| | | |
|---|---|---|
| 《Combining Spectrum-Based Fault Localization and Statistical Debugging- An Empirical Study》 | 2019 | ASE |
| 《An Empirical Study of Boosting Spectrum-based Fault Localization via PageRank》 | 2019 | TSE |
| 《Deep Transfer Bug Localization》 | 2019 | TSE |
| 《Boosting coverage-based fault localization via graph-based representation learning》 | 2021 | FSE/ESEC |
| 《Combining Query Reduction and Expansion for Text-Retrieval-Based Bug Localization》 | 2021 | SANER |

# Combining Spectrum-Based Fault Localization and Statistical Debugging: An Empirical Study

作者：Jiajun Jiang, Ran Wang, Yingfei Xiong

汇报人：陈冰婷　　导师：邹卫琴

# 目录

Contents

# SBFL and SD

$$Ochiai(e) = \frac{failed(e)}{\sqrt{totalfailed \cdot (failed(e) + passed(e))}}$$

$$Importance(p) = \frac{2}{\frac{1}{Increase(p)} + \frac{1}{Sensitive(p)}}$$

Different spectrum-based fault localization approaches follow the same paradigm but use different formulas to compute the suspicious scores.

- branches -i.e., *if* statements
- returns - i.e., $>, <, \geq, \leq, ==$ and $=$.
- scalar-pairs

$$Increase(p) = \frac{F(p)}{S(p) + F(p)} - \frac{F_o(p)}{S_o(p) + F_o(p)}$$

$$Sensitive(p) = \frac{\log(F(p))}{\log(totalfailed)}$$

# Unified Model

$$c(s, e, r) = max_{p \in s(e)} r(p)$$

$$UNI^{s,r,g,c}(E) = \{(e, max_{e_i \in g(e)} c(s, e_i, r)) \mid e \in E\}$$

# Four Variation Points

01

## Predicates

Which kinds of predicates are most important?

03

## Granularity of Data Collection

How does the granularity of data collection impactthe fault localization result?

02

## Risk Evaluation Formulas

How does the risk evaluation formula impact theeffectiveness of fault localization?

04

## Methods for Combining Suspicious Scores

How does combining method among different predicates impact the effectiveness of fault localization?

# Experiments

Benchmark: Defect4j

TABLE I: Details of the experiment benchmark.

| Project | #Bugs | #KLoC | #Tests |
|---|---|---|---|
| JFree**Chart** | 26 | 96 | 2,205 |
| Apache commons-**Math** | 106 | 85 | 3,602 |
| Apache commons-**Lang** | 65 | 22 | 2,245 |
| Joda-**Time** | 27 | 28 | 4,130 |
| **Closure** compiler | 133 | 90 | 7,927 |
| Total | 357 | 321 | 20,109 |

Evaluation Metrics

- **Recall of Top-k**
- **EXAM Score**

Frameword https://github.com/xgdsmileboy/StateCoverLocator

# Predicates

Branches

Returns

Scalar-Pairs

SBFL

# Predicates

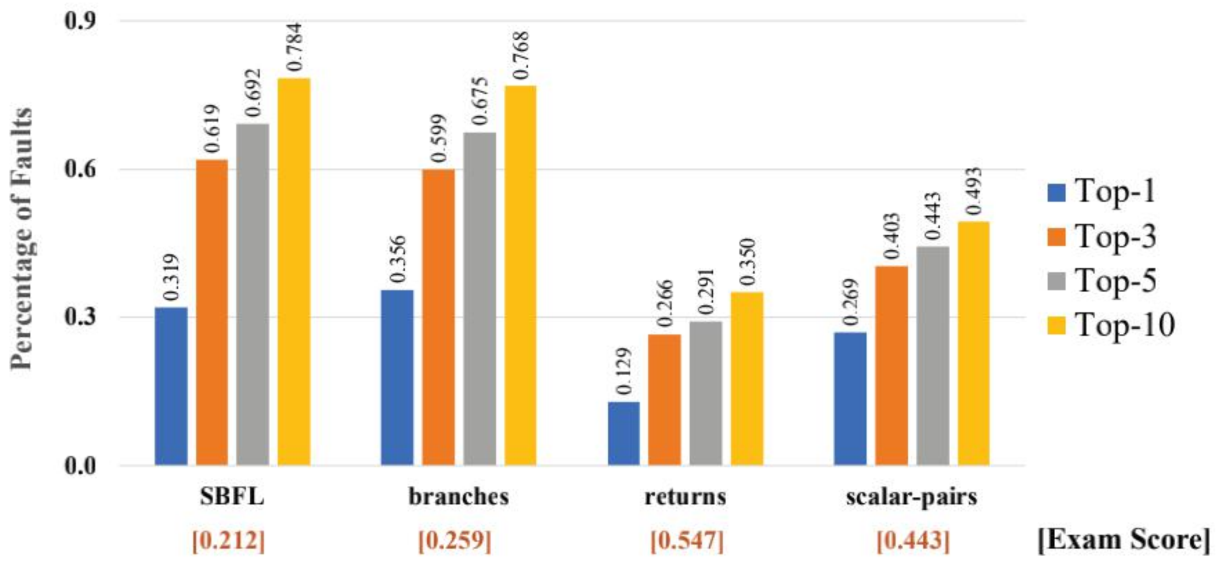



Fig. 1: Fault localization results when only employing individual group of predicates.

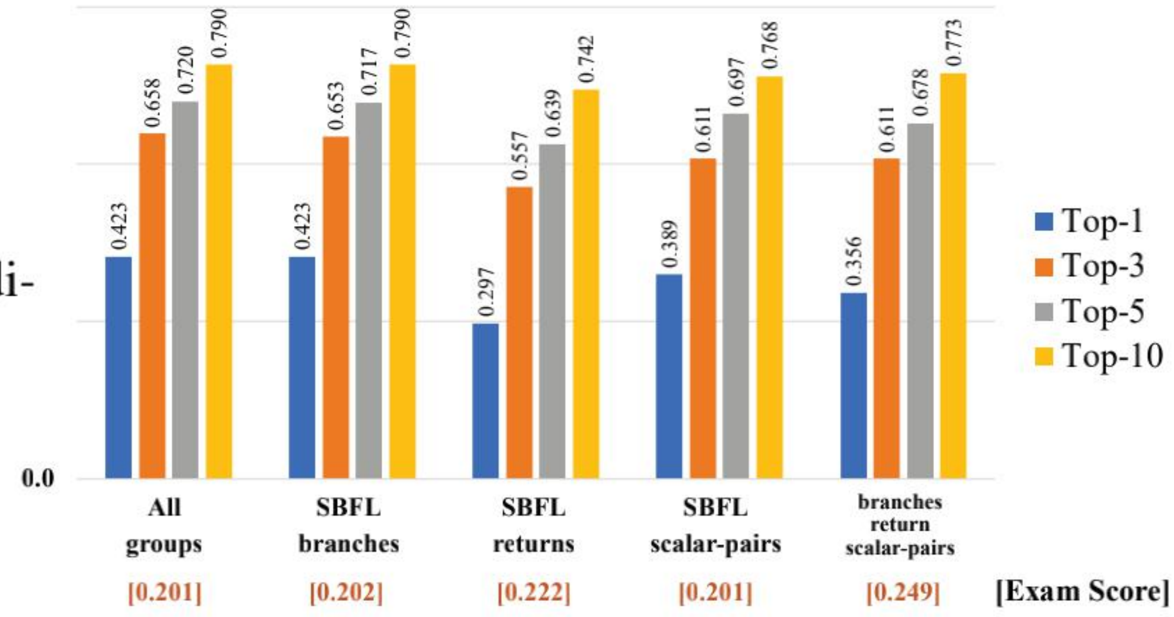



Fig. 2: Fault localization results when considering the combination among different groups of predicates.

# Risk Evaluation Formulas

TABLE II: Formulas employed in the experiment.

| Name | Formula |
|---|---|
| **Ochiai [24]** | $r(p) = \frac{failed(p)}{\sqrt{totalfailed \cdot (failed(p)+passed(p))}}$ |
| **Tarantula [2]** | $r(p) = \frac{failed(p)/totalfailed}{failed(p)/totalfailed+passed(p)/totalpassed}$ |
| **Barinel [35]** | $r(p) = 1-\frac{passed(p)}{passed(p)+failed(p)}$ |
| **DStar† [36]** | $r(p) = \frac{failed(p)^{*}}{passed(p)+(totalfailed-failed(p))}$ |
| **Op2 [37]** | $r(p) = failed(p)-\frac{passed(p)}{totalpassed+1}$ |
| **NewSD‡** | $r(p) = \frac{2}{1/Increase(p)+\log(totalfailed)/\log(F(p)+1)}$ |

# Risk Evaluation Formulas

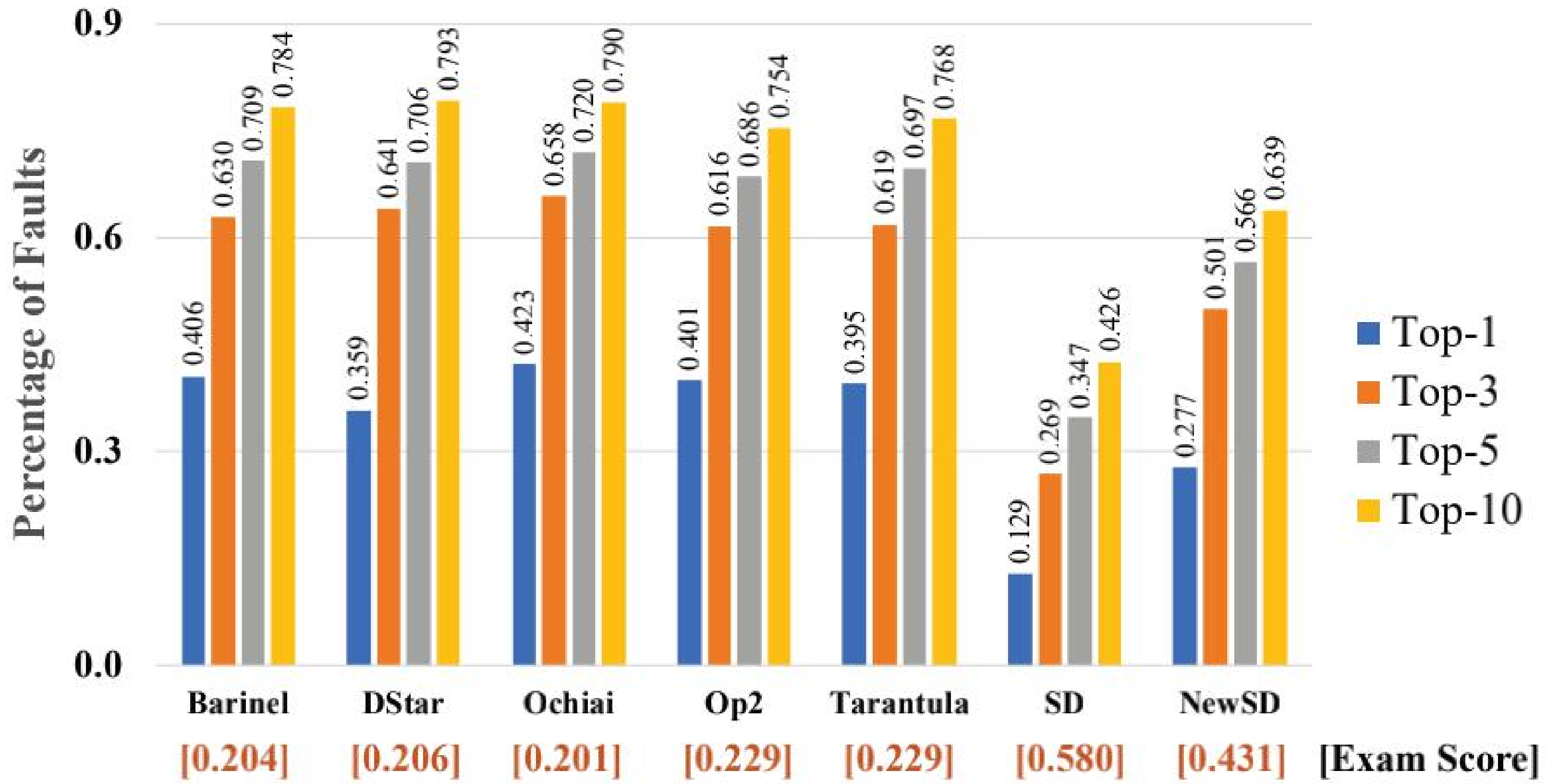



Fig. 3: Fault localization results when using different risk evaluation formulas.

# Granularity of Data Collection

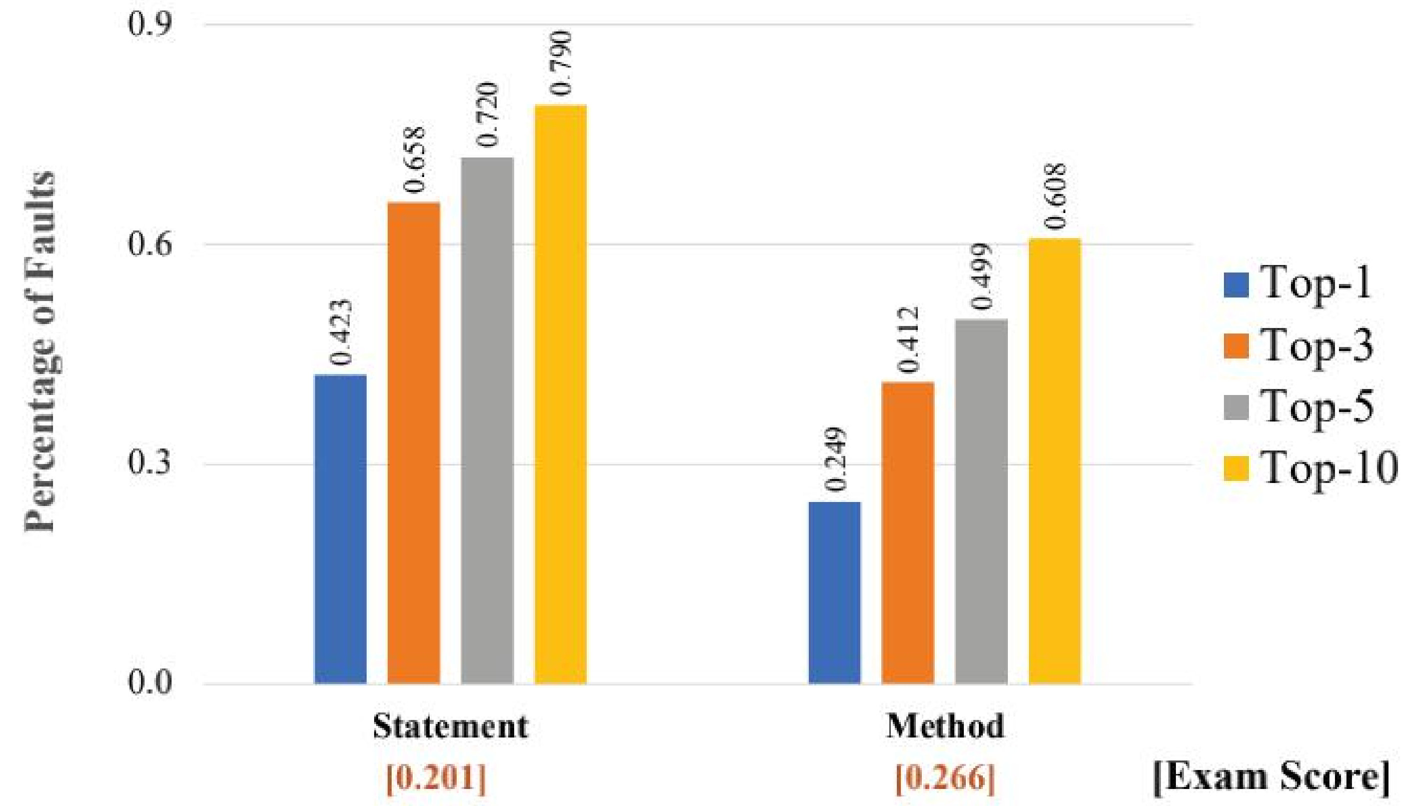
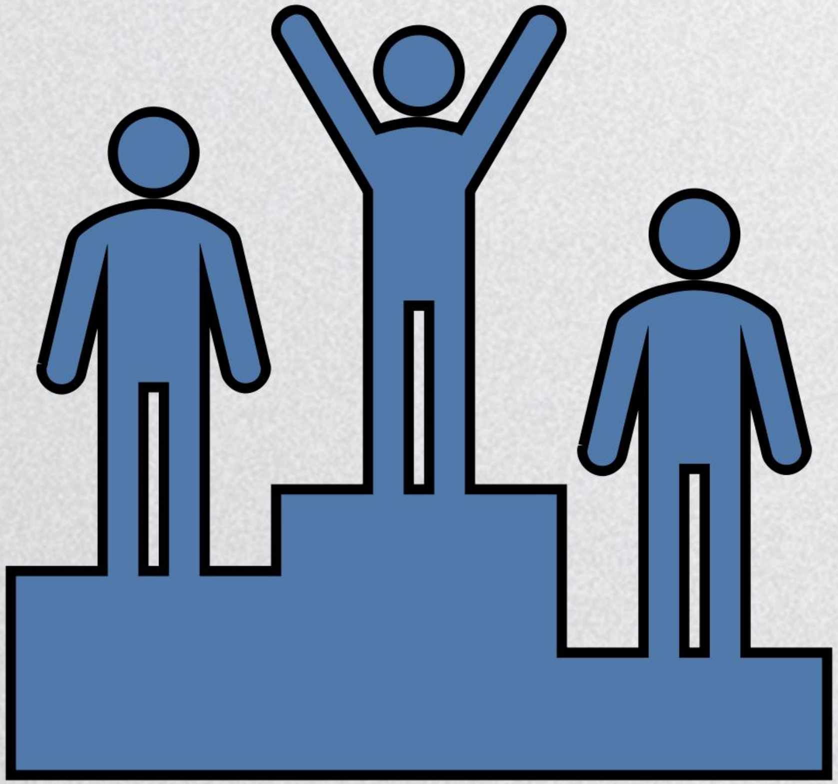



Fig. 4: Comparison of fault localization results under statement and method level data collection.

# Methods for Combining Suspicious Scores



## MaxPred

Given a program elemente, we compute its suspicious score as the maximum score of all predicates related to it, i.e.,c(s, e, r)=maxp ∈ s(e)r(p).

## LinPred

Given a program elemente, we partition the predicates related to it into two stand alone sets: P1 and P2, where P1 ∪ P2=s(e) and P1 contains one predicate from SBFL while the others constitute P2. Then, the combining method is defined as

$c(s, e, r) = (1-\alpha) \cdot max_{p \in P1} r(p) + \alpha \cdot max_{p \in P2} r(p)$, where $\alpha \in [0,1.0]$

# Methods for Combining Suspicious Scores

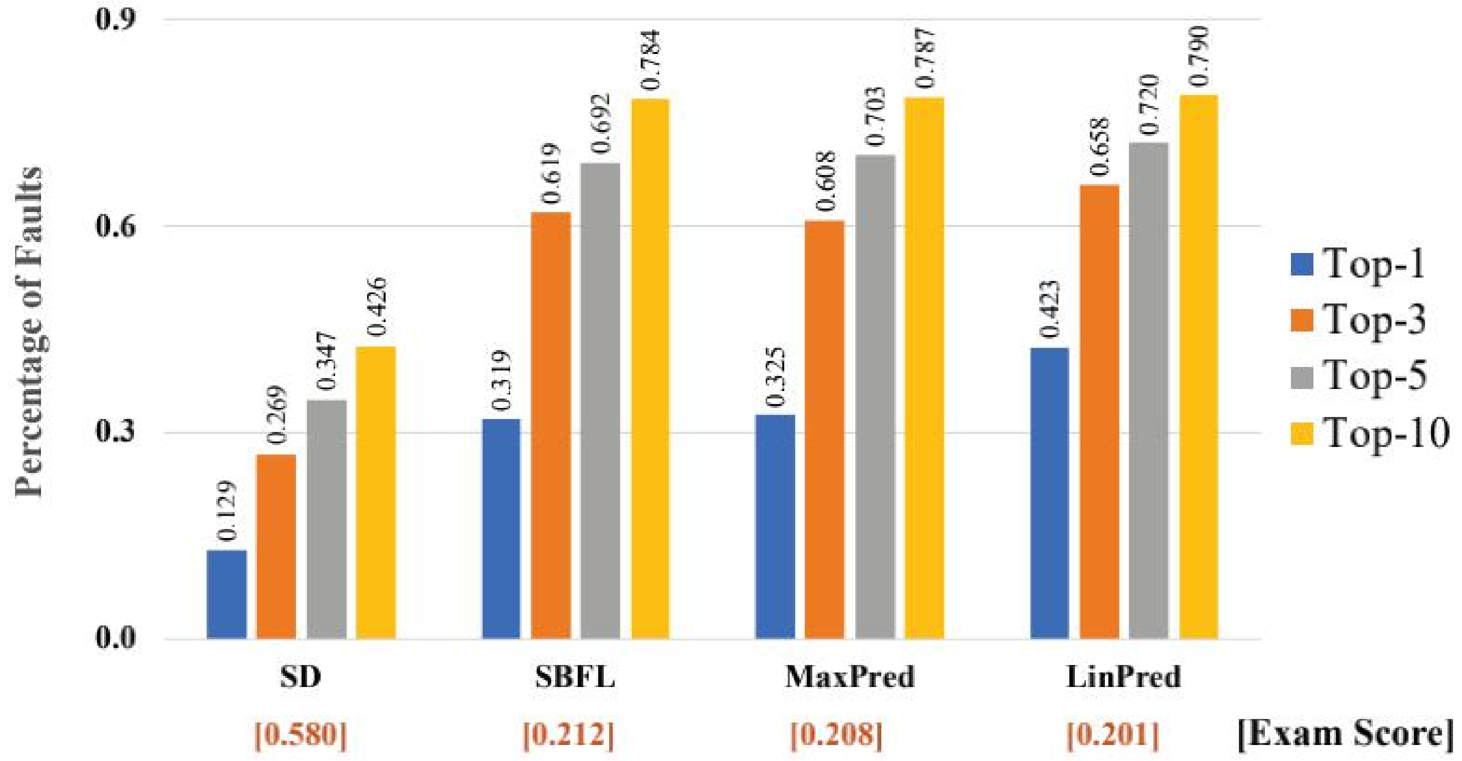



Fig. 5: Result comparison among traditional SBFL and SD approaches with the combined methods.

# Conclusion

- Among all predicates, those from existing conditions contribute most to the Top-1 fault localization accuracy;
- Fine-grained datacollection contributes more effective fault localization withlittle more execution overhead
- A linear combination ofsuspicious scores from SBFL and SD predicates leads to thebest result.

THANK YOU FOR YOUR LISTENING.

# 谢谢您的聆听