New method to predict patterns' importances

Aibolit team

June 4, 2020

Previous experiment

According to last experiments patterns 'Non final class', 'Non final attribute', 'Null check' and 'Var in the middle' are most frequent and important.

In previous experiment we trained model without this patterns and got best distribution of patterns' importances. Our model predict pattern which has the greatest impact on the complexity. On the graphics there is a distribution of most important patterns.

Results of previous experiment are on the table (how to change quality of the model after removing patterns).
p1 - remove pattern 'Non final class'
p2 - remove pattern 'Non final attribute'
p3 - remove pattern 'Null check'
p4 - remove pattern 'Var in the middle'

Patterns	mse	mae	r2
no removing	0.0182	0.0922	0.4792
p1	0.0181	0.0922	0.4811
p2	0.0195	0.0993	0.4395
p3	0.0209	0.1	0.4017
p4	0.0183	0.0932	0.4763
p1, p2	0.0196	0.0993	0.4366
p1, p3	0.0207	0.1	0.4052
p1, p4	0.0185	0.0935	0.4698
p2, p3	0.0228	0.1085	0.3453
p2, p4	0.02	0.1009	0.4259
p3, p4	0.0213	0.1016	0.3892
p1, p2, p3	0.0228	0.1087	0.3467
p1, p2, p4	0.02	0.1013	0.4264
p1, p3, p4	0.0214	0.1018	0.3874
p2, p3, p4	0.0235	0.1114	0.327
p1, p2, p3, p4	0.0237	0.1125	0.32

Table 1: Results of previous experiment

New method to predict patterns' importances

We suggested new method to predict patterns' importances. Model is trained on dataset.
To predict patterns' importances need to do following acts:
1. Each pattern (in rotation) decreased (increased) by 1 and predict complexity for modified snippet.

Calculate pattern importance as minimum of 3 complexity for non-modified snippet, increased by 1 snippet, decreased by 1 snippet.
2. sorted importances and got necessary ranked array.

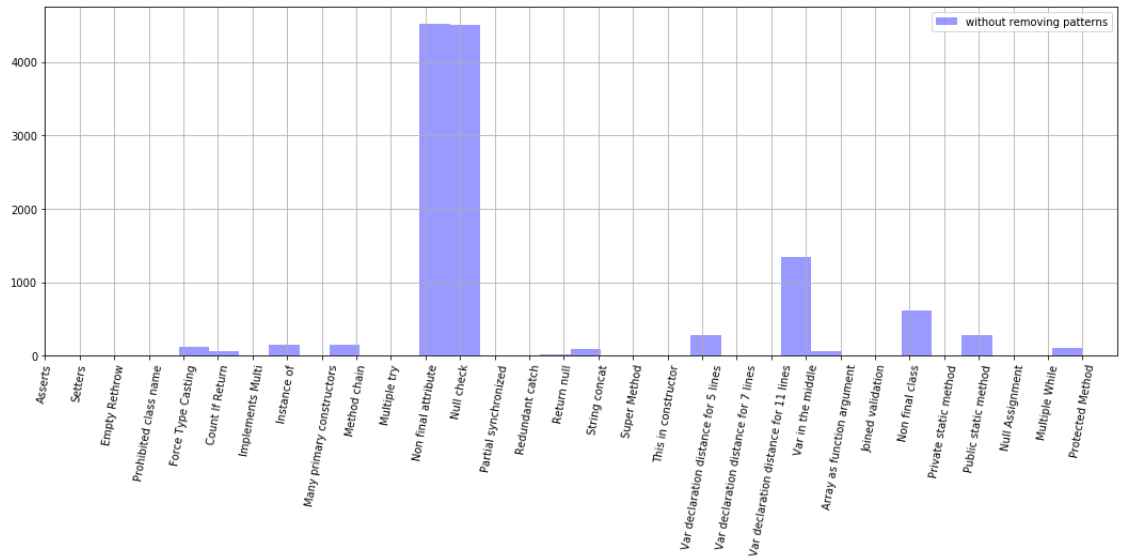

Figure 1: Removing all 4 patterns - old method

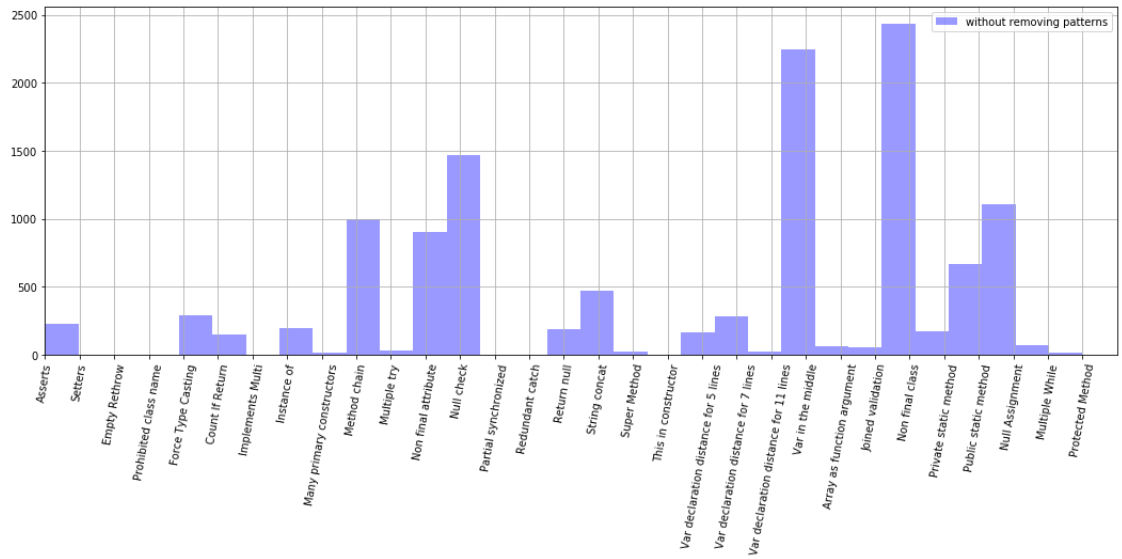

Figure 2: Removing all 4 patterns - new method

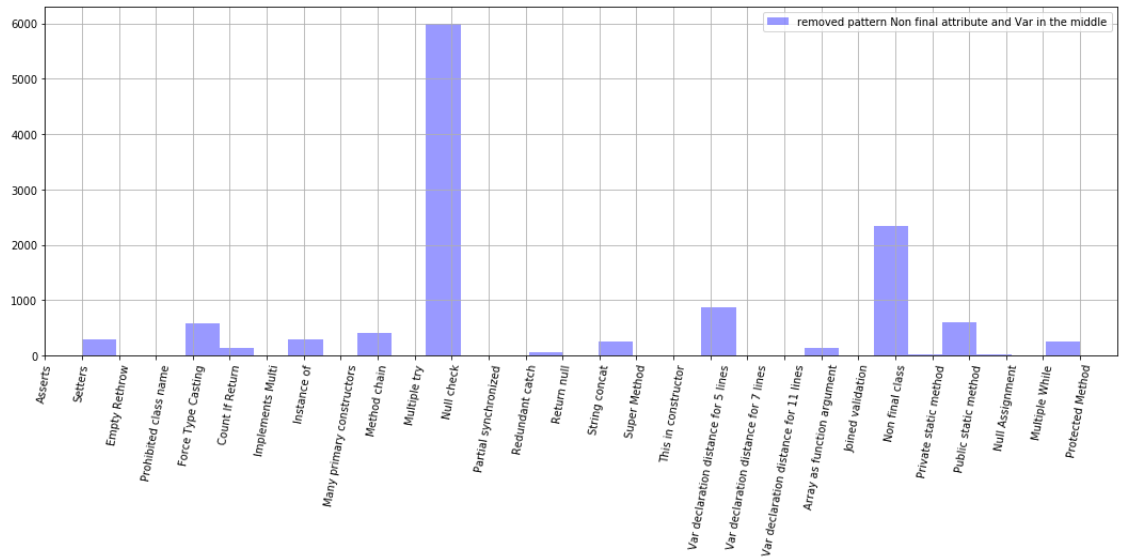

Figure 3: Removing patterns 'Non final attribute' and 'Var in the middle' - old method

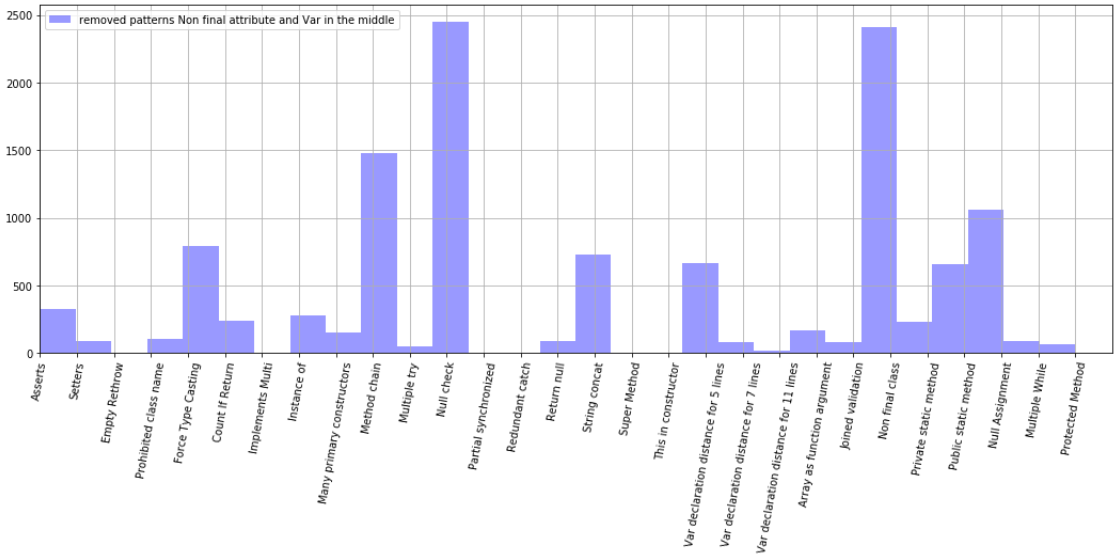

Figure 4: Removing patterns 'Non final attribute' and 'Var in the middle' - new method

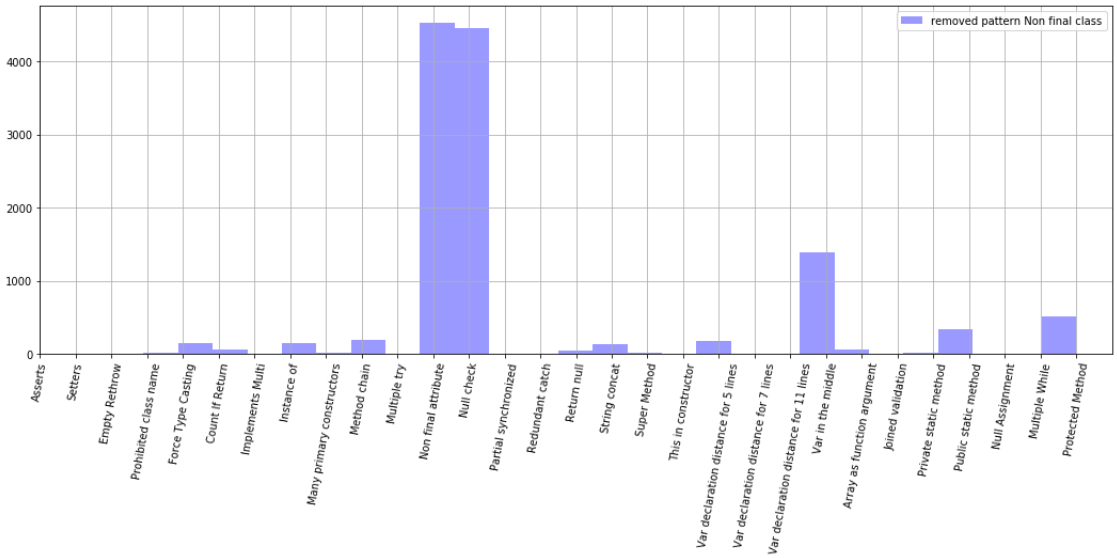

Figure 5: Removing pattern 'Non final class' - old method

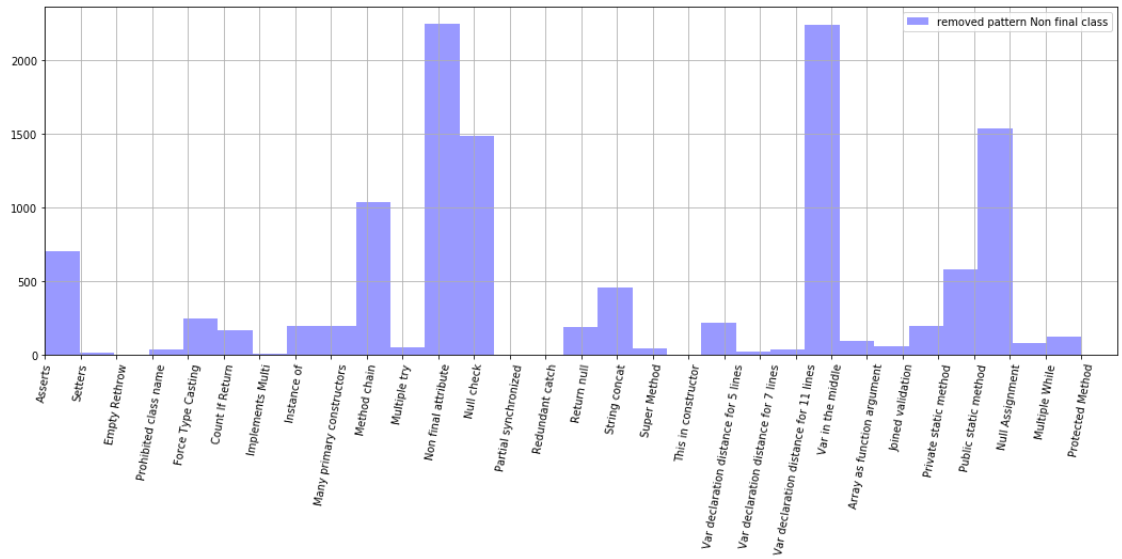

Figure 6: Removing pattern 'Non final class' - new method

Comparison of two methods

According to the graphics in case of new method distribution of patterns' importances got more balanced. For each combination of removing patterns (from table 1) kl-divergence between distributions of patterns' importances from new and old methods was calculated. Results are on the graphic below.

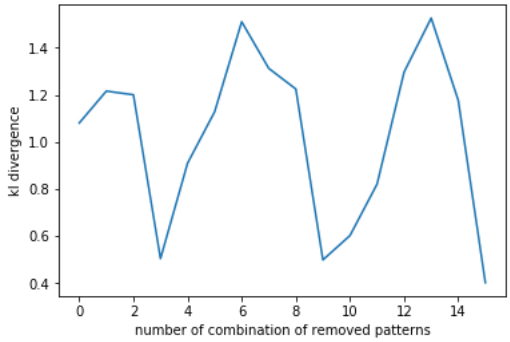

Figure 7: kl-divergence

According to the graphic 7 there are significant differences between distributions.

In case of new method next situation is probable: all changes doesn't decrease complexity. It's undefined behaviour of this method. So need to choose combinations of removing patterns where this mistake are not significant.

Conclusion

In case of new method distribution of patterns' importances get more balanced. Better to use new method without removing patterns because then correlation between quality of prediction complexity, quality of distribution and count of described above mistakes is optimal.