On Distribution Shift in Learning-based Bug Detectors

Jingxuan He, Luca Beurer-Kellner, Martin Vechev

ICML 2022

SRILAB

ETH zürich

Bug Distributions: Real v.s. Synthetic

Real bug

```
sum = 0
for i in len(A):
  for j in len(A[i]):
    sum += A[i][i]
```

v.s.

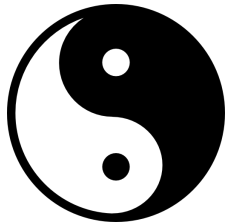
Synthetic bug

```
sum = 0
for i in len(A):
  for j in len(A[i]):
    sum += A[i][sum]
```

Data Imbalance

v.s.

Balanced Dataset



Only possible to obtain a small amount of real bugs

v.s.

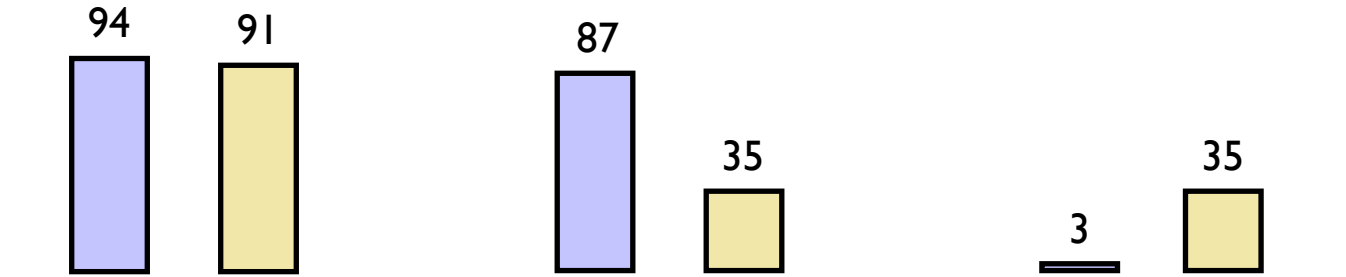
Easy to create a large amount of synthetic bugs

Existing ML-based Bug Detectors

Models: GNN, BERT, etc.

Training: Synthetic Bug Distribution

Testing: Precision Recall

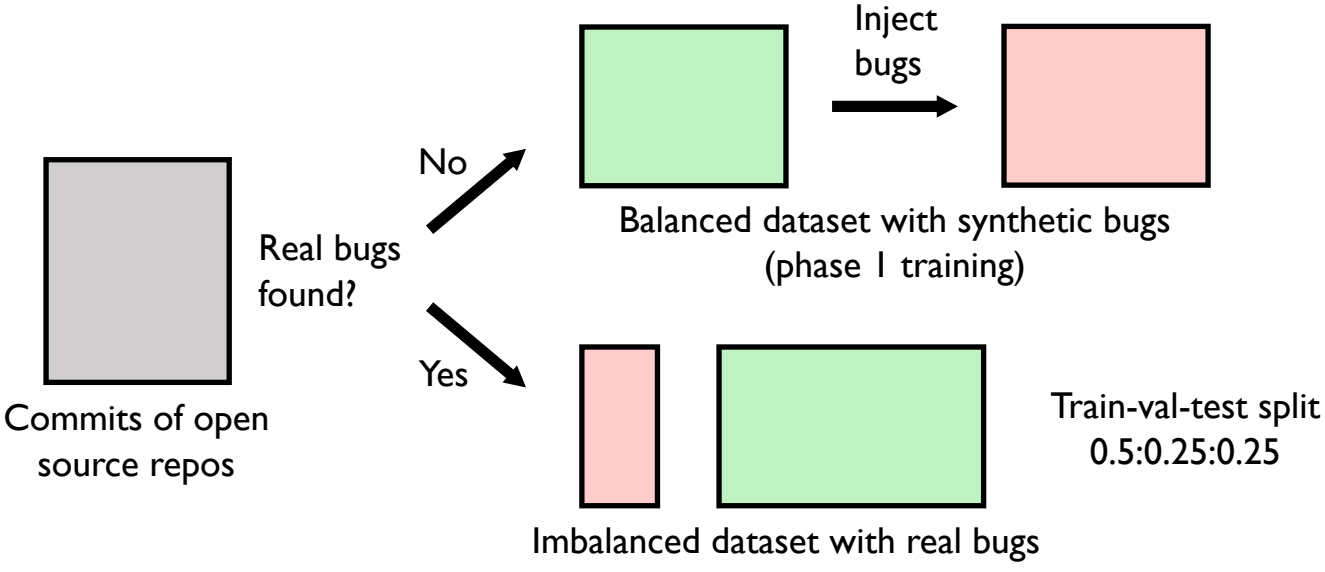


Our Solution: Two-phase Training

Phaes I: Pre-training on synthetic bug distribution

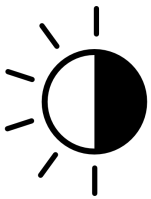
Phaes II: Fine-tuning on real bug distribution

Dataset construction:



Other Useful Techniques

A task hierarchy for classification, localization, and repair



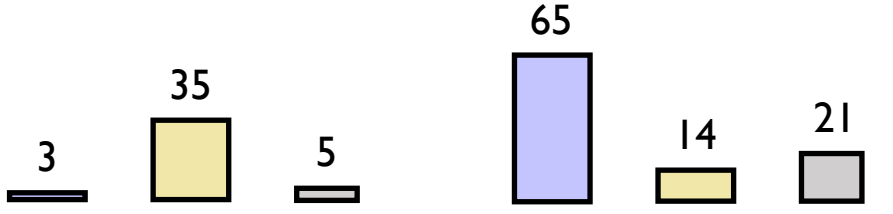
Contrastive loss for differentiating buggy/non-bgguy pairs

$-(1-p_y)\log p_y$ Focal loss for handling data imbalance

Evaluation (var-misuse bugs)

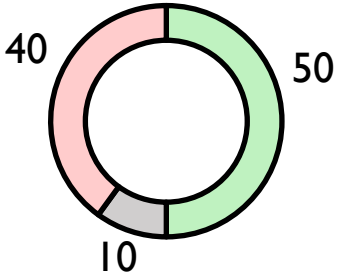
Results on the test set:

- Precision
- Recall
- Average precision



Scanning latest open source repos:

- Bugs
- Quality issues
- False positives



Precison matches test set!

Bug reports confirmed by

NumPy, Cython, pillow, CuPy, Apache Airflow, …

https://www.sri.inf.ethz.ch/publications/he22ds