

TABLE I: Data extraction table

ID	RQ1	RQ2	RQ3	RQ4
[8]	Jdeodorant	Long Method, Feature Envy, Type Checking, Duplicated Code, God Class	NA	NA, NA, Java
[26]	MTV-Checker	Meddling Model, Meddling View, Brain Persistence Method+, Improper Use of Manager, Laborious Persistence Method	Tree-based	NA, No, Python
[7]	Jdeodorant	NA	NA	NA, NA, Java
[44]	JSNose, aDoctor, Jdeodorant, JSPIRIT, DECOR, Stench Blossom	JSNose (Lazy Object, Long Message Chain, Closure, Coupling, Empty Cash, Global Variable, Large Object, Long Method, Long Parameter List, Nested Callback, Refused Bequest, Switch Statement, Dead Code), aDoctor, Jdeodorant, JSPIRIT, DECOR, Stench Blossom (NA)	NA	JSNose (NA, NA, JavaScript), aDoctor, Jdeodorant, JSPIRIT, DECOR, Stench Blossom (NA, NA, NA)
[67]	DECOR, Jdeodorant	DECOR (Blob+, God Class+, Spaghetti Code, Class, Data Should Be Private, Complex Class+, Long Method+), Jdeodorant (Feature Envy+, State Checking+, Long Method+, Blob+)	DECOR (Rule-based), Jdeodorant (Metric-based)	DECOR (Yes, NA, Java), Jdeodorant (NA, NA, Java)
[96]	GodExpo	God Class+	Metric-based	Yes, No, Golang
[69]	Arcan	Unstable Dependency, Hub-Like Dependency, Cyclic Dependency, Feature Concentration	Graph-based	NA, NA, Java
[14]	JSPIRIT, Jdeodorant, FindSmells	JSPIRIT (Large Class+, Long Method+, Data Class+, Refused Bequest+, Feature Envy+), Jdeodorant (God Class, God Method, Feature Envy, Switch Statement), FindSmell (NA)	JSPIRIT(NA), Jdeodorant (NA), FindSmells (Rule-based)	JSPIRIT(NA, No, Java), Jdeodorant (NA, NO, Java), FindSmells (NO, NA, NA)
[4]	DECOR, iPlasma, inCode, inFusion, jDeodorant, PMD, jCOSMO, Analyst4j, Cultivate (from jTransformer suite), Reek, RevJava, SA4J, Together, Stench Blossom, ConcernReCS, SourceMiner, BSDT, JCodeCanine,, CodeVizard, JSNose, Hist-Inspect, PTIDEJ suite (containing DECOR and its evolution DETEX), BLOP, JSPIRIT, SpIRIT, CCFinder, Sissy, Understand, Pysmell	inFusion (Brain Method+, Cyclic Dependencies+, Data Class+, Feature Envy+, God Class+, Intensive Coupling+, Missing Template Method+, Refused Bequest+, Sibling Duplication+, Shotgun Surgery+, SAP-Breakers+, Internal Duplication+, External Duplication+, Blob+, Blob Operation+, Data Clump+, Message Chain+, Distorted Hierarchy+, Schizophrenic Class+, Tradition Breaker+, Unstable Dependencies+), inCode (Data Class+, Data Clump+, Duplicate Code+, Feature Envy+, God Class+, God Method+, Misplaced Class+, Refused Bequest+, Shotgun Surgery+, Message Chain+, Schizophrenic Class+), iPlasma (Brain Class+, Brain Method+, Data Class+, Dispersed Coupling+, Feature Envy+, God Class+, Intensive Coupling+, Shotgun Surgery+, Refused Bequest+, Tradition Breaker+, Long Method+, Long Parameter List+, Speculative Generality+), JDeodorant (Feature Envy+, God Class+, Long Method+, Type Checking, Duplicate code), PMD (Large Class+, Long Method+, Long Parameter List+), DECOR (Large Class+, Lazy Class+, Long Method+, Long Parameter List+, Refused Bequest+, Speculative Generality, Message Chain+, Shotgun Surgery+, Duplicate Code, Comments, Data Class+, No Polymorphism, Global Variable, Controller Class+, Procedural Class, Low Cohesion+, Divergent Change+, Blob+, Spaghetti Code+, Functional Decomposition, Swiss Army Knife), jCOSMO, Analyst4j, Cultivate (from jTransformer suite), Reek, RevJava, SA4J, Together, Stench Blossom, ConcernReCS, SourceMiner, BSDT, JCodeCanine,, CodeVizard, JSNose, Hist-Inspect, PTIDEJ suite (containing DECOR and its evolution DETEX), BLOP, JSPIRIT, SpIRIT, CCFinder, Sissy, Understand, Pysmell (NA)	inFusion (Metric-based), inCode (Metric-based), iPlasma (Metric-based), Jdeodorant (Tree-based), PMD (Rule-based), DECOR (Metric-based), jCOSMO, Analyst4j, Cultivate (from jTransformer suite), Reek, RevJava, SA4J, Together, Stench Blossom, ConcernReCS, SourceMiner, BSDT, JCodeCanine, CodeVizard, JSNose, Hist-Inspect, PTIDEJ suite (containing DECOR and its evolution DETEX), BLOP, JSPIRIT, SpIRIT, CCFinder, Sissy, Understand, Pysmell (NA)	inFusion (NA, No, (Java, C++, C)), inCode (NA, No, (Java, C++, C)), iPlasma(NA, No, (Java, C++)), Jdeodorant (NA, No, Java), PMD (NA, NA, (Java, C, C++, C#, PHP, Ruby, Python etc.)), DECOR (NA, NA, Java), CCFinder (NA, NA (COBOL, Visual Basic)), Pysmell(NA, NA, Python), Sissy (NA, NA, (Object-Pascal, Delphi)), jCOSMO, Analyst4j, Cultivate (from jTransformer suite), Reek, RevJava, SA4J, Together, Stench Blossom, ConcernReCS, SourceMiner, BSDT, JCodeCanine, CodeVizard, JSNose, Hist-Inspect, PTIDEJ suite (containing DECOR and its evolution DETEX), BLOP, JSPIRIT, SpIRIT, Understand, (NA, NA, NA)
[3]	iPerfDetector	Accessing data models on the UI thread, Updating UIs on background threads, Applying UI blurring effect inefficiently, Applying UI transparency inefficiently, Retain cycle	Tree-based	Yes, No, Swift
[29]	cASpER	Feature Envy+, Misplaced Class+, Blob+, Promiscuous Package	Rule-based, Text-based	Yes, No, Java

ID	RQ1	RQ2	RQ3	RQ4
[52]	Droidlens	Internal Getter/Setter+, Leaking Inner Class+, Member Ignoring Method+, No Low Memory Resolver+, HashMap Usage+, Data Transmission Without Compression+, Inefficient Data Format And Parser+, Unclosed Closeable+, Inefficient SQL Query+, Leaking Thread+, Slow Loop+, Durable Wakelock+, Public Data+, Rigid Alarm-Manager+, Debuggable Release+, Heavy Broadcast Receiver+, Heavy Service Start+, Set Config Changes+	Metric-based	NA, No, Java
[5]	TAJSLint	Argument Count Mismatch+,Argument Type Mismatch, Array Length Assignment, Primitive Property Assignment, Negative Array Index, Udeclared Variables, Duplicate Declaration+, Unused Declaration+, Unreachable Code, Loosely-typed Variable, Lon Method Chain+, Large/Lazy Class+, Long parametr list+, Cyclomethc Complexity+	Metric-based, Rule-based	Yes, NA, JavaScript
[84]	XData grading system	Long Method+	Metric-based, Graph-based	No, NA, Language independent
[88]	NiCad, CPD, iClones, TCORE	Code clones	NiCad (Text-based), CPD (Token-based), iClones (Token-based, Tree-based), TCORE (Tree-based, Graph-based)	NiCad (Yes, No, Java), CPD (Yes, No, Java), iClones (Yes, No, Java), TCORE (Yes, No, Java)
[11]	No name, Table E1	Internal Getter/Setter+, Leaking Inner Class+, Member Ignoring Method+, No Low Memory Resolver+, HashMap Usage+, Data Transmission Without Compression+, Inefficient Data Format And Parser+, Unclosed Closeable+, Inefficient SQL Query+, Leaking Thread+, Slow Loop+, Durable Wakelock+, Public Data+, Rigid Alarm-Manager+, Debuggable Release+, Heavy Broadcast Receiver+, Heavy Service Start+, Set Config Changes+, Large Class+, Data Class+, Long Method+, Lazy Class+, Long Parameter List+, Feature Envy+, Refused Bequest+, Switch Statements+, Duplicate Code+, Message Chain+, Parallel Inheritance Hierarchy+, Speculative Generality+, Shotgun Surgery+, Primitive Obsession+, Temporary Field+, Middle Man+, Divergent Change+, Alternative Classes with Different Interface+, Inappropriate Intimacy+, Data Clump+ Table E1	No name (Metric-based), Tools from Table E1 (NA)	Yes, No, Java Table E1
[77]	Designite, Ndepend, Jarchitect, Lattix Architect, SotoArc, Structure101	Designite (Broken Hierarchy, Broken Modularization, Cyclically-dependent Modularization, Duplicate Abstraction, Hub-like Modularization, Imperative Abstraction, Insufficient Modularization, Multipath Hierarchy, Rebellious Hierarchy, Unnecessary Abstraction, Unutilized Abstraction, Wide Hierarchy, Cyclic Dependency+, Unstable Dependency+, God Component+, Feature Concentration+, Scattered Functionality+, Dense Structure+), Ndepend (Insufficient modularization, multifaceted abstraction, unutilized, abstraction, deep hierarchy), Jarchitect (Insufficient modularization, multifaceted abstraction, unutilized abstraction, deep hierarchy)	Ndepend (Metric-based), Jarchitect (Metric-based), Lattix Architect (Rule-based), SotoArc (Graph-based), Structure101 (Graph-based), Designite (Rule-based)	Designite (Yes, Yes, C#), Ndepend (Yes, NA, C#), Jarchitect (Yes, NA, C#), Lattix Architect, SotoArc, Structure101 (NA, NA, NA)

ID	RQ1	RQ2	RQ3	RQ4
[2]	<p>ADOCTOR, Arcan, Asta, c-JRefRec, CBCD, CCFinder, ClemanX, CloneDetection, Coogle (Eclipse plugin), DECKARD, DECOR, Dup, BSDT (Eclipse plugin), JCodeCanine (Eclipse plugin), LCAAnalyzer (Eclipse plugin), SCOOP (Eclipse plugin), FindSmells, Gemini, HIST, HULK, jCOSMO, Jdeodorant, Jplag, JBOO, JSNOSE, JSPIRIT, LAPD, LBSDetectors, LIME (Eclipse plugin), NICAD, PAD, Paprika, PC Detector, PRODETECTION, Pysmell, Smell Platform, SourcererCC, TACO, DuplicationDetector, XIAO</p>	<p>ADOCTOR (Data transmission without compression, Debuggable release, Durable wakelock, Inefficient data format and parser, Inefficient data structure, Inefficient SQL query, Internal getter and setter, Leaking inner class, leaking thread, Memberignoring method, No low memory resolver, Public data, Rigid alarm manager, Slow loop, Unclosed closable), Arcan (Unstable dependency+, hub-like dependency+, cyclic dependency+), Asta (Code clones), c-JRefRec (Feature envy), CBCD (Code clones), CCFinder (Code clones), ClemanX (Code clones), CloneDetection (Code clones), Coogle (Code clones), DECKARD (Code clones), DECOR(Blob, Functional decomposition, Spaghetti code, Swiss Army Knife), Dup (Code clones), BSDT (Large class+, Long method+, Data class+, Parallel inheritance hierarchies+, Long parameter list+, Switch statements+, Lazy class+), JCodeCanine (Feature envy+), LCAAnalyzer (Nullable objects, explicit cast, wrong verbosity level, Dup1, Dup2, and malformed output), SCOOP (Hereditary anomaly, replicated external network, multiple-anomaly syndrome), FindSmells (Large class, Long method, Data class, Feature envy, and Refused bequest), Gemini (Code clones), HIST (Divergent change, Shotgun surgery, Parallel inheritance, Blob, Feature envy), HULK (God class, Blob, Swiss Army knife, Spaghetti code), jCOSMO (Instanceof, Typecast), Jdeodorant (Feature envy), Jplag (Code clones), JBOO (Use case clones), JSNOSE (Closure smells, Coupling between JavaScript, HTML, and CSS, Excessive global variables, Long message chain, Nested Callback, Refused bequest, Empty catch, Large/Lazy object, Long method/Function, Long parameter list, Switch statement, Unused/dead code), JSPIRIT (Brain class+, Brain method+, Data class+, Dispersed coupling+, Feature envy, God class+, Intensive coupling+, Refused parent bequest+, Shotgun surgery+, Tradition breaker), LAPD (Linguistic antipatterns), LBSDetectors (Odd grammatical structure, Term used to name both the whole and its parts, Inconsistent identifier use, Useless type indication, Identifier construction rules), LIME (Code clones), NICAD (Code clones), PAD (Empty semitrucks, Circuitous treasure hunt, Blob), Paprika (Blob, Swiss Army knife, Complex class, Long method, Internal getter/setter, No low memory resolver, Member ignoring method, Leaking inner class), PC Detector (Code clones), PRODETECTION (God class, Shotgun surgery, Refused bequest, Feature envy, God method, Misplaced class, God package, Wide subsystem, Interface, Data class), Pysmell (Large class+, Long parameter list+, Long method+, Long message chain+, Long scope chaining+, Long Base class list+, Useless exception handling+, Long lambda function+, Complex list comprehension+, Long element chain+, Long ternary+, Conditional expression+), Smell Platform (Long parameter list, Long method, God class, Feature envy), SourcererCC (Code clones), TACO (Long method, Feature envy, Blob, Promiscuous package, Misplaced class), DuplicationDetector (Code clones), XIAO (Code clones)</p>	<p>ADOCTOR (Rule-based), Arcan (Metrics-based), Asta (Token-based), c-JRefRec (Graph-based), CBCD (Graph-based), CCFinder (Tree-based), ClemanX (Tree-based), CloneDetection (Token-based),Coogle (Metrics-based), DECKARD (Tree-based), DECOR (Rule-based), Dup (Token-based), BSDT (Metrics-based), JCodeCanine (Metrics-based), LCAAnalyzer (Tree-based), SCOOP (Rule-based), FindSmells (Metrics-based), Gemini (Metrics-based), HIST (Rule-based), HULK (Rule-based), jCOSMO (Tree-based), Jdeodorant (Tree-based), Jplag (NA), JBOO (Graph-based), JSNOSE (Metrics-based), JSPIRIT (Metrics-based), LAPD (Text-based), LBSDetectors (Text-based), LIME (NA), NICAD (Text-based), PAD (Tree-based), Paprika (Graph-based), PC Detector (Token-based), PRODETECTION (Rule-based), Pysmell (Metrics-based), Smell Platform (Rule-based), SourcererCC (Token-based), TACO (Text-based), DuplicationDetector (Tree-based), XIAO (Text-based)</p>	<p>ADOCTOR (NA, NA, Java, Arcan (NA, NA, Java), Asta (NA, NA, NA), c-JRefRecCBCD (NA, NA, Java), CBCD (NA, NA, NA), CCFinder (NA, NA, (Java, C, C++)), ClemanX (NA, NA, NA), CloneDetection (NA, NA, (Java, C, C++)), Coogle (NA, NA, Java), DECKARD (NA, NA, NA), DECOR (NA, NA, NA), Dup (NA, NA, NA, NA), BSDT (NA, NA, NA), JCodeCanine (NA, NA, Java), LCAAnalyzer (NA, NA, Java), SCOOP (NA, NA, Java), FindSmells (NA, NA, NA), Gemini (NA, NA, NA), HIST (NA, NA, Java), HULK (NA, NA, Java), jCOSMO (NA, NA, Java), Jdeodorant (NA, NA, Java, Jplag (NA, NA, (C, C++)), JBOO (NA, NA, NA), JSNOSE (NA, NA, NA), JSPIRIT (NA, NA, Java), LAPD (NA, NA, Java), LBSDetectors (NA, NA, NA), LIME (NA, NA, NA), NICAD (NA, NA, NA), PAD (NA, NA, NA), Paprika (NA, NA, Java), PC Detector (NA, NA, (C, C++)), PRODETECTION (NA, NA, Java), Pysmell (NA, NA, NA), Smell Platform (NA, NA, NA), SourcererCC (NA, NA, (Java, C)), TACO (NA, NA, NA), DuplicationDetector (NA, NA, NA), XIAO (NA, NA, (C, C++))</p>

ID	RQ1	RQ2	RQ3	RQ4
[65]	No name	Empty catch block, Hard-coded secret, Empty password, Insecure hash algorithm, Smelly methods usage	Text-based, Metric-based	No, No, Java
[79]	Grace	Structural coupling, Co-evolution coupling	Graph-based	NA, No, Language independent DECOR (Yes, NA,(Java, C++)), PTIDEJ (Yes, NA, (Java, C++)), inCode (NA, No, (Java, C++)), inFusion (No, No, (Java, C++)), JDeodorant (No, No, Java), Bortland Together (NA, Na, Java), iPlasma (No, No, (Java, C++)), PMD (Yes, NA, (Java, C++)), Organic/SCOOP (NA,NA, Java), , Stench Blossom (NA, NA, Java), , Others from Table 14 (NA, NA, NA)
[81]	Table 14	Table 14	DECOR (Metric-bases, Rule-based), PTIDEJ (Metric-bases, Rule-based), inCode (Metric-based), inFusion (Metric-based), iPlasma (Metric-based), PMD (Rule-based), Others from Table 14 (NA)	
[47]	DLFinder	Code clones	Tree-based, Text-based	NA, No
[59]	Arcan, Sonargraph, Structure101, CLIO, Designite, DV8, SonarQube, inFusion, AI Reviewer, ARCADE, Hotspot Detector, Massey Architecture Explore, STAN,, SODA	Arcan (Unstable Dependency, Hub-like Dependency, Cyclic Dependency, Multiple Architecture Violations, Specification–implementation Violation), Sonargraph (Unstable Dependency, Hub-like Dependency, Cyclic Dependency, Package Cycles, Biggest Package Cycle Group), Structure101(Unstable Dependency, Hub-like Dependency, Cyclic Dependency, Multiple Architecture Violations, Specification–implementation Violation), CLIO (Modularity Violations), Designite (Unstable Dependency, Cyclic Dependency, Ambiguous Interface, God Component, Feature Concentration, Scattered Functionality, Dense Structure), DV8 (Package Cycles, Unstable Interface, Unhealthy Inheritance Hierarchy, Modularity Violations, Crossing, Clique), SonarQube (Cyclic Dependency), inFusion (Unstable Dependency, Cyclic Dependency,SAP Breaker), AI Reviewer, Hotspot Detector, Massey Architecture Explorer, STAN (Unstable Dependency, Hub-like Dependency, Cyclic Dependency, Implicit Cross-module Dependency, Ambiguous Interface, Cyclic Hierarchy, Multipath Hierarchy, Abstraction without Decoupling, Unutilized Abstraction, God Component, Scattered Functionality, Multiple Architecture Violations), ARCADE (Unstable Dependency, Hub-like Dependency, Cyclic Dependency, Implicit Cross-module Dependency, Ambiguous Interface, Cyclic Hierarchy, Multipath Hierarchy, Abstraction without Decoupling, Unutilized Abstraction, God Component, Scattered Functionality, Multiple Architecture Violations, Unused Interface, Sloppy Delegation, Co-change Coupling, Separation of Concerns, Concern Overload, Link Overload), SODA (Multi-service, Tiny Service, Sand Pile, Chatty Service, Knot Service, Nobody Home, Duplicated Service, Bottleneck Service, Service Chain, Data Service, Bloated Service)	Arcan (Graph-based), Sonargraph (Graph-based), Structure101 (Graph-based), CLIO (Metric-based), Designite (Rule-based), DV8 (Rule-based), SonarQube (Graph-based), inFusion (NA), AI Reviewer (Graph-based), ARCADE (Graph-based), Hotspot Detector (Graph-based), Massey Architecture Explore (NA), STAN (Graph-based), SODA (Rule-based)	Arcan (NA, NA, Java), Sonargraph (NA, NA, (Java, C, C++ C#, Python)), Structure101 (NA, NA, (Java, C++, C, Python)), CLIO (NA, NA, (Java, C++)), Designite (NA, NA, (Java, C#)), DV8 (NA, NA, (Java, C++, C, C#, Python)), SonarQube (NA, NA, (Java, C++, C, C#, Python)), inFusion (NA, NA, NA), AI Reviewer (NA, NA, (C++,C)), ARCADE (NA, NA, Java), Hotspot Detector (NA, NA, NA), Massey Architecture Explore (NA, NA, NA), STAN (NA, NA, Java) SODA (NA, NA, Java)
[89]	NiCad, CPD, iClones, CloneDR, DECKARD, CCFinder	Code clones	NiCad (Text-based), CPD (Token-based), iClones (Token-based), Tree-based), CloneDR, DECKARD,CCFinder (NA)	NiCad (Yes, No, Java), CPD (Yes, No, Java), iClones (Yes, No, Java), CloneDR, DECKARD,CCFinder (NA, NA, NA)

ID	RQ1	RQ2	RQ3	RQ4
[39]	Smell Detector Merger	Brain Class+, Brain Method+, Class Data+, Should be Private, Complex Class+, Data, Class+, Dispersed Coupling+, Duplicate Code, Feature Envy+, God Class+, Intensive Coupling+, Lazy Class+, Long Method+, Long Parameter List+, Message Chain+, Refused Bequest, Shotgun Surgery+, Spaghetti Code+, Speculative Generality+, Tradition Breaker, Type Checking+	Metric-based,	Text-based Yes, No, Java
[37]	HBSniff	Eager Fetch, Lacking Join Fetch, One-By-One, Missing ManyToOne, Collection Field, Final Entity, Missing No Argument Constructor, Missing Identifier, Missing Equals Method, Missing hashCode Method, Using Identifier in Equals or hashCode Methods, Not Serializable, Missing Accessor Methods, Local Pagination	Metric-based,	Tree-based No, No, Java
[71]	ADOCTOR, Paprika (SNIFFER)	NA	ADOCTOR (Rule-based), Paprika (Graph-based, Metric-based)	ADOCTOR (Yes, NA, Java), Paprika (NA, No, Java)
[56]	InCode	God Class+, Data Class+, Feature Envy+, Code Duplication+	Metrics-based	NA, No, Java
[31]	JSNOSE	Empty catch blocks+, Large object+, Lazy object+, Long functions+, Long parameter list+, Switch statements+, Unused/dead code+, Closure smell+, Coupling JS/HTML/CSS+, Excessive Global Variables+, Long Message Chain+, Nested Callback+, Refused Bequest+	Metrics-based	NA, No, Java Script
[28]	BSDT	LargeClass+, LongMethod+, LongParameterList+, SwitchStatements+, ParallelInheritanceHierarchies+Data Class+, Lazy Class+	Metrics-based	No, No, Java
[62]	HIST	Divergent Change, Shotgun Surgery, Parallel Inheritance, Blob, Feature Envy	Rule-based	Yes, NA, Java
[54]	iPlasma	Code Clone	Metrics-based	NA, No, (Java, C++)
[85] [32]	jDeodorant	Feature Envy, State Checking, Type Checking, Long Method, God Classes, Duplicated Code, Refused Bequest	Tree based	Yes, No
[48]	InsRefactor	Data Class+, Large Class+, Long Method+, Switch Statements+, Public Field+, Common Methods in Sibling Classes, Duplicate Code, Long Parameter List+, Duplicate Code	Metrics-based, based	Tree Yes, No, Java
[36]	Smell platform	Long Parameter List+, Feature Envy+, God Class+, Long Method+	Rule-based	Yes, Yes, Java
[63]	ADOCTOR	Data transmission without compression, debuggable release, durable wakelock, inefficient data format and parser, inefficient data structure, inefficient SQL query, internal getter and setter, leaking inner class, leaking thread, member ignoring method, no low memory resolver, public data, rigid alarm manager, slow loop, unclosed closable	Rule-based	SourceMiner
[33]	Arcan	Unstable dependency, hub-like dependency, cyclic dependency	Graph-based	NA, NA, Java
[30]	Asta	Code clones	Token-based	Yes, No, (C#, Java)
[86]	c-JRefRec	Feature Envy	Graph-based	No, No, Java
[45]	CBCD	Code clones	Graph-based	NA, No, C
[42]	CCFinder	Code clones	Tree based	NA, No, (C, C++, Java, COBOL)
[20]	ClemaX	Code clones	Tree based	NA, No, (Java)
[93]	CloneDetection	Code clones	Token-based	Yes, No, (C, C++, Java, PROLOG)
[75]	Coogle (Eclipse plugin)	Code clones	Tree based	No, No, Java
[40]	DECKARD	Code clones	Tree based	Yes, No, Language independent
[57]	DECOR (DETEX)	Blob+, Functional Decomposition+, Spaghetti Code+, Swiss Army Knife+	Rule-based	Yes, Yes, Java
[12]	Dup	Code clones	Token-based	NA, No, C
[61]	JCodeCanine (Eclipse plugin)	Feature Envy+	Metrics-based	NA, No, Java
[15]	JCodeCanine/Fluid	Switch Statement, Data Class, Feature Envy, Dupliacted Code	Three-based	Yes, No, Java
[21]	LCAnalyzer (Eclipse plugin)	Nullable objects, explicit cast, wrong verbosity level, Dup1, Dup2, malformed output	Tree based	NA, No, Java
[50]	SCOOP (Eclipse plugin)	That one that can be defined with domain-specific language	Rule-based	Yes, Yes, Java
[82]	FindSmells	Large class+, Long method+, Data class+, Feature envy+, Refused bequest+	Metrics-based, Rule-based	Yes, Yes, Java
[25]	Gemini	Code clones	Metrics-based	Yes, No, Java
[68]	HULK	God class+, blob+, Swiss Army knife+, spaghetti code+	Rule-based	Yes, NA, Java

ID	RQ1	RQ2	RQ3	RQ4
[90]	jCOSMO	Instanceof, Typecast	Tree based	NA, No, Java
[70]	Jplag	Code clones	Token-based	NA, No, (C, C++, Scheme)
[92]	JSpirit	Brain class+, Brain method+, Data class+, Dispersed coupling+, Feature envy+, God class+, Intensive coupling, Refused parent bequest+, Shotgun surgery+, Tradition breaker+	Metric-based	Yes, NA, Java
[10]	LAPD	Linguistic antipatterns	Text-based	NA, No, Java
[1]	LBSDetectors	Odd grammatical structure, term used to name both the whole and its parts, inconsistent identifier use, useless type indication, identifier construction rules	Text-based	No, NA, C++
[98]	LIME (Eclipse plugin)	Code clones	Token-based	NA, No, C
[74]	NICAD	Code clones	Text-based	Yes, No, C
[78]	PAD	Blob+, Empty Semi Trucks, Circuitous Treasure Hunt	Tree based	Yes, No, Java
[35]	Paprika	Blob+, Swiss Army knife, Complex class+, Long method+, Internal getter/setter, No low memory resolver, Member ignoring method, Leaking inner class	Graph-based, Metric-based	Yes, No, Java
[51]	PC Detector	Code clones	Token-based	NA, No, (C, C++)
[55]	PRODETECTION	God class, Shotgun surgery, Refused bequest, Feature envy, God method, Misplaced class, God package, Wide Subsystem Interface, Data class	Metric-based	Yes, Yes, (Java, C++)
[23] [24]	Pysmell	Large class+, Long parameter list+, Long method+, Long message chain+, Long scope chaining+, Long Base class list+, Useless exception handling+, Long lambda function+, Complex list comprehension+, Long element chain+, Long ternary conditional expression+, Complex Container Comprehension+, Multiply-Nested Container+	Metric-based	Yes, No, Python
[76]	SourcererCC	Code clones	Token-based	Yes, No, (Java, C, C#)
[64]	TACO	Long method, Feature envy, Blob, Promiscuous package, Misplaced class	Text-based	NA, NA, Java
[49]	DuplicationDetector	Code clones	Tree based	NA, NA, UML
[27]	XIAO	Code clones	Text-based	Yes, No, (C, C++, C#)
[95]	DuDe	Code clones	Metric-based	Yes, No, (C, Java)
[60]	Stench Blossom	Feature Envy+, Large Class+, Long Method+, Switch Statement+, Data Clumps+, InstanceOf +, Message Chain+, Typecast+	Rule-based	No, No, Java
[46]	Bortland Together	Data Class, Data Clump, Duplicated Code, Feature Envy, God Class, God Method, Interface Segregation Principle Violation, Misplaced Class, Refused Bequest, Temporary	Metric-based	Yes, NA, Java
[34]	iClones	Code clones	Tree-based	Yes, No, (Java, C++, C, Ada)
[87]	T-CORE	Code clones	Tree based, Graph-based	NA, No, Java
[94]	Code Navigator	God Class+, Brain Class+	Metric-based	Yes, NA, Language independent
[22]	FEED	Feature Envy+	Rule-based	Yes, NA, C#
[19]	SplitLongMethod	Long Method	Metric-based	NA, No, Java
[91]	SpIRIT	Brain class, Brain method, Data class, Disperse coupling, Feature envy, God class, Intensive coupling, Refused parent bequest, Shotgun surgery, Tradition breaker.	Metric-based	NA, No, Java
[97]	CodeVizard	God Class	Metric-based	NA, NA, (Java, C#)
[18]	SourceMiner	God Class, Divergent Change, Feature Envy	Graph-based	Yes, Yes, Java
[53]	Hist-Inspect	NA	Rule-based	Yes, Yes, NA
[16]	DV8	Package Cycles, Unstable Interface, Unhealthy Inheritance Hierarchy, Modularity Violations, Crossing, Clique	Rule-based	Na, NA, NA
[43]	RefactoringAssistant	NA	Metric-based	Na, NA, NA
[80]	Columbus framework	LargeMethod, Long Parameter List, Large Class, Temporary Field, Shotgun Survey, Lazy Class, Data Class, Speculative Generality, Middle Man, Feature Envy, Inappropriate Intimacy	Metric-based	NA, NA, NA
[38]	DART	Small method, Lazy Class, Duplicate code	Text-based, Metric-based	No, No, Java
[72]	Multiple language smells detector	Lazy class, Temporary field, Duplicated Code, Message Chain, Refused bequest, Large Class, Data Class, Feature envy, Inappropriate intimacy, Long parameter list, Shotgun surgery, Divergent change, Middle man, Long method, Parallel inheritance hierarchy, Switch statements, Speculative generality	Metric-based, Tree-based	Yes, NA, (Java, C#)

ID	RQ1	RQ2	RQ3	RQ4
[58]	2D-DSL	Blob, Functional Decomposition, Spaghetti Code, Swiss Army Knife, Large Class, Lazy Class, Long Method, Long Parameter List, Refused Parent Bequest, Speculative Generality	Rule-based	Yes, Yes, Java
[66]	Noseprints	Large Class, Feature Envy, Long Method, Data Class, Refused Bequest, Long Parameter List, Message Chains Class, Data Clumps, Middle Man, Primitive Obsession	Graph-based	Yes, NA, Java
[13]	MethodBook	Feature Envy	Metric-based	No, No, Java
[41]	OBEY	Middle Man	Rule-based	No, No, Java
[17]	Hot-Papper	HashMap Usage, Internal Getter/Setter, Member Ignoring Method, Picture Format, Picture Size, Picture Bitmap Usage	Graph-based, Metric-based	Yes, No, Java
[73]	Jsmell	Data Class, Message Chain, Primitive Obsession, Speculative Generality, Parallel Inheritance Hierarchy, Duplicated Code, Comment	Metric-based	NA, NA, Java
[83]	SYMake	Cyclic Dependency, Duplicated Prerequisites	Metric-based	NA, NA, C
[9]	VCS-Analyzer	"God Class, Data Class, Brain Method, Shotgun Surgery, Dispersed Coupling, Message Chain"	Metric-based	NA, NA, Java
[6]	ConcernReCS	Primitive Concern Constant, Static Concern Element, Attribute of a Nondedicated Type, Conditional Use of Local Variable, Divergent Join Point, Element out of Inheritance Tree	Rule-based	NA, NA, Java

REFERENCES

- [1] Surafel Lemma Abebe, Sonia Haiduc, Paolo Tonella, and Andrian Marcus. Lexicon bad smells in software. In *2009 16th Working Conference on Reverse Engineering*, pages 95–99, 2009.
- [2] Amjad AbuHassan, Mohammad Alshayeb, and Lahouari Ghouti. Software smell detection techniques: A systematic literature review. *Journal of Software: Evolution and Process*, 33(3):e2320, 2021. e2320 JSME-19-0205.R3.
- [3] Sara Seif Afjehei, Tse-Hsun Peter Chen, and Nikolaos Tsantalis. iperfdetector: Characterizing and detecting performance anti-patterns in ios applications. *Empirical Software Engineering*, pages 1–30, 2019.
- [4] Khalid Alkharabsheh, Yania Crespo, Esperanza Manso, and José A. Taboada. Software design smell detection: a systematic mapping study. *Software Quality Journal*, pages 1–80, 2018.
- [5] Nabil Almashfi and Lunjin Lu. Code smell detection tool for java script programs. In *2020 5th International Conference on Computer and Communication Systems (ICCCS)*, pages 172–176, 2020.
- [6] Pericles Alves, Diogo Santana, and Eduardo Figueiredo. Concernrecs: Finding code smells in software aspectization. *Proceedings - International Conference on Software Engineering*, pages 1463–1464, 06 2012.
- [7] Apostolos Ampatzoglou, Angeliki-Agathi Tsintzira, Elvira-Maria Arvanitou, Alexander Chatzigeorgiou, Ioannis Stamelos, Alexandru Moga, Robert Heb, Oliviu Matei, Nikolaos Tsiridis, and Dionisis Kehagias. Applying the single responsibility principle in industry: Modularity benefits and trade-offs. In *Proceedings of the Evaluation and Assessment on Software Engineering*, EASE '19, page 347–352, New York, NY, USA, 2019. Association for Computing Machinery.
- [8] Hina Anwar, Dietmar Pfahl, and Satish N. Srirama. Evaluating the impact of code smell refactoring on the energy consumption of android applications. In *2019 45th Euromicro Conference on Software Engineering and Advanced Applications (SEAA)*, pages 82–86, 2019.
- [9] Francesca Arcelli, Matteo Rolla, and Marco Zanon. Vcs-analyzer for software evolution empirical analysis. In *Proceedings of the 8th ACM/IEEE International Symposium on Empirical Software Engineering and Measurement*, ESEM '14, New York, NY, USA, 2014. Association for Computing Machinery.
- [10] Venera Amaoudova, Massimiliano Di Penta, Giuliano Antoniol, and Yann-Gaël Guéhéneuc. A new family of software anti-patterns: Linguistic anti-patterns. In *2013 17th European Conference on Software Maintenance and Reengineering*, pages 187–196, 2013.
- [11] Bahareh Bafandeh Mayvan, Abbas Rasoolzadegan, and Abbas Javan Jafari. Bad smell detection using quality metrics and refactoring opportunities. *Journal of Software: Evolution and Process*, 32(8):e2255, 2020. e2255 JSME-19-0162.R2.
- [12] Brenda S Baker. A program for identifying duplicated code. *Computing Science and Statistics*, pages 49–49, 1993.
- [13] Gabriele Bavota, Rocco Oliveto, Malcom Gethers, Denys Poshyvanyk, and Andrea De Lucia. Methodbook: Recommending move method refactorings via relational topic models. *IEEE Transactions on Software Engineering*, 40(7):671–694, 2014.
- [14] Mariza A.S. Bigonha, Kecia Ferreira, Priscila Souza, Bruno Sousa, Marcela Januário, and Daniele Lima. The usefulness of software metric thresholds for detection of bad smells and fault prediction. *Information and Software Technology*, 115:79–92, 2019.
- [15] John Tang Boyland and Kwankamol Nongpong. Integrating code smells detection with refactoring tool support. 2012.
- [16] Yuanfang Cai and Rick Kazman. Dv8: Automated architecture analysis tool suites. In *2019 IEEE/ACM International Conference on Technical Debt (TechDebt)*, pages 53–54, 2019.
- [17] Antonin Carette, Mehdi Adel Ait Younes, Geoffrey Hecht, Naouel Moha, and Romain Rouvoy. Investigating the energy impact of android smells. In *2017 IEEE 24th International Conference on Software Analysis, Evolution and Reengineering (SANER)*, pages 115–126, 2017.
- [18] Glauco Carneiro and Manoel Mendonça. Sourceminor - a multi-perspective software visualization environment. volume 2, 07 2013.
- [19] Sofia Charalampidou, Apostolos Ampatzoglou, Alexander Chatzigeorgiou, Antonios Gkortsis, and Paris Avgeriou. Identifying extract method refactoring opportunities based on functional relevance. *IEEE Transactions on Software Engineering*, 43(10):954–974, 2017.
- [20] Marsha Chechik and Martin Wirsing, editors. *Fundamental Approaches to Software Engineering*. Springer Berlin Heidelberg, 2009.
- [21] Boyuan Chen and Zhen Ming Jiang. Characterizing and detecting anti-patterns in the logging code. In *2017 IEEE/ACM 39th International Conference on Software Engineering (ICSE)*, pages 71–81, 2017.
- [22] Woei-Kae Chen, Chien-Hung Liu, and Bo-Hong Li. A feature envy detection method based on dataflow analysis. In *2018 IEEE 42nd Annual Computer Software and Applications Conference (COMPSAC)*, volume 02, pages 14–19, 2018.
- [23] Zhifei Chen, Lin Chen, Wanwangying Ma, and Baowen Xu. Detecting code smells in python programs. In *2016 International Conference on Software Analysis, Testing and Evolution (SATE)*, pages 18–23, 2016.
- [24] Zhifei Chen, Lin Chen, Wanwangying Ma, Xiaoyu Zhou, Yuming Zhou, and Baowen Xu. Understanding metric-based detectable smells in python software: A comparative study. *Information and Software Technology*, 94:14–29, 2018.
- [25] Eunjong Choi, Norihiro Yoshida, Takashi Ishio, Katsuro Inoue, and Tateki Sano. Extracting code clones for refactoring using combinations of clone metrics. In *Proceedings of the 5th International Workshop on Software Clones*, IWSC '11, page 7–13, New York, NY, USA, 2011. Association for Computing Machinery.
- [26] Renieri Correia and Eiji Adachi. Detecting design violations in django-based web applications. In *Proceedings of the XIII Brazilian Symposium on Software Components, Architectures, and Reuse*, SBCARS '19, page 33–42, New York, NY, USA, 2019. Association for Computing Machinery.
- [27] Yingnong Dang, Dongmei Zhang, Song Ge, Ray Huang, Chengyun Chu, and Tao Xie. Transferring code-clone detection and analysis to practice. In *2017 IEEE/ACM 39th International Conference on Software Engineering: Software Engineering in Practice Track (ICSE-SEIP)*, pages 53–62, 2017.
- [28] Phongphan Danphitsanuphan and Thanitta Suwantada. Code smell detecting tool and code smell-structure bug relationship. In *2012 Spring Congress on Engineering and Technology*, pages 1–5, 2012.
- [29] Manuel De Stefano, Michele Simone Gambardella, Fabiano Pecorelli, Fabio Palomba, and Andrea De Lucia. Casper: A plug-in for automated code smell detection and refactoring. In *Proceedings of the International Conference on Advanced Visual Interfaces*, AVI '20, New York, NY, USA, 2020. Association for Computing Machinery.
- [30] William S. Evans, Christopher W. Fraser, and Fei Ma. Clone detection via structural abstraction. 17(4):309–330, 2009.
- [31] Amin Milani Fard and Ali Mesbah. Jsnoze: Detecting javascript code smells. In *2013 IEEE 13th International Working Conference on Source Code Analysis and Manipulation (SCAM)*, pages 116–125, 2013.
- [32] Marios Fokaefs, Nikolaos Tsantalis, Eleni Stroulia, and Alexander Chatzigeorgiou. Jdeodorant: identification and application of extract class refactorings. In *2011 33rd International Conference on Software Engineering (ICSE)*, pages 1037–1039, 2011.
- [33] Francesca Arcelli Fontana, Ilaria Pigazzini, Riccardo Roveda, and Marco Zanon. Automatic detection of instability architectural smells. In *2016 IEEE International Conference on Software Maintenance and Evolution (ICSME)*, pages 433–437, 2016.
- [34] Nils Göde and Rainer Koschke. Incremental clone detection. In *2009 13th European Conference on Software Maintenance and Reengineering*, pages 219–228, 2009.
- [35] Geoffrey Hecht, Romain Rouvoy, Naouel Moha, and Laurence Duchien. Detecting antipatterns in android apps. In *Proceedings of the Second ACM International Conference on Mobile Software Engineering and Systems*, MOBILESoft '15, page 148–149. IEEE Press, 2015.
- [36] Mario Hozano, Henrique Ferreira, Italo Silva, Baldoino Fonseca, and Evandro Costa. Using developers' feedback to improve code smell detection. In *Proceedings of the 30th Annual ACM Symposium on Applied Computing*, SAC '15, page 1661–1663, New York, NY, USA, 2015. Association for Computing Machinery.
- [37] Zijie Huang, Zhiqing Shao, Guisheng Fan, Huiqun Yu, Kang Yang, and Ziyi Zhou. Hbsniff: A static analysis tool for java hibernate object-relational mapping code smell detection. *Science of Computer Programming*, 217:102778, 2022.
- [38] Rosziati Ibrahim, Maryam Ahmed, Richi Nayak, and Sapiee Jamel. Reducing redundancy of test cases generation using code smell detection and refactoring. *Journal of King Saud University - Computer and Information Sciences*, 32, 06 2018.
- [39] Apostolos Ichtisis, Nikolaos Mittas, Apostolos Ampatzoglou, and Alexander Chatzigeorgiou. Merging smell detectors: Evidence on the agreement of multiple tools. 05 2022.
- [40] Lingxiao Jiang, Ghassan Misherghi, Zhendong Su, and Stephane Glondu. Deckard: Scalable and accurate tree-based detection of code clones. In *29th International Conference on Software Engineering (ICSE'07)*, pages 96–105, 2007.
- [41] Hewijin Christine Jiau, Lee Wei Mar, and Jinghong Cox Chen. Obey: Optimal batched refactoring plan execution for class responsibility

- redistribution. *IEEE Transactions on Software Engineering*, 39(9):1245–1263, 2013.
- [42] T. Kamiya, S. Kusumoto, and K. Inoue. Ccfinder: a multilinguistic token-based code clone detection system for large scale source code. *IEEE Transactions on Software Engineering*, 28(7):654–670, 2002.
- [43] Y. Kataoka, T. Imai, H. Andou, and T. Fukaya. A quantitative evaluation of maintainability enhancement by refactoring. In *International Conference on Software Maintenance, 2002. Proceedings.*, pages 576–585, 2002.
- [44] Vaishali Khanve. Are existing code smells relevant in web games? an empirical study. pages 1241–1243, 08 2019.
- [45] Jingyue Li and Michael D. Ernst. Cbcd: Cloned buggy code detector. In *2012 34th International Conference on Software Engineering (ICSE)*, pages 310–320, 2012.
- [46] Wei Li and Raed Shatnawi. An empirical study of the bad smells and class error probability in the post-release object-oriented system evolution. *Journal of Systems and Software*, 80:1120–1128, 07 2007.
- [47] Zhenhao Li, Tse-Hsun Chen, Jinqu Yang, and Weiyi Shang. Studying duplicate logging statements and their relationships with code clones. *IEEE Transactions on Software Engineering*, 48(7):2476–2494, 2022.
- [48] Hui Liu, Xue Guo, and Weizhong Shao. Monitor-based instant software refactoring. *IEEE Transactions on Software Engineering*, 39(8):1112–1126, 2013.
- [49] Hui Liu, Zhiyi Ma, Lu Zhang, and Weizhong Shao. Detecting duplications in sequence diagrams based on suffix trees. *2006 13th Asia Pacific Software Engineering Conference (APSEC’06)*, pages 269–276, 2006.
- [50] Isela Macia, Roberta Arcoverde, Elder Cirilo, Alessandro Garcia, and Arndt von Staa. Supporting the identification of architecturally-relevant code anomalies. In *2012 28th IEEE International Conference on Software Maintenance (ICSM)*, pages 662–665, 2012.
- [51] Ginika Mahajan and Meena Bharti. Implementing a 3-way approach of clone detection and removal using pc detector tool. In *2014 IEEE International Advance Computing Conference (IACC)*, pages 1435–1441, 2014.
- [52] Chenguang Mao, Hao Wang, Gaojie Han, and Xiaofang Zhang. Droidlens: Robust and fine-grained detection for android code smells. In *2020 International Symposium on Theoretical Aspects of Software Engineering (TASE)*, pages 161–168, 2020.
- [53] Leandra Mara, Gustavo Honorato, Francisco Dantas Medeiros, Alessandro Garcia, and Carlos Lucena. Hist-inspect: A tool for history-sensitive detection of code smells. *AOSD ’11*, page 65–66, New York, NY, USA, 2011. Association for Computing Machinery.
- [54] Cristina Marinescu, Radu Marinescu, Petru Florin Mihancea, Daniel Ratiu, and Richard Wetzel. iplasma: An integrated platform for quality assessment of object-oriented design. In *International Conference on Smart Multimedia*, 2005.
- [55] R. Marinescu. Detection strategies: metrics-based rules for detecting design flaws. In *20th IEEE International Conference on Software Maintenance, 2004. Proceedings.*, pages 350–359, 2004.
- [56] Radu Marinescu, George Ganeva, and Ioana Verebi. Incode: Continuous quality assessment and improvement. In *2010 14th European Conference on Software Maintenance and Reengineering*, pages 274–275, 2010.
- [57] Naouel Moha, Yann-Gaël Guéhéneuc, Laurence Duchien, and Anne-Françoise Le Meur. Decor: A method for the specification and detection of code and design smells. *IEEE Transactions on Software Engineering*, 36(1):20–36, 2010.
- [58] Naouel Moha, Yann-Gaël Guéhéneuc, Anne-Françoise Meur, Laurence Duchien, and Alban Tiberghien. From a domain analysis to the specification and detection of code and design smells. *Formal Aspects of Computing*, 22, 05 2010.
- [59] Haris Mumtaz, Paramvir Singh, and Kelly Blincoe. A systematic mapping study on architectural smells detection. *Journal of Systems and Software*, 173:110885, 2021.
- [60] Emerson Murphy-Hill and Andrew P. Black. An interactive ambient visualization for code smells. In *Proceedings of the 5th International Symposium on Software Visualization, SOFTVIS ’10*, page 5–14, New York, NY, USA, 2010. Association for Computing Machinery.
- [61] Kwankamol Nongpong. Feature envy factor: A metric for automatic feature envy detection. In *2015 7th International Conference on Knowledge and Smart Technology (KST)*, pages 7–12, 2015.
- [62] Fabio Palomba, Gabriele Bavota, Massimiliano Di Penta, Rocco Oliveto, Andrea De Lucia, and Denys Poshyvanyk. Detecting bad smells in source code using change history information. In *2013 28th IEEE/ACM International Conference on Automated Software Engineering (ASE)*, pages 268–278, 2013.
- [63] Fabio Palomba, Dario Di Nucci, Annibale Panichella, Andy Zaidman, and Andrea De Lucia. Lightweight detection of android-specific code smells: The adocor project. In *2017 IEEE 24th International Conference on Software Analysis, Evolution and Reengineering (SANER)*, pages 487–491, 2017.
- [64] Fabio Palomba, Annibale Panichella, Andrea De Lucia, Rocco Oliveto, and Andy Zaidman. A textual-based technique for smell detection. In *2016 IEEE 24th International Conference on Program Comprehension (ICPC)*, pages 1–10, 2016.
- [65] Ranindya Paramitha and Yudistira Dwi Wardhana Asnar. Mining software repository for security smell code review. In *2021 International Conference on Data and Software Engineering (ICoDSE)*, pages 1–6, 2021.
- [66] Chris Parnin, Carsten Görg, and Ogechi Nnadi. A catalogue of lightweight visualizations to support code smell inspection. In *Proceedings of the 4th ACM Symposium on Software Visualization, SoftVis ’08*, page 77–86, New York, NY, USA, 2008. Association for Computing Machinery.
- [67] Fabiano Pecorelli, Fabio Palomba, Dario Di Nucci, and Andrea De Lucia. Comparing heuristic and machine learning approaches for metric-based code smell detection. In *2019 IEEE/ACM 27th International Conference on Program Comprehension (ICPC)*, pages 93–104, 2019.
- [68] Sven Peldszus, Géza Kulcsár, Malte Lochau, and Sandro Schulze. Continuous detection of design flaws in evolving object-oriented programs using incremental multi-pattern matching. In *2016 31st IEEE/ACM International Conference on Automated Software Engineering (ASE)*, pages 578–589, 2016.
- [69] Ilaria Pigazzini, Francesca Arcelli Fontana, and Andrea Maggioni. Tool support for the migration to microservice architecture: An industrial case study. In *European Conference on Software Architecture*, 2019.
- [70] Lutz Prechelt and Guido Malpohl. Finding plagiarisms among a set of programs with jplag. *Journal of Universal Computer Science*, 8, 03 2003.
- [71] Dimitri Prestat, Naouel Moha, and Roger Villemair. An empirical study of android behavioural code smells detection. *Empirical Software Engineering*, 27, 09 2022.
- [72] Ghulam Rasool and Zeeshan Arshad. A lightweight approach for detection of code smells. *Arabian Journal for Science and Engineering*, 42, 07 2016.
- [73] Naveen Roperia. Jsmell: A bad smell detection tool for java systems. 2009.
- [74] Chanchal K. Roy and James R. Cordy. Nicad: Accurate detection of near-miss intentional clones using flexible pretty-printing and code normalization. In *2008 16th IEEE International Conference on Program Comprehension*, pages 172–181, 2008.
- [75] Tobias Sager, Abraham Bernstein, Martin Pinzger, and Christoph Kiefer. Detecting similar java classes using tree algorithms. In *Proceedings of the 2006 international workshop on Mining software repositories*. ACM, May 2006.
- [76] Hitesh Sajjani, Vaibhav Saini, Jeffrey Svajlenko, Chanchal K. Roy, and Cristina V. Lopes. Sourcerercc: Scaling code clone detection to big-code. In *2016 IEEE/ACM 38th International Conference on Software Engineering (ICSE)*, pages 1157–1168, 2016.
- [77] Tushar Sharma, Paramvir Singh, and Diomidis Spinellis. An empirical investigation on the relationship between design and architecture smells. 07 2020.
- [78] Vibhu Saujanya Sharma and Samit Anwer. Performance antipatterns: Detection and evaluation of their effects in the cloud. In *2014 IEEE International Conference on Services Computing*, pages 758–765, 2014.
- [79] Bo Shen, Wei Zhang, Ailun Yu, Zhao Wei, Guangtai Liang, Haiyan Zhao, and Zhi Jin. Cross-language code coupling detection: A preliminary study on android applications. In *2021 IEEE International Conference on Software Maintenance and Evolution (ICSME)*, pages 378–388, 2021.
- [80] Satwinder Singh and Karanjeet Kahlon. Effectiveness of encapsulation and object-oriented metrics to refactor code and identify error prone classes using bad smells. *ACM SIGSOFT Software Engineering Notes*, 36:1–10, 09 2011.
- [81] Elder Vicente de Paulo Sobrinho, Andrea De Lucia, and Marcelo de Almeida Maia. A systematic literature review on bad smells–5 w’s: Which, when, what, who, where. *IEEE Transactions on Software Engineering*, 47(1):17–66, 2021.
- [82] Bruno L. Sousa, Priscila P. Souza, Eduardo M. Fernandes, Kécia A.M. Ferreira, and Mariza A.S. Bigonha. Findsmells: Flexible composition of bad smell detection strategies. In *2017 IEEE/ACM 25th International Conference on Program Comprehension (ICPC)*, pages 360–363, 2017.
- [83] Ahmed Tamrawi, Hoan Anh Nguyen, Hung Viet Nguyen, and Tien N. Nguyen. Symake: a build code analysis and refactoring tool for

- makefiles. In *2012 Proceedings of the 27th IEEE/ACM International Conference on Automated Software Engineering*, pages 366–369, 2012.
- [84] Omkarendra Tiwari and Rushikesh K. Joshi. Functionality based code smell detection and severity classification. In *Proceedings of the 13th Innovations in Software Engineering Conference on Formerly Known as India Software Engineering Conference*, ISEC 2020, New York, NY, USA, 2020. Association for Computing Machinery.
- [85] Nikolaos Tsantalis, Theodoros Chaikalas, and Alexander Chatzigeorgiou. Ten years of jdeodorant: Lessons learned from the hunt for smells. In *2018 IEEE 25th International Conference on Software Analysis, Evolution and Reengineering (SANER)*, pages 4–14, 2018.
- [86] Naoya Ujihara, Ali Ouni, Takashi Ishio, and Katsuro Inoue. c-jrefrec: Change-based identification of move method refactoring opportunities. In *2017 IEEE 24th International Conference on Software Analysis, Evolution and Reengineering (SANER)*, pages 482–486, 2017.
- [87] Brent van Bladel and Serge Demeyer. A novel approach for detecting type-iv clones in test code. In *2019 IEEE 13th International Workshop on Software Clones (IWSC)*, pages 8–12, 2019.
- [88] Brent van Bladel and Serge Demeyer. Clone detection in test code: An empirical evaluation. In *2020 IEEE 27th International Conference on Software Analysis, Evolution and Reengineering (SANER)*, pages 492–500, 2020.
- [89] Brent van Bladel and Serge Demeyer. A comparative study of test code clones and production code clones. *Journal of Systems and Software*, 176:110940, 03 2021.
- [90] E. van Emden and L. Moonen. Java quality assurance by detecting code smells. In *Ninth Working Conference on Reverse Engineering, 2002. Proceedings.*, pages 97–106, 2002.
- [91] Santiago Vidal, Claudia Marcos, and Andres Diaz-Pace. An approach to prioritize code smells for refactoring. *Automated Software Engineering*, 23, 12 2014.
- [92] Santiago Vidal, Hernan Vazquez, J. Andres Diaz-Pace, Claudia Marcos, Alessandro Garcia, and Willian Oizumi. Jspirit: a flexible tool for the analysis of code smells. In *2015 34th International Conference of the Chilean Computer Science Society (SCCC)*, pages 1–6, 2015.
- [93] V. Wahler, D. Seipel, J. Wolff, and G. Fischer. Clone detection in source code by frequent itemset techniques. In *Source Code Analysis and Manipulation, Fourth IEEE International Workshop on*, pages 128–135, 2004.
- [94] Bartosz Walter, Błażej Matuszyk, and Francesca Arcelli Fontana. Including structural factors into the metrics-based code smells detection. In *Scientific Workshop Proceedings of the XP2015, XP '15 workshops*, New York, NY, USA, 2015. Association for Computing Machinery.
- [95] R. Wetzel and R. Marinescu. Archeology of code duplication: recovering duplication chains from small duplication fragments. In *Seventh International Symposium on Symbolic and Numeric Algorithms for Scientific Computing (SYNASC'05)*, pages 8 pp.–, 2005.
- [96] Rafed Muhammad Yasir, Moumita Asad, Asadullah Hill Galib, Kishan Kumar Ganguly, and Md Saeed Siddik. Godexpo: An automated god structure detection tool for go lang. In *2019 IEEE/ACM 3rd International Workshop on Refactoring (IWor)*, pages 47–50, 2019.
- [97] Nico Zazworka and Christopher Ackermann. Codevizard: a tool to aid the analysis of software evolution. In *International Symposium on Empirical Software Engineering and Measurement*, 2010.
- [98] Minhaz F. Zibran and Chanchal K. Roy. Towards flexible code clone detection, management, and refactoring in ide. In *Proceedings of the 5th International Workshop on Software Clones, IWSC '11*, page 75–76, New York, NY, USA, 2011. Association for Computing Machinery.