プロセスへの組込」が今後の課題である．