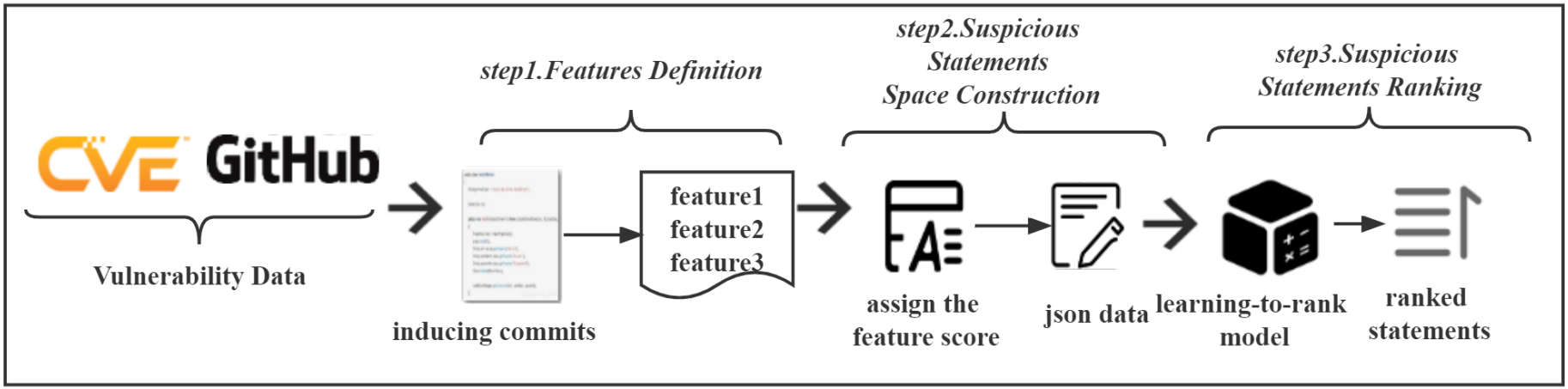
VulLoc: Vulnerability Localization Based on Inducing Commits and Fixing Commits

Lili BO, Yue LI, Xiaobing SUN , Xiaoxue WU, Bin LI

Frontiers of Computer Science, DOI: 10.1007/s11704-022-1729-x

Problems & Ideas

- Problems of conventional vulnerability localization:
  - Vulnerability localization mostly uses bug localization, and mostly uses dynamic information during program runtime as input.
- Ideas: Based on exploring the relationship between vulnerability-inducing commit and vulnerability-fixing commit, use machine learning technology to complete vulnerability localization.



The framework of the approach. VulLoc includes three phases: (1) Feature Definition. (2) Suspicious Statements Space Construction.

Main Contributions

- Contributions:
  - Created a vulnerability-inducing commit and vulnerability-fixing commit dataset, and explored the correlation between vulnerability-inducing commit and vulnerability-fixing commit.
  - Due to relatively little information about the vulnerability. The approach utilizes its correlation to alleviate the problem of using dynamic information on program operation in vulnerability localization.

Comparison results of effectiveness between VulLoc and BugLocator

Tec/Metrics	VulLoc	BugLocator
$E_{inspect}$@1	36.6%	8.3%
$E_{inspect}$@3	45%	15.3%
$E_{inspect}$@5	46%	22.2%
$E_{inspect}$@10	53%	33.3%
MAP	41.7%	14.13%
MRR	46.1%	14.0%

Comparison results of efficiency between VulLoc and BugLocator

Runtime	VulLoc	BugLocator
Time1	0.165(↑1112%)	2s
Time2	0.158s(↑659%)	1.2s
Time3	0.156s(↑1246%)	2.1s
Time4	0.149(↑705%)	1.2s
Time5	0.148s(↑2197%)	3.4s
Time6	0.156s(↑1438%)	2.4s
Time7	0.165s(↑809%)	1.5s
Time8	0.228s(↑1075%)	3.1s
Time9	0.249s(↑984%)	2.7s
Time10	0.2s (↑650%)	1.5s
Average	0.1774s(↑1089%)	2.11s

The experimental results show that VulLoc is significantly better than BugLocator in both localization effect and efficiency.