Appendix for "Unsupervised Software Defect Prediction through Multi-view Clustering"

Zhiqiang Li, Hongyu Zhang, Xiao-Yuan Jing, Jingwen Niu, Wangyang Yu, Yueyue Liu

Abstract—Software defect prediction (SDP) aims at identifying more likely defect-prone modules to prioritize software quality assurance activities with low inspection effort. Up to now, there are a large number of supervised defect prediction methods that are extensively studied. However, these methods require the need for labeling data to get enough training modules, which will cause a lot of waste of human resources. Cross-project defect prediction has been the main solution by reusing prediction models from other projects that have enough historical data. Unfortunately, there often exists large distribution differences between different projects and privacy concerns of data. Unsupervised learning technique is an alternative solution to the unlabeled defect data, but it mainly focuses on single-view prediction by concatenating all the software metrics. This ignores the diversity and complementarity of different types of metrics. In this paper, we propose a novel approach, namely multi-view unsupervised software defect prediction (MUSDP), which addresses the above limitations. MUSDP aims to collaboratively learn the diversity and complementarity of different views to build a robust and reliable defect prediction model. Extensive experiments on 28 releases across 8 open-source software projects indicate that MUSDP achieves better or comparable *AUC*, *MCC*, P_{opt}, and *CostEffort@20%* results against a range of competing methods. The interpretation of the predictions of MUSDP is largely influenced by the number of deleted lines in a module. We suggest that future work can use it to conduct unsupervised defect prediction, especially for the projects with limited labeled data.

1 Software Projects and Metrics

In this paper, we conduct defect prediction on 28 releases across 8 open-source software projects from the Apache community [1]. Each project consists of 65 software metrics along 3 dimensions, i.e., 54 code metrics, 5 process metrics, and 6 ownership metrics. Among them, code metrics (e.g., LOC and cyclomatic complexity) describe the relationship between code properties and software quality, process metrics (e.g., the number of added lines and deleted lines) describe the relationship between development process and software quality, and ownership metrics (e.g., the number of own commits and own lines) describe the relationship between the ownership of modules and software quality. We refer each metric dimension as one view for multi-view clustering and there are 3 views in total. Tables 1 and 2 separately show the statistical summary of the studied software projects and metrics.

TABLE 1
An overview of the used software projects

Project	#Metrics	#Files	#Defective files	%Defective files
activemq-5.0.0	65	1884	293	15.55%
activemq-5.1.0	65	1970	154	7.82%
activemq-5.2.0	65	2040	219	10.74%
activemq-5.3.0	65	2367	258	10.90%
activemq-5.8.0	65	3420	206	6.02%
derby-10.2.1.6	65	1963	661	33.67%
derby-10.3.1.4	65	2206	669	30.33%
derby-10.5.1.1	65	2705	383	14.16%
groovy-1_5_7	65	757	26	3.43%
groovy-1_6beta1	65	821	70	8.53%
groovy-1_6beta2	65	884	76	8.60%
hbase-0.94.0	65	1059	218	20.59%
hbase-0.95.0	65	1669	383	22.95%
hbase-0.95.2	65	1834	483	26.34%
hive-0.10.0	65	1560	176	11.28%
hive-0.12.0	65	2662	213	8.00%
hive-0.9.0	65	1416	283	19.99%
jruby-1.1	65	731	87	11.90%
jruby-1.4.0	65	978	180	18.40%
jruby-1.5.0	65	1131	82	7.25%
jruby-1.7.0	65	1614	87	5.39%
lucene-2.3.0	65	805	196	24.35%
lucene-2.9.0	65	1368	273	19.96%
lucene-3.0.0	65	1337	155	11.59%
lucene-3.1	65	2806	107	3.81%
wicket-1.3.0beta2	65	1763	130	7.37%
wicket-1.3.0beta1	65	1672	101	6.04%
wicket-1.5.3	65	2578	105	4.07%

2 Metrics After Correlation and Redundancy Analysis

Prior work [2] points out that the correlated or redundant software metrics impact the performance and interpretation of defect models. Therefore, we conduct correlation and redundancy analyses prior to building our MUSDP models. Similar to the previous defect prediction studies [3], [4], [5], we use the *AutoSpearman* algorithm [6] to remove the correlated and redundant metrics for data pre-processing. The rest of software metrics after performing the *AutoSpearman* algorithm are shown in Table 3. From the table,

- *Zhiqiang Li, Jingwen Niu, and Wangyang Yu are with the School of Computer Science, Shaanxi Normal University, Xi'an 710119, China. E-mail: lizq@snnu.edu.cn, niujw66@163.com, and ywy191@snnu.edu.cn.*
- *Hongyu Zhang is the School of Big Data and Software Engineering, Chongqing University, Chongqing 401331, China. E-mail: hyzhang@cqu.edu.cn.*
- *Xiao-Yuan Jing is with the School of Computer Science, Wuhan University, China, and the School of Computer, Guangdong University of Petrochemical Technology, China. E-mail: jingxy_2000@126.com.*
- *Yueyue Liu is with the School of Information and Physical Sciences, The University of Newcastle, Callaghan, NSW 2308, Australia. E-mail: yueyue.liu@uon.edu.au.*

we can observe that more than half of the software metrics have been removed for each project. Subsequently, we use these metrics to preform unsupervised defect prediction experiments for MUSDP.

3 Comparison Results with Baselines

In this section, we report the detailed experimental results for MUSDP and the competing baseline methods. Tables 4, 5, 6, and 7 show the median *AUC*, *MCC*, P_{opt}, and *CostEffort@20%* results on each project for each method with 100 random runs, respectively. The overall median results (i.e., median of those $28 \times 100 = 2800$ values) and the average rank (AR) obtained from the non-parametric Scott-Knott effect size difference (NPSKESD) [4] test are also reported in the last two rows of the table (denoted as "*Median*" and "*AR*"). The lower the *AR* value, the better the model performance. In these tables, the best values in the last two rows are in bold font.

From these tables, we can observe that MUSDP, especially MUSDP_v2, performs better prediction performance in terms of *AUC*, *MCC*, P_{opt}, and *CostEffort@20%* with statistical significance compared to the state-of-the-art UDP methods. These results demonstrate the effectiveness and utility of MUSDP in enhancing the accuracy of unsupervised defect prediction.

4 Comparison results of MUSDP with different types of software metrics

In this section, we conduct a comparative analysis of defect prediction models built using MUSDP and its three variants (i.e., MUSDP_CP, MUSDP_CO, and MUSDP_PO). That is, MUSDP only uses code metrics and process metrics, code metrics and ownership metrics, process metrics and ownership metrics, respectively. Tables 8, 9, 10, and 11 separately show the median *AUC*, *MCC*, P_{opt}, and *CostEffort@20%* results on each project for each method with 100 repetitions.

From these tables, we can observe that the prediction performance of MUSDP varies depending on the specific combination of different types of software metrics compared to MUSDP and its three variants. Hence, the impact of different types of software metrics should be carefully considered when using MUSDP for UDP.

References

[1] S. Yatish, J. Jiarpakdee, P. Thongtanunam, and C. Tantithamthavorn, "Mining software defects: Should we consider affected releases?" in *ICSE'19*, 2019, pp. 654–665.

[2] J. Jiarpakdee, C. Tantithamthavorn, and A. Hassan, "The impact of correlated metrics on the interpretation of defect models," *IEEE Transactions on Software Engineering*, vol. 47, no. 2, pp. 320–331, 2021.

[3] J. Jiarpakdee, C. Tantithamthavorn, H. K. Dam, and J. Grundy, "An empirical study of model-agnostic techniques for defect prediction models," *IEEE Transactions on Software Engineering*, vol. 48, no. 1, pp. 166–185, 2022.

[4] C. Tantithamthavorn, S. McIntosh, A. E. Hassan, and K. Matsumoto, "The impact of automated parameter optimization on defect prediction models," *IEEE Transactions on Software Engineering*, vol. 45, no. 7, pp. 683–711, 2019.

[5] C. Tantithamthavorn, A. E. Hassan, and K. Matsumoto, "The impact of class rebalancing techniques on the performance and interpretation of defect prediction models," *IEEE Transactions on Software Engineering*, vol. 46, no. 11, pp. 1200–1219, 2020.

[6] J. Jiarpakdee, C. Tantithamthavorn, and C. Treude, "Autospearman: Automatically mitigating correlated software metrics for interpreting defect models," in *2018 IEEE International Conference on Software Maintenance and Evolution (ICSME)*. IEEE, 2018, pp. 92–103.

TABLE 2
An overview of the used software metrics

Metric type	Metric name	Description
Code metrics (54)	CountDeclMethodPrivate	the number of local (not inherited) private methods
	AvgLineCode	the average number of lines containing source code for all nested functions or methods
	CountLine	the number of physical lines
	MaxCyclomatic	the maximum cyclomatic complexity of all nested functions or methods
	CountDeclMethodDefault	the number of local default methods
	AvgEssential	the average essential complexity for all nested functions or methods
	CountDeclClassVariable	the number of class variables
	SumCyclomaticStrict	the sum of strict cyclomatic complexity of all nested functions or methods
	AvgCyclomatic	the average cyclomatic complexity for all nested functions or methods
	AvgLine	the average number of lines for all nested functions or methods
	CountDeclClassMethod	the number of class methods
	AvgLineComment	the average number of lines containing comment for all nested functions or methods
	AvgCyclomaticModified	the modified McCabe Cyclomatic complexity
	CountDeclFunction	the number of functions
	CountLineComment	the number of lines containing comment
	CountDeclClass	the number of classes
	CountDeclMethod	the number of local (not inherited) methods
	SumCyclomaticModified	the sum of modified cyclomatic complexity of all nested functions or methods
	CountLineCodeDecl	the number of lines containing declarative source code
	CountDeclMethodProtected	the number of local protected methods
	CountDeclInstanceVariable	the number of instance variables
	MaxCyclomaticStrict	the maximum strict cyclomatic complexity of nested functions or methods
	CountDeclMethodPublic	the number of local (not inherited) public methods
	CountLineCodeExe	the number of lines containing executable source code
	SumCyclomatic	the sum of cyclomatic complexity of all nested functions or methods
	SumEssential	the sum of essential complexity of all nested functions or methods
	CountStmtDecl	the number of declarative statements
	CountLineCode	the number of lines containing source code
	CountStmtExe	the number of executable statements
	RatioCommentToCode	the ratio of comment lines to code lines
	CountLineBlank	the number of blank lines
	CountStmt	the number of statements
	MaxCyclomaticModified	the maximum modified cyclomatic complexity of nested functions or methods
	CountSemicolon	the number of semicolons
	AvgLineBlank	the average number of blank lines for all nested functions or methods
	CountDeclInstanceMethod	the number of instance methods
	AvgCyclomaticStrict	the average strict cyclomatic complexity for all nested functions or methods
	PercentLackOfCohesion	the 100% minus the average cohesion for package entities
	MaxInheritanceTree	the maximum depth of class in inheritance tree
	CountClassDerived	the number of immediate subclasses
	CountClassCoupled	the number of other classes to which a class is coupled
	CountClassBase	the number of immediate base classes
	CountInput (max, min, mean)	the number of calling subprograms plus global variables read
	CountOutput (max, min, mean)	the number of called subprograms plus global variables set
	CountPath (max, min, mean)	the number of unique paths trhough a body of code
	MaxNesting (max, min, mean)	the maximum nesting level of control constructs
Process metrics (5)	COMM	the number of Git commits
	ADEV	the number of active developers
	DDEV	the number of distinct developers
	Added_lines	the normalized number of lines added to the module
	Del_lines	the number normalized of lines deleted to the module
Ownership metircs (6)	OWN_LINE	the proportion of lines of code written by the developer who has the highest contribution of lines of code on the module
	OWN_COMMIT	the proportion of code changes (i.e., Git commits) made by the developer who has the highest contribution of code changes on the module
	MINOR_COMMIT	the number of unique developers who have contributed less than 5% of the total code changes (i.e., Git commits) on the module
	MINOR_LINE	the number of unique developers who have contributed less than 5% of the total lines of code on the module
	MAJOR_COMMIT	the number of unique developers who have contributed more than 5% of the total code changes (i.e., Git commits) on the module
	MAJOR_LINE	the number of unique developers who have contributed more than 5% of the total lines of code on the module

TABLE 3
An overview of the selected metrics used by AutoSpearman

Project	Metrics
activemq-5.0.0	AvgCyclomaticModified, AvgEssential, AvgLineBlank, AvgLineComment, CountClassBase, CountClassCoupled, CountClassDerived, CountDeclClass, CountDeclClassMethod, CountDeclClassVariable, CountDeclInstanceVariable, CountDeclMethodDefault, CountDeclMethodPrivate, CountDeclMethodProtected, CountDeclMethodPublic, CountInput_Mean, CountInput_Min, CountOutput_Min, MaxInheritanceTree, MaxNesting_Min, PercentLackOfCohesion, RatioCommentToCode, Added_lines, COMM, DDEV, MAJOR_LINE, MINOR_COMMIT, MINOR_LINE, OWN_COMMIT
activemq-5.1.0	AvgCyclomaticModified, AvgEssential, AvgLineBlank, AvgLineComment, CountClassBase, CountClassCoupled, CountClassDerived, CountDeclClass, CountDeclClassMethod, CountDeclClassVariable, CountDeclInstanceVariable, CountDeclMethodDefault, CountDeclMethodPrivate, CountDeclMethodProtected, CountDeclMethodPublic, CountInput_Mean, CountInput_Min, CountOutput_Min, MaxInheritanceTree, MaxNesting_Min, PercentLackOfCohesion, RatioCommentToCode, Added_lines, MAJOR_LINE, MINOR_COMMIT, MINOR_LINE, OWN_COMMIT
activemq-5.2.0	AvgCyclomaticModified, AvgEssential, AvgLineBlank, AvgLineComment, CountClassBase, CountClassCoupled, CountClassDerived, CountDeclClass, CountDeclClassMethod, CountDeclClassVariable, CountDeclInstanceVariable, CountDeclMethodDefault, CountDeclMethodPrivate, CountDeclMethodProtected, CountInput_Mean, CountInput_Min, CountLineComment, CountOutput_Min, MaxInheritanceTree, MaxNesting_Min, PercentLackOfCohesion, RatioCommentToCode, Added_lines, MAJOR_LINE, MINOR_COMMIT, MINOR_LINE, OWN_COMMIT
activemq-5.3.0	AvgCyclomaticModified, AvgEssential, AvgLineBlank, CountClassBase, CountClassCoupled, CountClassDerived, CountDeclClass, CountDeclClassMethod, CountDeclClassVariable, CountDeclInstanceVariable, CountDeclMethodDefault, CountDeclMethodPrivate, CountDeclMethodProtected, CountDeclMethodPublic, CountInput_Mean, CountInput_Min, CountLineComment, CountOutput_Mean, CountOutput_Min, MaxInheritanceTree, MaxNesting_Min, PercentLackOfCohesion, RatioCommentToCode, Added_lines, MAJOR_LINE, MINOR_COMMIT, MINOR_LINE, OWN_COMMIT
activemq-5.8.0	AvgCyclomaticModified, AvgEssential, AvgLineBlank, CountClassBase, CountClassCoupled, CountClassDerived, CountDeclClass, CountDeclClassMethod, CountDeclInstanceVariable, CountDeclMethodDefault, CountDeclMethodPrivate, CountDeclMethodProtected, CountDeclMethodPublic, CountInput_Mean, CountInput_Min, CountLineComment, CountOutput_Mean, CountOutput_Min, MaxInheritanceTree, MaxNesting_Min, PercentLackOfCohesion, RatioCommentToCode, Added_lines, DDEV, Del_lines, MAJOR_LINE, MINOR_COMMIT, MINOR_LINE, OWN_COMMIT
derby-10.2.1.6	AvgEssential, AvgLineBlank, AvgLineComment, CountClassBase, CountClassCoupled, CountClassDerived, CountDeclClass, CountDeclClassMethod, CountDeclClassVariable, CountDeclInstanceVariable, CountDeclMethodDefault, CountDeclMethodPrivate, CountDeclMethodProtected, CountDeclMethodPublic, CountInput_Mean, CountInput_Min, CountLineComment, CountOutput_Min, MaxInheritanceTree, MaxNesting_Mean, MaxNesting_Min, PercentLackOfCohesion, RatioCommentToCode, Added_lines, MAJOR_LINE, MINOR_COMMIT, OWN_COMMIT, OWN_LINE
derby-10.3.1.4	AvgEssential, AvgLineBlank, AvgLineComment, CountClassBase, CountClassCoupled, CountClassDerived, CountDeclClass, CountDeclClassMethod, CountDeclClassVariable, CountDeclInstanceVariable, CountDeclMethodDefault, CountDeclMethodPrivate, CountDeclMethodProtected, CountDeclMethodPublic, CountInput_Mean, CountInput_Min, CountLineComment, CountOutput_Min, CountPath_Min, MaxInheritanceTree, MaxNesting_Mean, PercentLackOfCohesion, RatioCommentToCode, Added_lines, MAJOR_COMMIT, MAJOR_LINE, MINOR_COMMIT, OWN_COMMIT, OWN_LINE
derby-10.5.1.1	AvgEssential, AvgLineBlank, AvgLineComment, CountClassBase, CountClassCoupled, CountClassDerived, CountDeclClass, CountDeclClassMethod, CountDeclClassVariable, CountDeclInstanceVariable, CountDeclMethodDefault, CountDeclMethodPrivate, CountDeclMethodProtected, CountDeclMethodPublic, CountInput_Mean, CountInput_Min, CountLineComment, CountOutput_Min, MaxInheritanceTree, MaxNesting_Mean, MaxNesting_Min, PercentLackOfCohesion, RatioCommentToCode, Added_lines, Del_lines, MAJOR_COMMIT, MAJOR_LINE, MINOR_COMMIT, MINOR_LINE
groovy-1_5_7	AvgCyclomaticStrict, AvgEssential, AvgLineBlank, AvgLineComment, CountClassBase, CountClassCoupled, CountClassDerived, CountDeclClass, CountDeclClassMethod, CountDeclClassVariable, CountDeclInstanceMethod, CountDeclInstanceVariable, CountDeclMethodDefault, CountDeclMethodPrivate, CountDeclMethodProtected, CountInput_Mean, CountInput_Min, CountOutput_Min, MaxInheritanceTree, MaxNesting_Min, RatioCommentToCode, Added_lines, MAJOR_LINE, MINOR_COMMIT, MINOR_LINE, OWN_COMMIT
groovy-1_6beta1	AvgCyclomaticStrict, AvgEssential, AvgLineBlank, AvgLineComment, CountClassBase, CountClassCoupled, CountClassDerived, CountDeclClass, CountDeclClassVariable, CountDeclInstanceMethod, CountDeclInstanceVariable, CountDeclMethodDefault, CountDeclMethodPrivate, CountDeclMethodProtected, CountInput_Mean, CountInput_Min, CountLineComment, CountOutput_Min, MaxInheritanceTree, MaxNesting_Min, PercentLackOfCohesion, RatioCommentToCode, Added_lines, MAJOR_LINE, MINOR_COMMIT, MINOR_LINE, OWN_COMMIT
groovy-1_6beta2	AvgCyclomaticStrict, AvgEssential, AvgLineBlank, AvgLineComment, CountClassBase, CountClassCoupled, CountClassDerived, CountDeclClass, CountDeclClassVariable, CountDeclInstanceMethod, CountDeclInstanceVariable, CountDeclMethodDefault, CountDeclMethodPrivate, CountDeclMethodProtected, CountInput_Mean, CountInput_Min, CountLineComment, CountOutput_Min, MaxInheritanceTree, MaxNesting_Min, PercentLackOfCohesion, RatioCommentToCode, Added_lines, MAJOR_COMMIT, MAJOR_LINE, MINOR_COMMIT, MINOR_LINE
hbase-0.94.0	AvgEssential, AvgLineBlank, AvgLineComment, CountClassBase, CountClassCoupled, CountClassDerived, CountDeclClassMethod, CountDeclClassVariable, CountDeclMethodDefault, CountDeclMethodProtected, CountInput_Mean, CountInput_Min, CountOutput_Mean, CountOutput_Min, CountPath_Min, MaxInheritanceTree, MaxNesting_Mean, MaxNesting_Min, PercentLackOfCohesion, RatioCommentToCode, Del_lines, MAJOR_COMMIT, MAJOR_LINE, MINOR_COMMIT, MINOR_LINE, OWN_COMMIT
hbase-0.95.0	AvgEssential, AvgLineBlank, AvgLineComment, CountClassBase, CountClassCoupled, CountClassDerived, CountDeclMethodDefault, CountDeclMethodProtected, CountInput_Mean, CountInput_Min, CountLineComment, CountOutput_Min, CountPath_Min, MaxInheritanceTree, MaxNesting_Mean, MaxNesting_Min, PercentLackOfCohesion, RatioCommentToCode, Added_lines, Del_lines, MAJOR_COMMIT, MAJOR_LINE, MINOR_COMMIT, MINOR_LINE, OWN_COMMIT
hbase-0.95.2	AvgEssential, AvgLineBlank, AvgLineComment, CountClassBase, CountClassCoupled, CountClassDerived, CountDeclClass, CountDeclMethodDefault, CountDeclMethodProtected, CountInput_Mean, CountInput_Min, CountLineComment, CountOutput_Min, CountPath_Min, MaxInheritanceTree, MaxNesting_Mean, MaxNesting_Min, PercentLackOfCohesion, RatioCommentToCode, Del_lines, MAJOR_COMMIT, MAJOR_LINE, MINOR_COMMIT, MINOR_LINE, OWN_COMMIT
hive-0.10.0	AvgEssential, AvgLineBlank, AvgLineComment, CountClassBase, CountClassCoupled, CountClassDerived, CountDeclInstanceVariable, CountDeclMethodPrivate, CountDeclMethodProtected, CountInput_Mean, CountInput_Min, CountLineComment, CountOutput_Min, CountPath_Min, MaxInheritanceTree, MaxNesting_Mean, MaxNesting_Min, PercentLackOfCohesion, RatioCommentToCode, Added_lines, MAJOR_LINE, MINOR_COMMIT, MINOR_LINE, OWN_COMMIT
hive-0.12.0	AvgEssential, AvgLineBlank, AvgLineComment, CountClassBase, CountClassCoupled, CountClassDerived, CountDeclClass, CountDeclInstanceVariable, CountDeclMethodDefault, CountDeclMethodPrivate, CountDeclMethodProtected, CountInput_Mean, CountInput_Min, CountLineComment, CountOutput_Min, CountPath_Min, MaxInheritanceTree, MaxNesting_Mean, MaxNesting_Min, PercentLackOfCohesion, RatioCommentToCode, Added_lines, MAJOR_COMMIT, MAJOR_LINE, MINOR_COMMIT, MINOR_LINE, OWN_COMMIT
hive-0.9.0	AvgEssential, AvgLineBlank, AvgLineComment, CountClassBase, CountClassCoupled, CountClassDerived, CountDeclClass, CountDeclClassMethod, CountDeclInstanceVariable, CountDeclMethodPrivate, CountInput_Mean, CountInput_Min, CountLineComment, CountOutput_Min, CountPath_Min, MaxInheritanceTree, MaxNesting_Mean, MaxNesting_Min, PercentLackOfCohesion, RatioCommentToCode, Added_lines, MAJOR_LINE, MINOR_COMMIT, MINOR_LINE, OWN_COMMIT
jruby-1.1	AvgEssential, AvgLineBlank, AvgLineComment, CountClassBase, CountClassCoupled, CountClassDerived, CountDeclClass, CountDeclClassMethod, CountDeclClassVariable, CountDeclInstanceMethod, CountDeclInstanceVariable, CountDeclMethodDefault, CountDeclMethodPrivate, CountDeclMethodProtected, CountInput_Mean, CountInput_Min, CountLineComment, CountOutput_Min, CountPath_Min, MaxInheritanceTree, MaxNesting_Mean, MaxNesting_Min, PercentLackOfCohesion, RatioCommentToCode, Added_lines, MAJOR_COMMIT, MAJOR_LINE, MINOR_COMMIT, MINOR_LINE, OWN_COMMIT
jruby-1.4.0	AvgEssential, AvgLineBlank, AvgLineComment, CountClassBase, CountClassCoupled, CountClassDerived, CountDeclClass, CountDeclClassMethod, CountDeclClassVariable, CountDeclInstanceMethod, CountDeclInstanceVariable, CountDeclMethodDefault, CountDeclMethodPrivate, CountDeclMethodProtected, CountInput_Mean, CountInput_Min, CountLineComment, CountOutput_Mean, CountOutput_Min, CountPath_Min, MaxInheritanceTree, MaxNesting_Mean, MaxNesting_Min, PercentLackOfCohesion, RatioCommentToCode, Added_lines, MAJOR_LINE, MINOR_COMMIT, MINOR_LINE, OWN_COMMIT
jruby-1.5.0	AvgEssential, AvgLineBlank, AvgLineComment, CountClassBase, CountClassCoupled, CountClassDerived, CountDeclClass, CountDeclClassMethod, CountDeclClassVariable, CountDeclInstanceMethod, CountDeclInstanceVariable, CountDeclMethodDefault, CountDeclMethodPrivate, CountDeclMethodProtected, CountInput_Mean, CountInput_Min, CountLineComment, CountOutput_Min, CountPath_Min, MaxInheritanceTree, MaxNesting_Mean, MaxNesting_Min, PercentLackOfCohesion, RatioCommentToCode, Added_lines, MAJOR_LINE, MINOR_COMMIT, MINOR_LINE, OWN_COMMIT
jruby-1.7.0	AvgEssential, AvgLineBlank, AvgLineComment, CountClassBase, CountClassCoupled, CountClassDerived, CountDeclClass, CountDeclClassMethod, CountDeclClassVariable, CountDeclInstanceVariable, CountDeclMethodDefault, CountDeclMethodPrivate, CountDeclMethodProtected, CountInput_Mean, CountInput_Min, CountLineComment, CountOutput_Mean, CountOutput_Min, CountPath_Min, MaxInheritanceTree, MaxNesting_Mean, MaxNesting_Min, PercentLackOfCohesion, RatioCommentToCode, Added_lines, Del_lines, MAJOR_LINE, MINOR_COMMIT, MINOR_LINE, OWN_COMMIT
lucene-2.3.0	AvgEssential, AvgLineBlank, AvgLineComment, CountClassBase, CountClassCoupled, CountClassDerived, CountDeclClass, CountDeclClassMethod, CountDeclClassVariable, CountDeclInstanceVariable, CountDeclMethodDefault, CountDeclMethodPrivate, CountDeclMethodProtected, CountDeclMethodPublic, CountInput_Mean, CountInput_Min, CountOutput_Min, CountPath_Min, MaxInheritanceTree, MaxNesting_Mean, PercentLackOfCohesion, RatioCommentToCode, Added_lines, MAJOR_LINE, MINOR_COMMIT, MINOR_LINE, OWN_COMMIT
lucene-2.9.0	AvgEssential, AvgLineBlank, AvgLineComment, CountClassBase, CountClassCoupled, CountClassDerived, CountDeclClass, CountDeclClassMethod, CountDeclClassVariable, CountDeclInstanceVariable, CountDeclMethodDefault, CountDeclMethodPrivate, CountDeclMethodProtected, CountDeclMethodPublic, CountInput_Mean, CountInput_Min, CountOutput_Mean, CountOutput_Min, CountPath_Min, MaxInheritanceTree, MaxNesting_Mean, PercentLackOfCohesion, RatioCommentToCode, Added_lines, Del_lines, MAJOR_COMMIT, MAJOR_LINE, MINOR_COMMIT, MINOR_LINE, OWN_COMMIT
lucene-3.0.0	AvgEssential, AvgLineBlank, AvgLineComment, CountClassBase, CountClassCoupled, CountClassDerived, CountDeclClass, CountDeclClassMethod, CountDeclClassVariable, CountDeclInstanceVariable, CountDeclMethodDefault, CountDeclMethodPrivate, CountDeclMethodProtected, CountDeclMethodPublic, CountInput_Mean, CountInput_Min, CountLineComment, CountOutput_Mean, CountOutput_Min, MaxInheritanceTree, MaxNesting_Mean, MaxNesting_Min, PercentLackOfCohesion, RatioCommentToCode, Added_lines, Del_lines, MAJOR_COMMIT, MAJOR_LINE, MINOR_COMMIT, MINOR_LINE, OWN_COMMIT
lucene-3.1	AvgEssential, AvgLineBlank, CountClassBase, CountClassCoupled, CountClassDerived, CountDeclClass, CountDeclClassMethod, CountDeclClassVariable, CountDeclInstanceVariable, CountDeclMethodDefault, CountDeclMethodPrivate, CountDeclMethodProtected, CountDeclMethodPublic, CountInput_Mean, CountInput_Min, CountLineComment, CountOutput_Mean, CountOutput_Min, MaxInheritanceTree, MaxNesting_Mean, MaxNesting_Min, PercentLackOfCohesion, RatioCommentToCode, Added_lines, Del_lines, MAJOR_LINE, MINOR_COMMIT, MINOR_LINE, OWN_COMMIT
wicket-1.3.0beta2	AvgCyclomaticModified, AvgEssential, AvgLineBlank, AvgLineComment, CountClassBase, CountClassCoupled, CountClassDerived, CountDeclClass, CountDeclClassMethod, CountDeclInstanceVariable, CountDeclMethodDefault, CountDeclMethodPrivate, CountDeclMethodProtected, CountInput_Mean, CountInput_Min, CountLineComment, CountOutput_Min, MaxInheritanceTree, MaxNesting_Min, PercentLackOfCohesion, RatioCommentToCode, Added_lines, Del_lines, MAJOR_LINE, MINOR_COMMIT, OWN_COMMIT, OWN_LINE
wicket-1.3.0beta1	AvgCyclomaticModified, AvgEssential, AvgLineBlank, AvgLineComment, CountClassBase, CountClassCoupled, CountClassDerived, CountDeclClass, CountDeclClassMethod, CountDeclInstanceVariable, CountDeclMethodDefault, CountDeclMethodPrivate, CountDeclMethodProtected, CountInput_Mean, CountInput_Min, CountLineComment, CountOutput_Mean, CountOutput_Min, MaxInheritanceTree, MaxNesting_Min, PercentLackOfCohesion, RatioCommentToCode, Added_lines, Del_lines, MAJOR_LINE, MINOR_COMMIT, MINOR_LINE, OWN_COMMIT
wicket-1.5.3	AvgEssential, AvgLineBlank, AvgLineComment, CountClassBase, CountClassCoupled, CountClassDerived, CountDeclClass, CountDeclClassMethod, CountDeclInstanceVariable, CountDeclMethodDefault, CountDeclMethodPrivate, CountDeclMethodProtected, CountDeclMethodPublic, CountInput_Mean, CountInput_Min, CountOutput_Min, CountPath_Min, MaxInheritanceTree, MaxNesting_Mean, MaxNesting_Min, PercentLackOfCohesion, RatioCommentToCode, Added_lines, MAJOR_COMMIT, MAJOR_LINE, MINOR_COMMIT, MINOR_LINE

TABLE 4
Comparison results of each method in terms of *AUC*

Project	Unsupervised methods					Supervised methods						
	CLA	CLAMI	SC	ManualDown	ManualUp	LR	RF	NN	MVKNN	EASC	MUSDP_v1	MUSDP_v2
activemq-5.0.0	0.732	0.709	0.758	0.686	0.315	0.722	0.741	0.756	0.697	0.728	0.194	0.809
activemq-5.1.0	0.745	0.709	0.748	0.714	0.286	0.587	0.580	0.616	0.548	0.695	0.744	0.744
activemq-5.2.0	0.711	0.653	0.743	0.682	0.317	0.709	0.700	0.701	0.671	0.688	0.806	0.806
activemq-5.3.0	0.711	0.663	0.730	0.680	0.321	0.620	0.621	0.630	0.565	0.672	0.758	0.758
activemq-5.8.0	0.736	0.734	0.754	0.711	0.289	0.606	0.579	0.615	0.526	0.718	0.771	0.771
derby-10.2.1.6	0.709	0.664	0.718	0.684	0.316	0.762	0.783	0.765	0.761	0.690	0.195	0.805
derby-10.3.1.4	0.689	0.659	0.691	0.673	0.327	0.704	0.713	0.733	0.684	0.676	0.736	0.736
derby-10.5.1.1	0.715	0.709	0.721	0.692	0.308	0.639	0.623	0.667	0.572	0.712	0.688	0.688
groovy-1_5_7	0.716	0.695	0.757	0.661	0.335	0.628	0.595	0.658	0.500	0.799	0.771	0.771
groovy-1_6beta1	0.710	0.649	0.668	0.664	0.337	0.637	0.578	0.711	0.531	0.667	0.783	0.783
groovy-1_6beta2	0.689	0.613	0.693	0.648	0.353	0.603	0.574	0.700	0.519	0.657	0.761	0.761
hbase-0.94.0	0.696	0.660	0.719	0.694	0.307	0.670	0.672	0.690	0.636	0.707	0.752	0.752
hbase-0.95.0	0.644	0.584	0.637	0.628	0.374	0.622	0.619	0.701	0.600	0.655	0.657	0.657
hbase-0.95.2	0.672	0.599	0.667	0.658	0.343	0.632	0.646	0.651	0.630	0.674	0.672	0.672
hive-0.10.0	0.738	0.697	0.762	0.719	0.280	0.662	0.649	0.693	0.596	0.756	0.787	0.787
hive-0.12.0	0.652	0.594	0.678	0.630	0.370	0.571	0.539	0.645	0.511	0.636	0.722	0.722
hive-0.9.0	0.699	0.714	0.711	0.696	0.305	0.657	0.687	0.750	0.601	0.724	0.675	0.675
jruby-1.1	0.755	0.781	0.777	0.738	0.262	0.714	0.739	0.741	0.698	0.794	0.781	0.781
jruby-1.4.0	0.748	0.710	0.753	0.729	0.273	0.661	0.677	0.696	0.640	0.738	0.729	0.729
jruby-1.5.0	0.782	0.764	0.802	0.745	0.255	0.663	0.634	0.664	0.581	0.825	0.771	0.771
jruby-1.7.0	0.753	0.762	0.793	0.735	0.265	0.611	0.565	0.616	0.545	0.746	0.743	0.743
lucene-2.3.0	0.680	0.627	0.687	0.662	0.337	0.815	0.818	0.837	0.772	0.652	0.810	0.809
lucene-2.9.0	0.658	0.638	0.665	0.670	0.329	0.612	0.626	0.677	0.562	0.651	0.711	0.711
lucene-3.0.0	0.667	0.633	0.700	0.686	0.313	0.614	0.586	0.672	0.532	0.675	0.760	0.760
lucene-3.1	0.675	0.694	0.691	0.672	0.329	0.547	0.524	0.549	0.500	0.689	0.672	0.672
wicket-1.3.0beta2	0.748	0.737	0.761	0.721	0.279	0.585	0.566	0.610	0.533	0.726	0.751	0.751
wicket-1.3.0beta1	0.764	0.758	0.783	0.726	0.274	0.658	0.628	0.674	0.572	0.750	0.768	0.772
wicket-1.5.3	0.739	0.758	0.771	0.726	0.275	0.548	0.513	0.545	0.500	0.695	0.714	0.714
Median	0.713	0.683	0.729	0.691	0.309	0.637	0.625	0.676	0.576	0.699	0.741	**0.750**
AR	2.964	4.607	2.250	4.321	8.786	5.607	5.786	3.929	7.286	3.393	2.750	**1.929**

TABLE 5
Comparison results of each method in terms of *MCC*

Project	Unsupervised methods					Supervised methods						
	CLA	CLAMI	SC	ManualDown	ManualUp	LR	RF	NN	MVKNN	EASC	MUSDP_v1	MUSDP_v2
activemq-5.0.0	0.343	0.329	0.383	0.269	-0.268	0.499	0.555	0.537	0.489	0.426	-0.485	0.489
activemq-5.1.0	0.271	0.246	0.274	0.229	-0.228	0.259	0.306	0.284	0.228	0.320	0.284	0.284
activemq-5.2.0	0.271	0.211	0.311	0.224	-0.223	0.503	0.494	0.456	0.498	0.358	0.444	0.444
activemq-5.3.0	0.273	0.228	0.292	0.226	-0.225	0.352	0.359	0.292	0.293	0.326	0.356	0.356
activemq-5.8.0	0.232	0.256	0.250	0.200	-0.199	0.318	0.302	0.280	0.201	0.274	0.273	0.273
derby-10.2.1.6	0.399	0.363	0.425	0.347	-0.347	0.554	0.577	0.542	0.564	0.383	-0.603	0.603
derby-10.3.1.4	0.349	0.342	0.361	0.316	-0.316	0.462	0.472	0.472	0.436	0.369	0.446	0.446
derby-10.5.1.1	0.304	0.341	0.319	0.265	-0.265	0.368	0.367	0.344	0.306	0.333	0.293	0.293
groovy-1_5_7	0.165	0.202	0.198	0.118	-0.118	0.285	0.337	0.395	0.000	0.309	0.207	0.207
groovy-1_6beta1	0.234	0.199	0.203	0.183	-0.183	0.341	0.289	0.475	0.209	0.235	0.331	0.331
groovy-1_6beta2	0.212	0.168	0.227	0.173	-0.171	0.227	0.273	0.469	0.184	0.225	0.311	0.311
hbase-0.94.0	0.315	0.316	0.358	0.312	-0.312	0.406	0.422	0.410	0.362	0.341	0.424	0.424
hbase-0.95.0	0.245	0.196	0.232	0.214	-0.214	0.336	0.336	0.428	0.317	0.265	0.296	0.296
hbase-0.95.2	0.308	0.234	0.299	0.280	-0.278	0.339	0.369	0.314	0.355	0.316	0.338	0.338
hive-0.10.0	0.306	0.328	0.347	0.279	-0.279	0.422	0.408	0.417	0.337	0.391	0.379	0.379
hive-0.12.0	0.166	0.133	0.195	0.143	-0.142	0.199	0.186	0.340	0.113	0.164	0.247	0.247
hive-0.9.0	0.325	0.459	0.346	0.313	-0.312	0.410	0.447	0.515	0.355	0.432	0.290	0.290
jruby-1.1	0.334	0.458	0.381	0.302	-0.305	0.501	0.575	0.517	0.533	0.434	0.378	0.378
jruby-1.4.0	0.389	0.407	0.432	0.354	-0.353	0.407	0.442	0.419	0.409	0.437	0.363	0.363
jruby-1.5.0	0.294	0.336	0.336	0.255	-0.255	0.400	0.390	0.394	0.317	0.396	0.286	0.286
jruby-1.7.0	0.230	0.293	0.280	0.212	-0.212	0.293	0.263	0.275	0.244	0.226	0.222	0.222
lucene-2.3.0	0.313	0.247	0.335	0.276	-0.279	0.641	0.663	0.684	0.602	0.291	0.529	0.529
lucene-2.9.0	0.252	0.251	0.280	0.272	-0.273	0.299	0.352	0.371	0.244	0.264	0.345	0.345
lucene-3.0.0	0.216	0.194	0.269	0.235	-0.236	0.292	0.317	0.393	0.188	0.232	0.335	0.335
lucene-3.1	0.136	0.169	0.154	0.131	-0.131	0.169	0.161	0.134	0.000	0.163	0.137	0.137
wicket-1.3.0beta2	0.261	0.286	0.291	0.232	-0.232	0.246	0.268	0.251	0.191	0.333	0.284	0.284
wicket-1.3.0beta1	0.256	0.279	0.283	0.215	-0.214	0.388	0.410	0.379	0.320	0.328	0.275	0.280
wicket-1.5.3	0.194	0.224	0.224	0.178	-0.178	0.145	0.111	0.133	0.000	0.247	0.176	0.176
Median	0.272	0.265	0.295	0.240	-0.239	0.354	0.369	**0.388**	0.306	0.319	0.304	0.320
AR	5.107	5.036	3.964	6.321	7.964	2.500	**2.000**	2.071	4.357	3.571	4.036	3.250

TABLE 6
Comparison results of each method in terms of P_{opt}

Project	Unsupervised methods					Supervised methods						
	CLA	CLAMI	SC	ManualDown	ManualUp	LR	RF	NN	MVKNN	EASC	MUSDP_v1	MUSDP_v2
activemq-5.0.0	0.550	0.533	0.598	0.449	0.550	0.642	0.664	0.698	0.637	0.555	0.397	0.721
activemq-5.1.0	0.567	0.548	0.578	0.492	0.508	0.548	0.558	0.586	0.561	0.536	0.631	0.631
activemq-5.2.0	0.555	0.510	0.615	0.420	0.581	0.674	0.680	0.686	0.664	0.537	0.732	0.732
activemq-5.3.0	0.539	0.502	0.571	0.436	0.564	0.533	0.516	0.542	0.508	0.488	0.651	0.651
activemq-5.8.0	0.553	0.569	0.589	0.503	0.497	0.585	0.565	0.595	0.564	0.535	0.619	0.619
derby-10.2.1.6	0.557	0.397	0.553	0.221	0.779	0.682	0.703	0.695	0.685	0.505	0.503	0.726
derby-10.3.1.4	0.532	0.395	0.497	0.235	0.764	0.534	0.530	0.617	0.518	0.469	0.609	0.609
derby-10.5.1.1	0.567	0.496	0.556	0.346	0.654	0.520	0.501	0.564	0.513	0.517	0.574	0.574
groovy-1_5_7	0.448	0.557	0.499	0.502	0.497	0.505	0.483	0.553	0.458	0.575	0.570	0.570
groovy-1_6beta1	0.604	0.424	0.472	0.300	0.699	0.450	0.385	0.602	0.375	0.403	0.739	0.739
groovy-1_6beta2	0.568	0.430	0.539	0.304	0.696	0.523	0.481	0.664	0.503	0.425	0.688	0.688
hbase-0.94.0	0.591	0.424	0.650	0.275	0.726	0.552	0.531	0.622	0.500	0.612	0.671	0.671
hbase-0.95.0	0.586	0.330	0.589	0.130	0.871	0.564	0.552	0.657	0.552	0.563	0.539	0.539
hbase-0.95.2	0.601	0.336	0.599	0.142	0.858	0.496	0.514	0.572	0.523	0.562	0.521	0.521
hive-0.10.0	0.692	0.495	0.735	0.252	0.748	0.558	0.538	0.608	0.478	0.633	0.785	0.785
hive-0.12.0	0.621	0.395	0.677	0.198	0.802	0.571	0.585	0.708	0.555	0.496	0.774	0.774
hive-0.9.0	0.585	0.461	0.586	0.221	0.779	0.425	0.444	0.583	0.375	0.493	0.579	0.579
jruby-1.1	0.604	0.522	0.582	0.358	0.642	0.559	0.568	0.590	0.572	0.603	0.617	0.617
jruby-1.4.0	0.649	0.502	0.568	0.244	0.756	0.473	0.458	0.520	0.474	0.536	0.642	0.642
jruby-1.5.0	0.647	0.515	0.613	0.398	0.603	0.562	0.509	0.573	0.499	0.647	0.632	0.632
jruby-1.7.0	0.605	0.519	0.618	0.406	0.594	0.465	0.450	0.477	0.461	0.632	0.632	0.632
lucene-2.3.0	0.563	0.370	0.543	0.263	0.737	0.770	0.769	0.799	0.720	0.464	0.821	0.820
lucene-2.9.0	0.544	0.443	0.513	0.286	0.714	0.517	0.522	0.602	0.523	0.482	0.668	0.668
lucene-3.0.0	0.597	0.482	0.633	0.307	0.694	0.621	0.615	0.681	0.612	0.568	0.738	0.738
lucene-3.1	0.532	0.528	0.536	0.404	0.595	0.413	0.396	0.427	0.376	0.508	0.559	0.559
wicket-1.3.0beta2	0.598	0.550	0.587	0.464	0.537	0.529	0.531	0.564	0.531	0.551	0.578	0.578
wicket-1.3.0beta1	0.605	0.572	0.605	0.489	0.511	0.655	0.641	0.682	0.628	0.592	0.593	0.608
wicket-1.5.3	0.577	0.576	0.628	0.490	0.511	0.590	0.590	0.593	0.593	0.582	0.584	0.584
Median	0.576	0.492	0.582	0.339	**0.661**	0.552	0.541	0.606	0.531	0.538	0.622	0.638
AR	3.714	6.143	3.536	7.929	2.821	4.571	4.964	2.929	5.464	4.893	2.321	**1.857**

TABLE 7
Comparison results of each method in terms of *CostEffort@20%*

Project	Unsupervised methods					Supervised methods						
	CLA	CLAMI	SC	ManualDown	ManualUp	LR	RF	NN	MVKNN	EASC	MUSDP_v1	MUSDP_v2
activemq-5.0.0	0.232	0.220	0.279	0.143	0.237	0.379	0.400	0.447	0.363	0.270	0.123	0.409
activemq-5.1.0	0.292	0.250	0.298	0.168	0.173	0.238	0.264	0.300	0.276	0.255	0.336	0.336
activemq-5.2.0	0.263	0.230	0.318	0.099	0.253	0.442	0.423	0.435	0.385	0.287	0.472	0.472
activemq-5.3.0	0.271	0.224	0.295	0.144	0.247	0.256	0.245	0.286	0.193	0.211	0.381	0.381
activemq-5.8.0	0.247	0.272	0.276	0.181	0.154	0.240	0.222	0.264	0.283	0.233	0.302	0.302
derby-10.2.1.6	0.344	0.220	0.352	0.037	0.434	0.448	0.475	0.469	0.452	0.306	0.215	0.508
derby-10.3.1.4	0.325	0.216	0.302	0.049	0.444	0.345	0.344	0.421	0.327	0.277	0.395	0.395
derby-10.5.1.1	0.289	0.252	0.293	0.074	0.329	0.259	0.231	0.314	0.191	0.248	0.342	0.342
groovy-1_5_7	0.125	0.222	0.154	0.211	0.231	0.270	0.222	0.323	0.143	0.273	0.222	0.222
groovy-1_6beta1	0.345	0.208	0.266	0.071	0.470	0.291	0.190	0.440	0.135	0.240	0.535	0.535
groovy-1_6beta2	0.333	0.158	0.300	0.066	0.480	0.263	0.175	0.448	0.180	0.222	0.500	0.500
hbase-0.94.0	0.350	0.280	0.412	0.026	0.355	0.358	0.320	0.425	0.306	0.380	0.456	0.456
hbase-0.95.0	0.444	0.216	0.476	0.000	0.651	0.324	0.348	0.526	0.408	0.447	0.442	0.437
hbase-0.95.2	0.453	0.229	0.486	0.006	0.617	0.313	0.329	0.415	0.311	0.441	0.422	0.422
hive-0.10.0	0.452	0.319	0.485	0.000	0.380	0.356	0.324	0.446	0.257	0.443	0.562	0.562
hive-0.12.0	0.486	0.264	0.522	0.000	0.550	0.251	0.250	0.438	0.183	0.368	0.616	0.616
hive-0.9.0	0.351	0.327	0.356	0.010	0.427	0.286	0.280	0.445	0.207	0.309	0.373	0.373
jruby-1.1	0.269	0.241	0.241	0.076	0.278	0.333	0.338	0.373	0.333	0.273	0.278	0.278
jruby-1.4.0	0.376	0.307	0.352	0.045	0.421	0.254	0.246	0.328	0.233	0.340	0.407	0.407
jruby-1.5.0	0.300	0.233	0.276	0.077	0.281	0.308	0.250	0.333	0.190	0.333	0.286	0.286
jruby-1.7.0	0.286	0.286	0.324	0.113	0.235	0.217	0.198	0.258	0.219	0.333	0.344	0.344
lucene-2.3.0	0.360	0.255	0.345	0.040	0.411	0.610	0.597	0.646	0.545	0.286	0.561	0.559
lucene-2.9.0	0.319	0.253	0.313	0.045	0.381	0.256	0.265	0.390	0.262	0.286	0.433	0.433
lucene-3.0.0	0.357	0.281	0.414	0.037	0.353	0.423	0.452	0.504	0.447	0.346	0.533	0.533
lucene-3.1	0.293	0.258	0.300	0.108	0.241	0.175	0.146	0.195	0.129	0.261	0.307	0.307
wicket-1.3.0beta2	0.271	0.279	0.288	0.129	0.172	0.224	0.211	0.260	0.194	0.297	0.302	0.302
wicket-1.3.0beta1	0.263	0.269	0.307	0.176	0.190	0.342	0.302	0.390	0.242	0.313	0.272	0.296
wicket-1.5.3	0.238	0.282	0.308	0.131	0.179	0.214	0.218	0.221	0.279	0.284	0.266	0.266
Median	0.314	0.250	0.320	0.068	0.332	0.296	0.280	0.385	0.260	0.295	0.382	**0.398**
AR	4.000	5.786	3.464	7.964	3.857	4.429	4.821	2.536	5.429	4.500	2.500	**1.893**

TABLE 8
Comparison results of MUSDP and its variants in terms of *AUC*

Project	MUSDP_v1				MUSDP_v2			
	CP	CO	PO	ALL	CP	CO	PO	ALL
activemq-5.0.0	0.793	0.779	0.186	0.194	0.793	0.779	0.814	0.809
activemq-5.1.0	0.744	0.737	0.688	0.744	0.744	0.737	0.688	0.744
activemq-5.2.0	0.750	0.782	0.782	0.806	0.750	0.782	0.782	0.806
activemq-5.3.0	0.739	0.746	0.688	0.758	0.739	0.746	0.688	0.758
activemq-5.8.0	0.777	0.751	0.765	0.771	0.777	0.751	0.765	0.771
derby-10.2.1.6	0.728	0.227	0.205	0.195	0.728	0.773	0.795	0.805
derby-10.3.1.4	0.691	0.726	0.668	0.736	0.691	0.726	0.332	0.736
derby-10.5.1.1	0.717	0.730	0.584	0.688	0.717	0.730	0.584	0.688
groovy-1_5_7	0.772	0.730	0.776	0.771	0.772	0.730	0.776	0.771
groovy-1_6beta1	0.748	0.734	0.763	0.783	0.748	0.732	0.763	0.783
groovy-1_6beta2	0.724	0.674	0.769	0.761	0.724	0.674	0.769	0.761
hbase-0.94.0	0.666	0.762	0.723	0.752	0.666	0.762	0.723	0.752
hbase-0.95.0	0.624	0.674	0.665	0.657	0.624	0.672	0.665	0.657
hbase-0.95.2	0.586	0.687	0.664	0.672	0.586	0.687	0.664	0.672
hive-0.10.0	0.747	0.775	0.773	0.787	0.747	0.775	0.773	0.787
hive-0.12.0	0.707	0.699	0.723	0.722	0.707	0.320	0.277	0.722
hive-0.9.0	0.700	0.692	0.634	0.675	0.700	0.692	0.634	0.675
jruby-1.1	0.801	0.751	0.767	0.781	0.801	0.716	0.767	0.781
jruby-1.4.0	0.727	0.746	0.676	0.729	0.727	0.745	0.674	0.729
jruby-1.5.0	0.782	0.748	0.733	0.771	0.782	0.747	0.277	0.771
jruby-1.7.0	0.770	0.716	0.692	0.743	0.770	0.292	0.692	0.743
lucene-2.3.0	0.693	0.790	0.816	0.810	0.693	0.747	0.816	0.809
lucene-2.9.0	0.699	0.694	0.684	0.711	0.699	0.691	0.684	0.711
lucene-3.0.0	0.717	0.736	0.733	0.760	0.717	0.736	0.733	0.760
lucene-3.1	0.694	0.664	0.600	0.672	0.694	0.664	0.404	0.672
wicket-1.3.0beta2	0.758	0.728	0.726	0.751	0.758	0.728	0.726	0.751
wicket-1.3.0beta1	0.782	0.276	0.712	0.768	0.782	0.726	0.749	0.772
wicket-1.5.3	0.738	0.715	0.650	0.714	0.738	0.715	0.650	0.714
Median	0.730	0.727	0.703	0.741	0.730	0.728	0.706	0.750
AR	2.036	2.286	2.786	1.786	2.107	2.500	2.643	1.607

TABLE 9
Comparison results of MUSDP and its variants in terms of *MCC*

Project	MUSDP_v1				MUSDP_v2			
	CP	CO	PO	ALL	CP	CO	PO	ALL
activemq-5.0.0	0.439	0.412	-0.532	-0.485	0.439	0.412	0.532	0.489
activemq-5.1.0	0.276	0.258	0.245	0.284	0.276	0.258	0.245	0.284
activemq-5.2.0	0.329	0.366	0.547	0.444	0.329	0.366	0.547	0.444
activemq-5.3.0	0.311	0.312	0.319	0.356	0.311	0.312	0.319	0.356
activemq-5.8.0	0.278	0.246	0.291	0.273	0.278	0.246	0.291	0.273
derby-10.2.1.6	0.474	-0.526	-0.585	-0.603	0.474	0.526	0.585	0.603
derby-10.3.1.4	0.389	0.421	0.315	0.446	0.389	0.421	-0.315	0.446
derby-10.5.1.1	0.320	0.332	0.150	0.293	0.320	0.332	0.150	0.293
groovy-1_5_7	0.234	0.174	0.237	0.207	0.234	0.174	0.237	0.207
groovy-1_6beta1	0.315	0.266	0.303	0.331	0.315	0.258	0.303	0.331
groovy-1_6beta2	0.273	0.205	0.328	0.311	0.273	0.205	0.328	0.311
hbase-0.94.0	0.303	0.423	0.376	0.424	0.303	0.423	0.376	0.424
hbase-0.95.0	0.259	0.305	0.313	0.296	0.259	0.303	0.313	0.296
hbase-0.95.2	0.228	0.348	0.311	0.338	0.228	0.348	0.311	0.338
hive-0.10.0	0.349	0.354	0.361	0.379	0.349	0.354	0.361	0.379
hive-0.12.0	0.247	0.216	0.244	0.247	0.247	-0.197	-0.244	0.247
hive-0.9.0	0.345	0.314	0.242	0.290	0.345	0.314	0.242	0.290
jruby-1.1	0.441	0.323	0.381	0.378	0.441	0.273	0.381	0.378
jruby-1.4.0	0.407	0.387	0.276	0.363	0.407	0.386	0.271	0.363
jruby-1.5.0	0.340	0.254	0.242	0.286	0.340	0.254	-0.232	0.286
jruby-1.7.0	0.274	0.193	0.174	0.222	0.274	-0.185	0.174	0.222
lucene-2.3.0	0.356	0.497	0.554	0.529	0.356	0.423	0.554	0.529
lucene-2.9.0	0.334	0.313	0.302	0.345	0.334	0.310	0.302	0.345
lucene-3.0.0	0.294	0.305	0.305	0.335	0.294	0.305	0.305	0.335
lucene-3.1	0.158	0.128	0.077	0.137	0.158	0.128	-0.073	0.137
wicket-1.3.0beta2	0.297	0.250	0.254	0.284	0.297	0.250	0.254	0.284
wicket-1.3.0beta1	0.295	-0.209	0.207	0.275	0.295	0.214	0.245	0.280
wicket-1.5.3	0.194	0.170	0.132	0.176	0.194	0.170	0.132	0.176
Median	0.308	0.289	0.274	0.304	0.308	0.287	0.281	0.320
AR	1.929	2.464	2.643	1.857	2.036	2.643	2.536	1.679

TABLE 10
Comparison results of MUSDP and its variants in terms of P_{opt}

Project	MUSDP_v1				MUSDP_v2			
	CP	CO	PO	ALL	CP	CO	PO	ALL
activemq-5.0.0	0.682	0.674	0.344	0.397	0.682	0.674	0.753	0.721
activemq-5.1.0	0.594	0.604	0.625	0.631	0.594	0.604	0.625	0.631
activemq-5.2.0	0.627	0.699	0.774	0.732	0.627	0.699	0.774	0.732
activemq-5.3.0	0.603	0.642	0.638	0.651	0.603	0.642	0.638	0.651
activemq-5.8.0	0.622	0.613	0.639	0.619	0.622	0.613	0.639	0.619
derby-10.2.1.6	0.563	0.565	0.400	0.503	0.563	0.679	0.779	0.726
derby-10.3.1.4	0.486	0.614	0.707	0.609	0.486	0.614	0.477	0.609
derby-10.5.1.1	0.566	0.603	0.529	0.574	0.566	0.603	0.529	0.574
groovy-1_5_7	0.534	0.528	0.638	0.570	0.534	0.528	0.638	0.570
groovy-1_6beta1	0.587	0.687	0.764	0.739	0.587	0.672	0.764	0.739
groovy-1_6beta2	0.592	0.550	0.767	0.688	0.592	0.550	0.767	0.688
hbase-0.94.0	0.486	0.728	0.648	0.671	0.486	0.727	0.648	0.671
hbase-0.95.0	0.448	0.610	0.563	0.539	0.448	0.607	0.563	0.539
hbase-0.95.2	0.338	0.590	0.572	0.521	0.338	0.590	0.541	0.521
hive-0.10.0	0.659	0.803	0.804	0.785	0.659	0.803	0.804	0.785
hive-0.12.0	0.682	0.816	0.938	0.774	0.682	0.596	0.397	0.774
hive-0.9.0	0.553	0.626	0.539	0.579	0.553	0.626	0.539	0.579
jruby-1.1	0.595	0.668	0.697	0.617	0.595	0.571	0.650	0.617
jruby-1.4.0	0.541	0.708	0.624	0.642	0.541	0.702	0.615	0.642
jruby-1.5.0	0.572	0.611	0.697	0.632	0.572	0.606	0.481	0.632
jruby-1.7.0	0.592	0.650	0.619	0.632	0.592	0.452	0.619	0.632
lucene-2.3.0	0.580	0.832	0.870	0.821	0.580	0.744	0.870	0.820
lucene-2.9.0	0.616	0.654	0.649	0.668	0.616	0.648	0.649	0.668
lucene-3.0.0	0.657	0.736	0.707	0.738	0.657	0.736	0.707	0.738
lucene-3.1	0.559	0.543	0.594	0.559	0.559	0.543	0.404	0.559
wicket-1.3.0beta2	0.586	0.572	0.561	0.578	0.586	0.572	0.561	0.578
wicket-1.3.0beta1	0.607	0.519	0.558	0.593	0.607	0.589	0.606	0.608
wicket-1.5.3	0.593	0.571	0.609	0.584	0.593	0.571	0.609	0.584
Median	0.587	0.629	0.640	0.622	0.587	0.622	0.626	0.638
AR	2.964	2.071	1.821	1.964	2.821	2.179	2.000	1.679

TABLE 11
Comparison results of MUSDP and its variants in terms of *CostEffort@20%*

Project	MUSDP_v1				MUSDP_v2			
	CP	CO	PO	ALL	CP	CO	PO	ALL
activemq-5.0.0	0.362	0.343	0.121	0.123	0.362	0.343	0.472	0.409
activemq-5.1.0	0.301	0.305	0.354	0.336	0.301	0.305	0.354	0.336
activemq-5.2.0	0.329	0.406	0.578	0.472	0.329	0.406	0.578	0.472
activemq-5.3.0	0.318	0.359	0.398	0.381	0.318	0.359	0.398	0.381
activemq-5.8.0	0.309	0.301	0.359	0.302	0.309	0.301	0.359	0.302
derby-10.2.1.6	0.359	0.239	0.186	0.215	0.359	0.458	0.543	0.508
derby-10.3.1.4	0.308	0.395	0.461	0.395	0.308	0.395	0.175	0.395
derby-10.5.1.1	0.312	0.338	0.294	0.342	0.312	0.338	0.294	0.342
groovy-1_5_7	0.182	0.160	0.310	0.222	0.182	0.160	0.310	0.222
groovy-1_6beta1	0.406	0.455	0.565	0.535	0.406	0.440	0.565	0.535
groovy-1_6beta2	0.364	0.308	0.591	0.500	0.364	0.308	0.591	0.500
hbase-0.94.0	0.314	0.487	0.434	0.456	0.314	0.482	0.434	0.456
hbase-0.95.0	0.340	0.498	0.462	0.442	0.340	0.493	0.462	0.437
hbase-0.95.2	0.211	0.481	0.459	0.422	0.211	0.481	0.425	0.422
hive-0.10.0	0.459	0.550	0.593	0.562	0.459	0.550	0.593	0.562
hive-0.12.0	0.558	0.611	0.736	0.616	0.558	0.211	0.023	0.616
hive-0.9.0	0.358	0.374	0.367	0.373	0.358	0.374	0.367	0.373
jruby-1.1	0.278	0.308	0.352	0.278	0.278	0.269	0.292	0.278
jruby-1.4.0	0.333	0.440	0.395	0.407	0.333	0.430	0.386	0.407
jruby-1.5.0	0.274	0.268	0.353	0.286	0.274	0.259	0.143	0.286
jruby-1.7.0	0.333	0.313	0.365	0.344	0.333	0.176	0.365	0.344
lucene-2.3.0	0.360	0.566	0.622	0.561	0.360	0.490	0.622	0.559
lucene-2.9.0	0.415	0.409	0.458	0.433	0.415	0.401	0.458	0.433
lucene-3.0.0	0.472	0.509	0.533	0.533	0.472	0.509	0.533	0.533
lucene-3.1	0.310	0.316	0.326	0.307	0.310	0.316	0.218	0.307
wicket-1.3.0beta2	0.314	0.255	0.296	0.302	0.314	0.255	0.296	0.302
wicket-1.3.0beta1	0.304	0.129	0.212	0.272	0.304	0.270	0.300	0.296
wicket-1.5.3	0.274	0.235	0.290	0.266	0.274	0.235	0.290	0.266
Median	0.331	0.367	0.403	0.382	0.331	0.358	0.397	0.398
AR	2.643	2.357	1.464	2.000	2.607	2.321	1.536	1.679