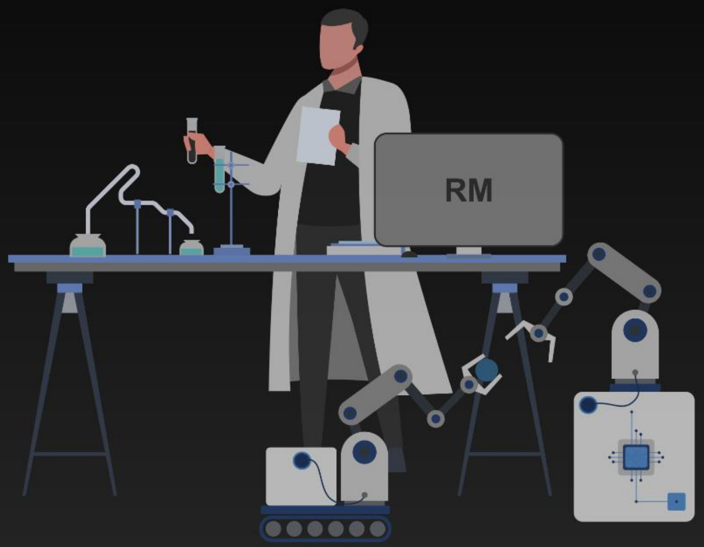
RESEARCH METHODOLOGY

Romi Satria Wahono

romi@romisatriawahono.net
http://romisatriawahono.net
08118228331

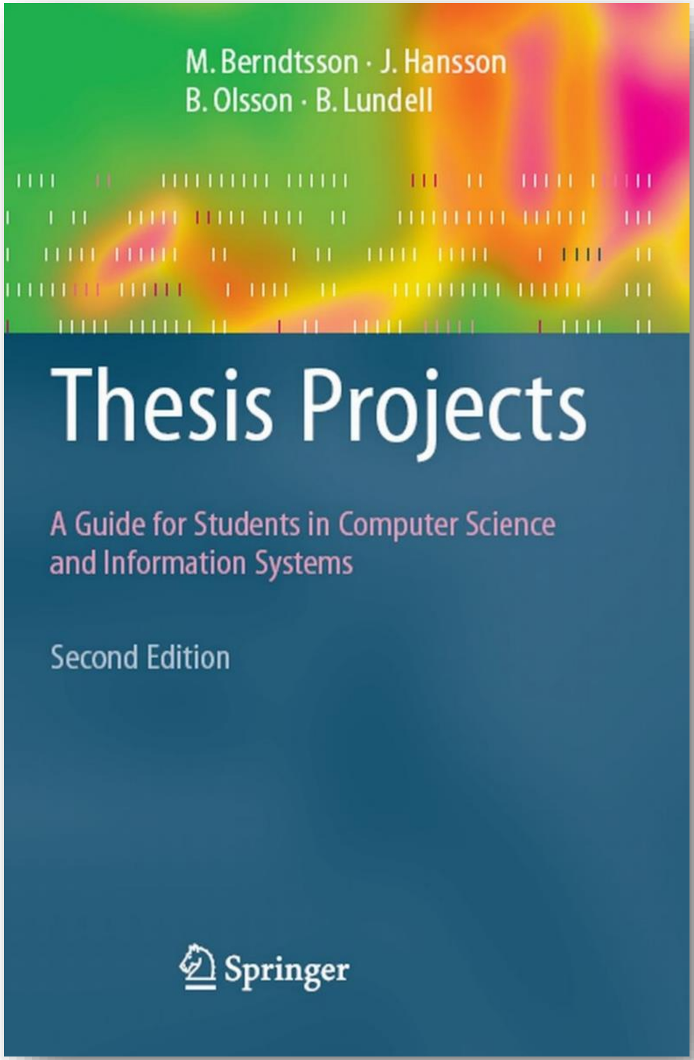


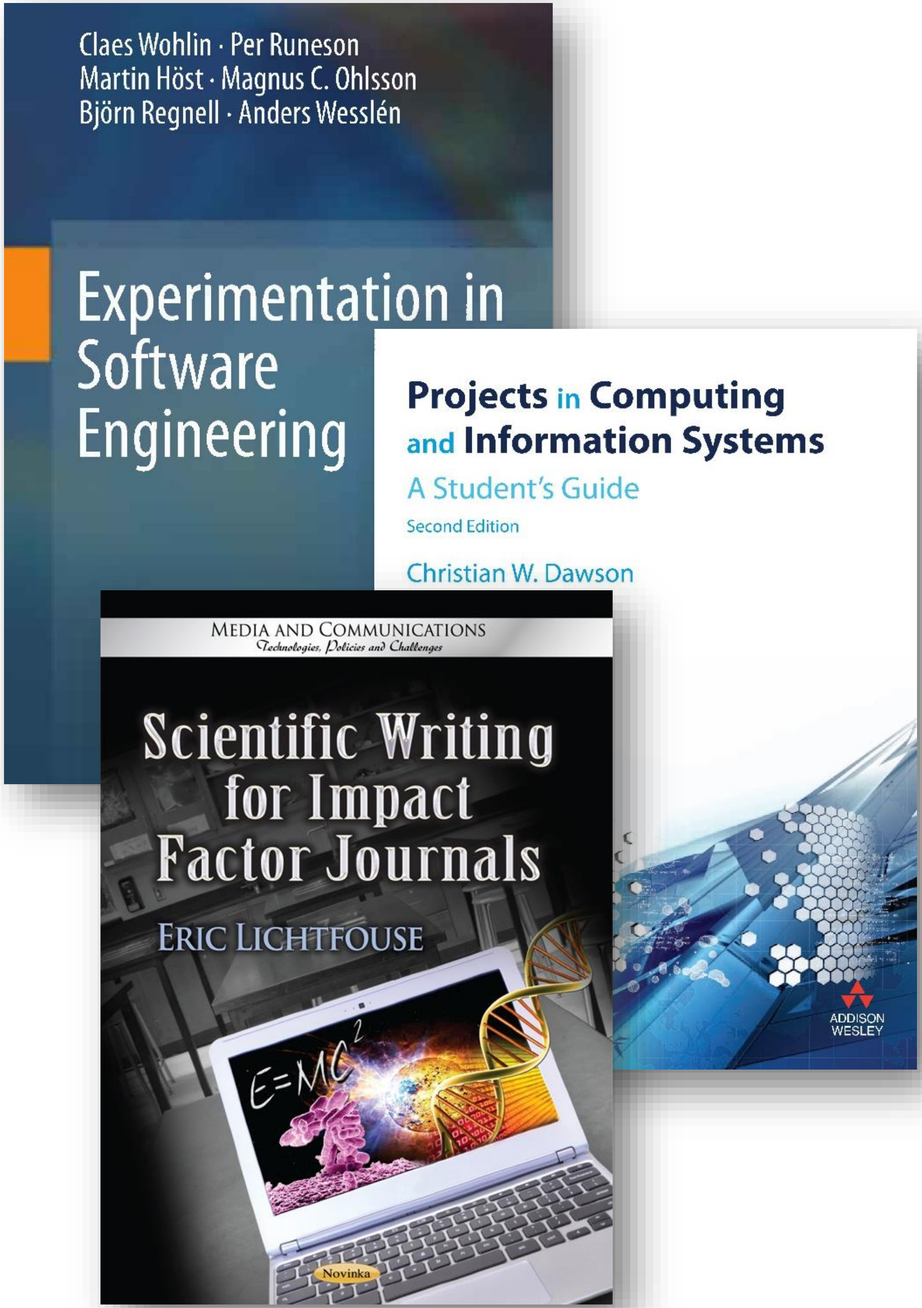
Romi Satria Wahono

- SMA Taruna Nusantara Magelang (1993)
- B.Eng, M.Eng and Ph.D in Software Engineering
  Saitama University Japan (1994-2004)
  Universiti Teknikal Malaysia Melaka (2014)
- Core Competency in Enterprise Architecture, Software Engineering and Machine Learning
- LIPI Researcher (2004-2007)
- Founder and CEO:
  - PT Brainmatics Cipta Informatika (2005)
  - PT IlmuKomputerCom Braindevs Sistema (2014)
- Professional Member of IEEE, ACM and PMI
- IT and Research Award Winners from WSIS (United Nations), Kemdikbud, Ristekdikti, LIPI, etc
- SCOPUS/ISI Indexed Journal Reviewer: Information and Software Technology, Journal of Systems and Software, Software: Practice and Experience, etc
- Industrial IT Certifications: TOGAF, ITIL, CCAI, CCNA, etc
- Enterprise Architecture Consultant: KPK, RistekDikti, INSW, BPPT, Kemsos Kemenkeu (Itjend, DJBC, DJPK), Telkom, FIF, PLN, PJB, Pertamina EP, etc


RomiSatriaWahono.Net
Lecture Notes in Software Engineering, Computing Research and Technopreneurship
HOME ABOUT ME IN THE NEWS LECTURES PUBLICATIONS RESEARCH
RESEARCH 18 DEC 2014
BAGAIMANA MELAKUKAN PENELITIAN YANG BAIK?
Home » Lectures
i universitas di Indonesia. Seluruh materi kuliah bisa diunduh dan
nan mata kuliah memuat course description, standard competency,
ook yang digunakan.
Research Methodology (updated January 2015)
Data Mining (updated January 2015)
Theory of Computation (updated March 2015)
Java Fundamentals (updated October 2013)
Java Enterprise Edition
Systems Analysis and Design (updated January 2015)
Business Process Model and Notation (updated January 2015)
Software Engineering
Software Testing
Software Quality Assurance
Project Management
TOGAF 9.1 Fundamental
TOGAF 9.1 Foundation
TOGAF 9.1 Certified
YouTube
Upload
189 subscribers
4,535 views
Video Manager
View as public
Romi Satria Wahono
Home
Videos
Playlists
Channels
Discussion
About
Uploads
Date added (newest - oldest)
5:30
Menjadi Programmer Technopreneur
116 views • 3 weeks ago
13:37
Kuliah 10 Menit tentang Enterprise Architecture
816 views • 3 weeks ago
WATCHED
18:42
Kuliah 20 Menit tentang Metodologi Penelitian
1,756 views • 4 weeks ago
11:33
Kuliah 10 Menit tentang Data Mining
1,898 views • 1 month ago

Textbooks


M. Berndtsson · J. Hansson
B. Olsson · B. Lundell
Thesis Projects
A Guide for Students in Computer Science and Information Systems
Second Edition
Springer


Claes Wohlin · Per Runeson
Martin Höst · Magnus C. Ohlsson
Björn Regnell · Anders Wesslén
Experimentation in Software Engineering
Projects in Computing and Information Systems
A Student's Guide
Second Edition
Christian W. Dawson
ADDISON WESLEY
Media and Communications
Technologies, Policies and Challenges
Scientific Writing for Impact Factor Journals
Eric Lichtfouse
Novinka

Course Outline

1. Pengantar Penelitian

- 1.1 Definisi Penelitian
- 1.2 Klasifikasi Penelitian
- 1.3 Gaya Penelitian Computing
- 1.4 Kontribusi dan Orisinalitas

2. Tahapan Penelitian

- 2.1 Tahapan Penelitian Umum
- 2.2 Tahapan Penelitian Computing
- 2.3 Tahapan Penelitian Computing Fokus Perbaikan Algoritma

3. Literature Review

- 3.1 Literatur Ilmiah
- 3.2 Teknik Mengelola Paper
- 3.3 Teknik Mereview Paper

4. Penulisan Ilmiah

- 4.1 Mengapa Penulisan dan Publikasi Ilmiah?
- 4.2 Sitasi dan Penulisan Referensi
- 4.3 Penulisan Tesis
- 4.4 Publikasi Ilmiah untuk Jurnal Internasional
- 4.5 Penulisan Systematic Literature Review (SLR)

5. Pembimbingan dan Presentasi Penelitian

- 5.1 Pembimbingan Penelitian
- 5.2 Presentasi Penelitian

1. Pengantar Penelitian

1.1 Definisi Penelitian

1.2 Klasifikasi Penelitian

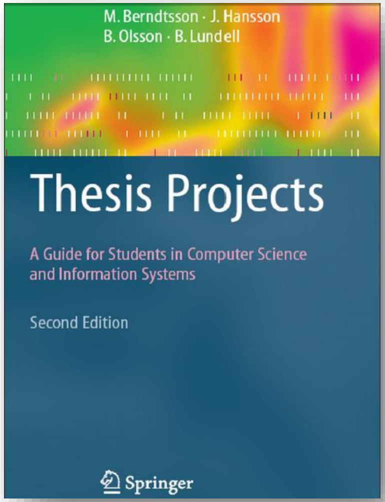
1.3 Gaya Penelitian Bidang Computing

1.4 Kontribusi dan Orisinalitas

1.1 Definisi Penelitian

Mengapa Melakukan Penelitian?

- Berangkat dari adanya masalah penelitian
  - yang mungkin sudah diketahui metode pemecahannya
  - tapi belum diketahui metode pemecahan yang lebih baik



- Research (Inggris) dan recherche (Prancis)
  - re (kembali)
  - to search (mencari)

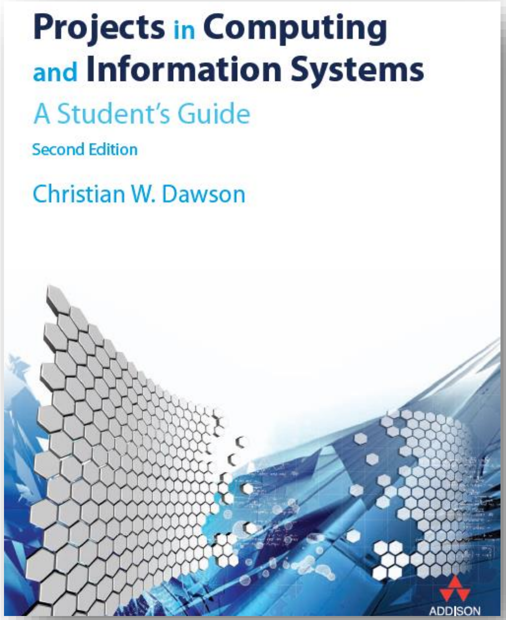
- The process of exploring the unknown, studying and learning new things, building new knowledge about things that no one has understood before *(Berndtsson et al., 2008)*

Apa Yang Dikejar di Penelitian?

Research is a considered activity, which aims to make an original contribution to knowledge

(contribution to the body of knowledge, in the research field of interest)

(Dawson, 2009)



Bentuk Kontribusi ke Pengetahuan

Kegiatan penyelidikan dan investigasi terhadap suatu masalah yang dilakukan secara berulang-ulang dan sistematis, dengan tujuan untuk menemukan atau merevisi teori, metode, fakta, dan aplikasi

(Berndtsson et al., 2008)

Kontribusi ke Pengetahuan vs Kontribusi ke Masyarakat

Kontribusi ke masyarakat tidak secara langsung bisa diukur, karena itu tidak dimasukkan ke tujuan penelitian, tapi ke manfaat penelitian

Apa itu Penelitian?

- Penelitian dilakukan karena ada masalah penelitian, dimana masalah penelitian sendiri muncul karena adanya latar belakang masalah, yang terlahir dari masalah kehidupan
- Penelitian dilakukan secara terencana, sistematis, berulang-ulang dan terukur
- Penelitian harus memiliki orisinalitas (*originality*) dan kebaruan (*novelty*), serta menghasilkan kontribusi yang orisinil pada pengetahuan dalam bentuk menemukan atau merevisi teori, metode, fakta, dan aplikasi

Pengembangan Software vs Penelitian

- Membangun software bukanlah tujuan utama penelitian, hanya *testbed* untuk mempermudah kita dalam mengukur hasil penelitian
  - Tidak ada listing code, UML atau screenshot software di paper-paper journal (SCOPUS/WoS), kecuali penelitian tentang perbaikan paradigma pemrograman, analisis design, dsb

- Ketika pada penelitian kita mengusulkan perbaikan suatu algoritma (*proposed method*)
  - Bidang image processing, topik penelitian face recognition, memikirkan perbaikan metode/algoritma untuk pengenalan wajah dengan akurat/efisien
  - Bidang data mining, topik decision tree, memikirkan perbaikan algoritma decision tree sehingga bisa memprediksi (klasifikasi) dengan lebih akurat
  - Untuk mempermudah eksperimen dan evaluasi, kita menulis kode program (software) untuk menguji dan mengevaluasi performance dari algoritma yang kita usulkan

Karakter Peneliti

- Peneliti itu boleh salah
  - salah hipotesis
  - salah analisis
  - salah pengujian hipotesis
  - dsb
- Tapi peneliti tidak boleh bohong atau menipu
  - mempermainkan data
  - manipulasi hasil pengolahan statistik
  - dsb

1.2 Klasifikasi Penelitian

Klasifikasi Penelitian

1. Pendekatan

1. Pendekatan Kualitatif
2. Pendekatan Kuantitatif

2. Metode

1. Metode Penelitian Tindakan
2. Metode Eksperimen
3. Metode Studi Kasus
4. Metode Survei

3. Jenis

1. Dasar vs Terapan
2. Eksplanatori vs Konfirmatori
3. Deskripsi vs Eksperimen vs Korelasi

1. Pendekatan

1. Pendekatan Kualitatif:

- Dari ilmu sosial, konsepnya peningkatan pemahaman terhadap sesuatu, dan bukan membangun penjelasan dari sesuatu *(Berndtsson et al., 2008)*
- Sifatnya subyektif, berorientasi ke observasi tanpa dikontrol, dan secara umum tidak ada generalisasi *(Blaxter, Hughes, & Tight, 2006)*
- Dilakukan bidang sistem informasi, dengan metode penelitian seperti "studi kasus" dan "survei", berbasis pola alur induktif

Pengamatan terhadap obyek penelitian ***(Khusus)*** → Menghasilkan konsep atau teori baru ***(Umum)***

2. Pendekatan Kuantitatif:

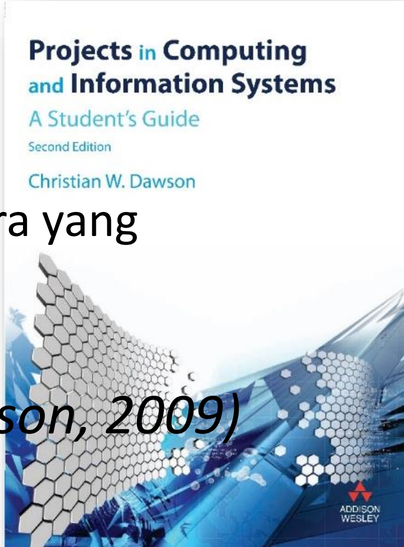
- Dari ilmu alam, konsepnya bagaimana sesuatu dibangun dan bekerja, dan membangun penjelasan dari sesuatu
- Sifatnya obyektif, berorientasi ke verifikasi, observasi yang dikontrol, dan secara umum ada generalisasi *(Blaxter et al., 2006)*
- Menggunakan skala numerik, berbasis pola alur deduktif-induktif

Kumpulkan Teori ***(Umum)*** → Hasilkan Konsep → Rumuskan Hipotesis ***(Khusus)*** → Uji Hipotesis → Tarik Kesmpulan ***(Umum)***

(Berndtsson et al., 2008)

2. Metode

1. Penelitian Tindakan
   - Studi berupa monitoring dan pencatatan penerapan sesuatu oleh peneliti secara hati-hati, yang tujuannya untuk memecahkan masalah dan mengubah situasi *(Herbert, 1990)*
   - Penelitian Tindakan Kelas (PTK) di bidang Pendidikan
2. Eksperimen
   - Investigasi hubungan sebab akibat dengan menggunakan ujicoba yang dikontrol oleh peneliti
   - Melibatkan pengembangan dan evaluasi
   - Penelitian bidang Science dan Teknik
3. Studi Kasus
   - Eksplorasi satu situasi secara mendalam dan hati hati *(Cornford and Smithson, 2006)*
   - Penelitian bidang Sosial, Ekonomi, Politik
4. Survei
   - Pengumpulan data dari populasi yang bisa diukur, dengan cara yang ekonomis *(Saunders et al., 2007)*
   - Melibatkan penggunaan kuesioner dan interview



(Dawson, 2009)

3. Jenis

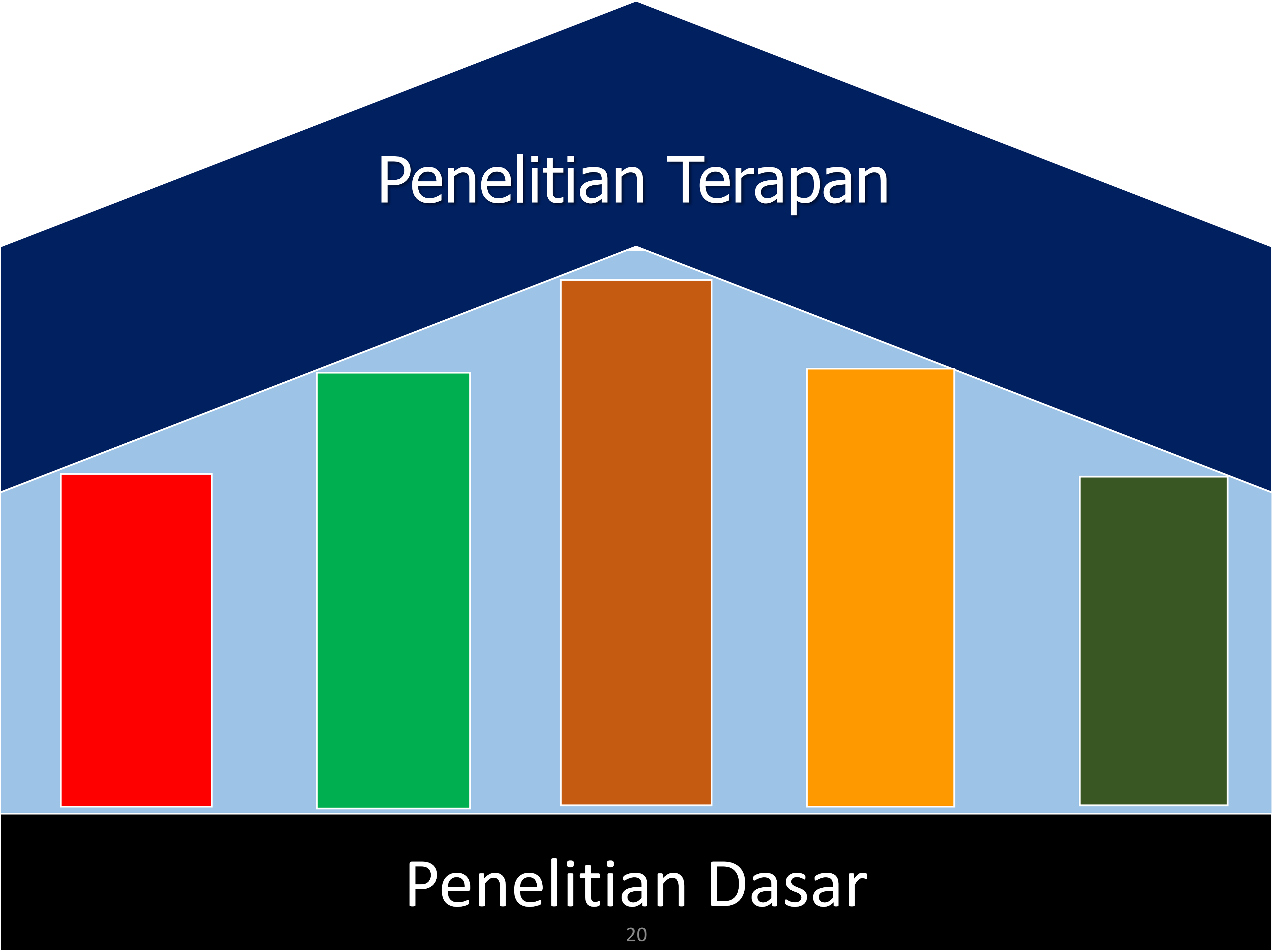
Deskripsi | Eksperimen | Korelasi

Kualitatif | Kuantitatif

Eksplanatori | Konfirmatori

Terapan

Dasar


Penelitian Terapan
Penelitian Dasar

Penerapan C4.5 untuk Prediksi Kelulusan Mahasiswa pada STMIK ABC

Split Criterion

C4.5

Gain Ratio

(Quinland, 1993)

Teori Gain *(Kullback & Leibler, 1951)*

Penerapan Credal C4.5 untuk Prediksi Kelulusan Mahasiswa pada STMIK ABC

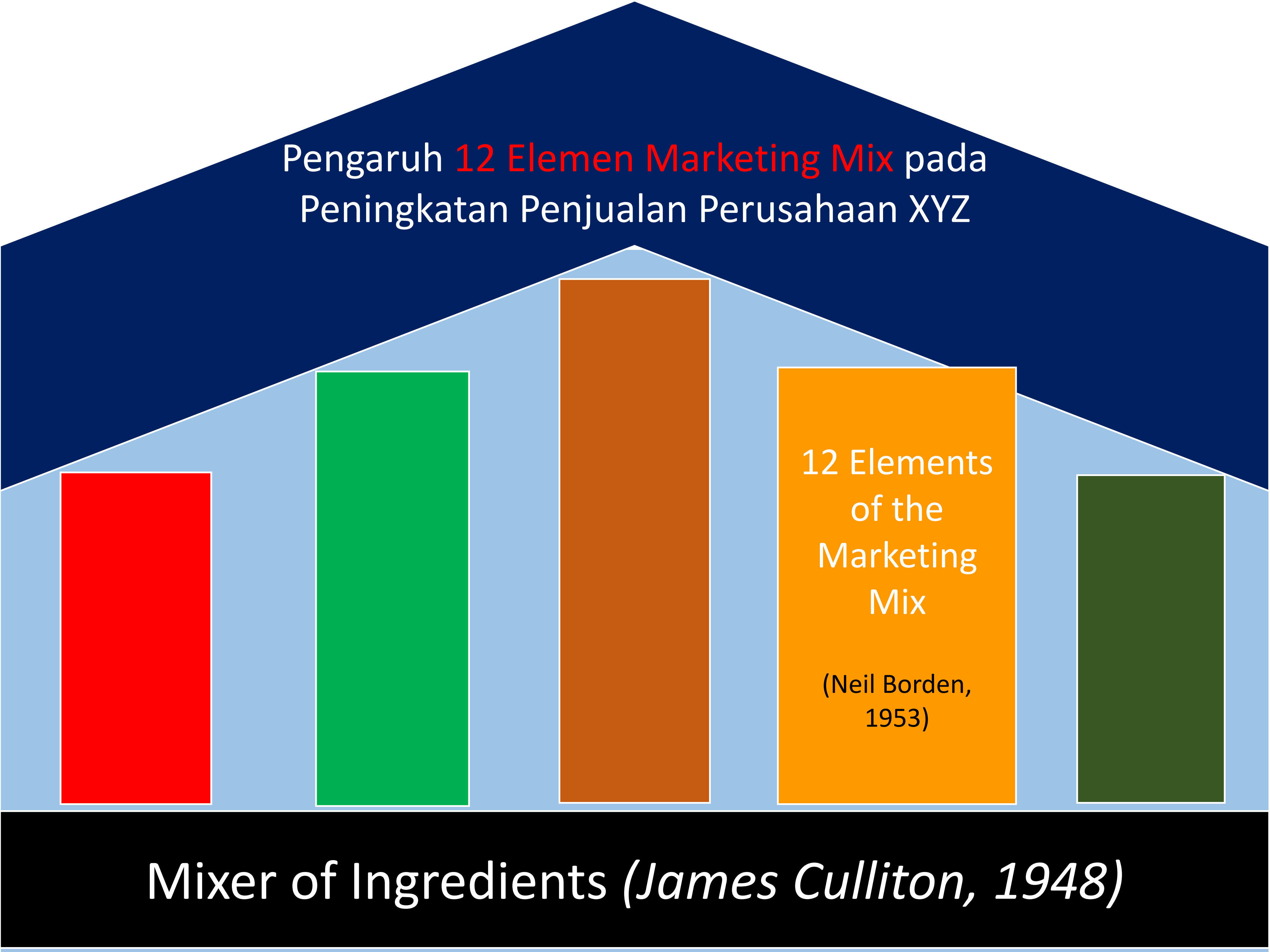
Split Criterion

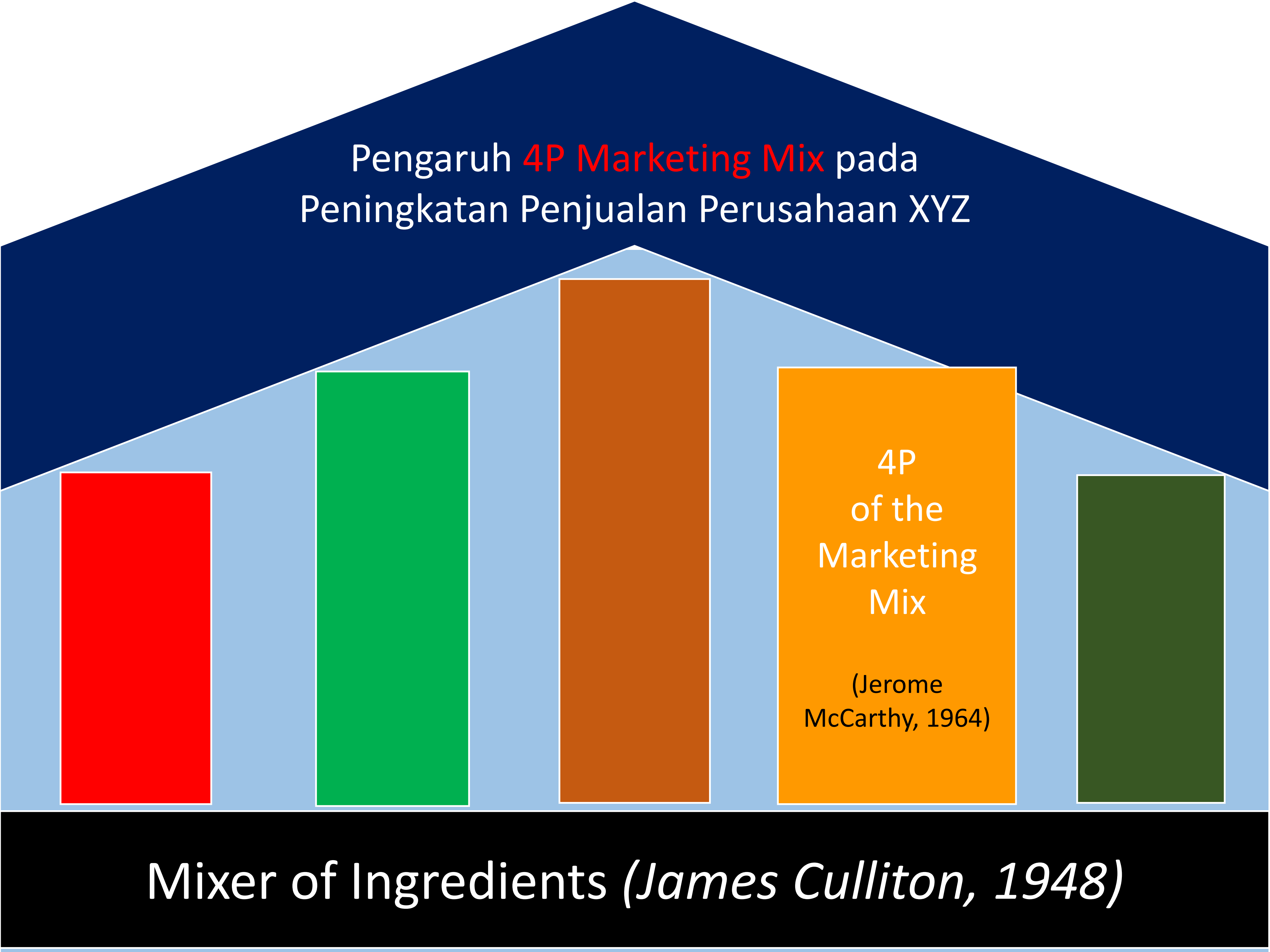
Credal C4.5

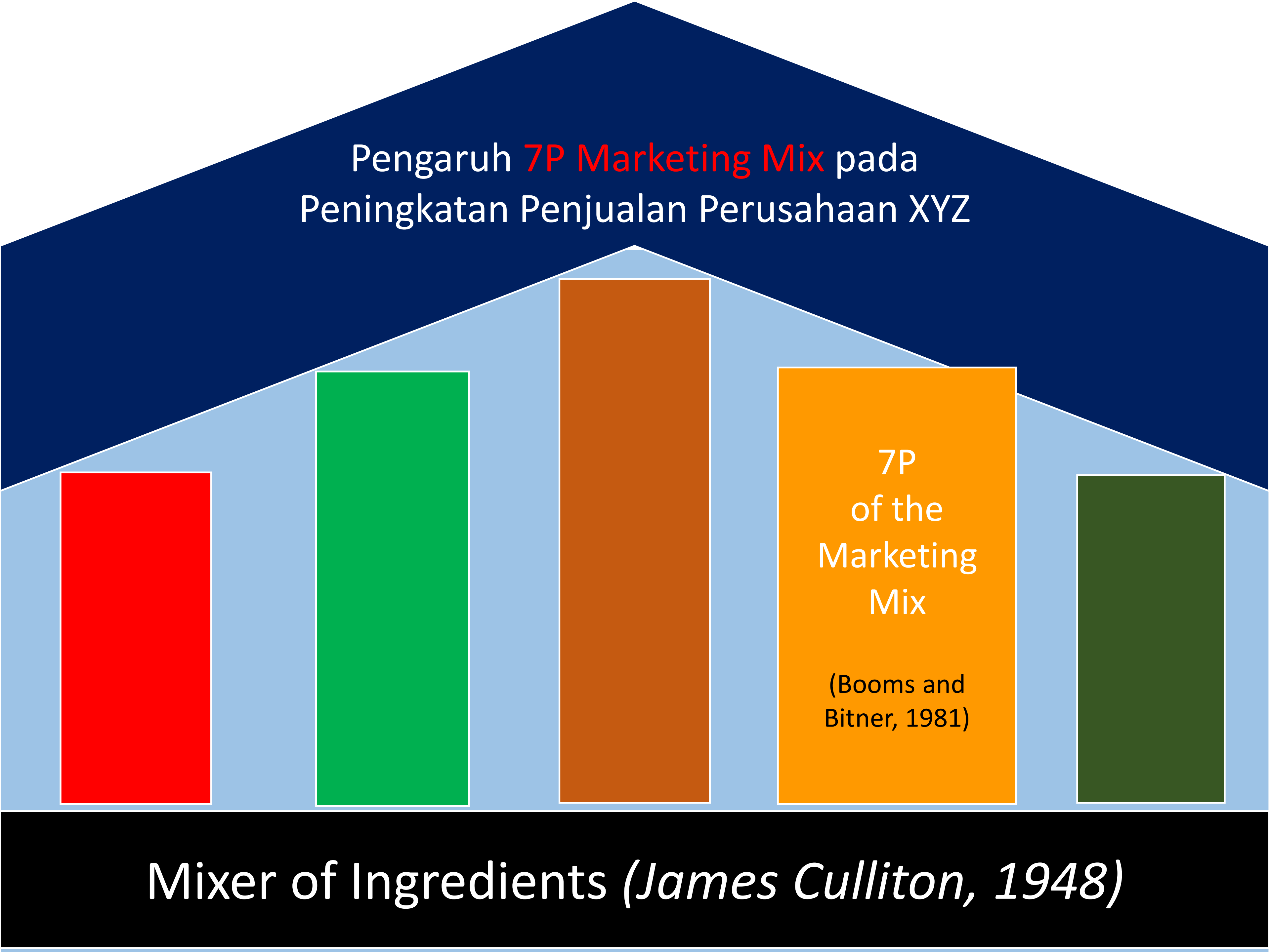
Imprecise Gain Ratio

(Mantas, 2013)

Imprecise Probability Theory *(Walley, 1996)*


Pengaruh 12 Elemen Marketing Mix pada Peningkatan Penjualan Perusahaan XYZ
12 Elements of the Marketing Mix
(Neil Borden, 1953)
Mixer of Ingredients *(James Culliton, 1948)*


Pengaruh 4P Marketing Mix pada
Peningkatan Penjualan Perusahaan XYZ
4P
of the
Marketing
Mix
(Jerome
McCarthy, 1964)
Mixer of Ingredients (James Culliton, 1948)


Pengaruh 7P Marketing Mix pada
Peningkatan Penjualan Perusahaan XYZ
7P
of the
Marketing
Mix
(Booms and
Bitner, 1981)
Mixer of Ingredients *(James Culliton, 1948)*

1.3 Gaya Penelitian Bidang Computing

IEEE/ACM Computing Curricula 2005

Computer Engineering (CE)

pengembangan sistem terintegrasi (software dan hardware)

Computer Engineer

Information System (IS)

analisa kebutuhan dan proses bisnis serta desain sistem

System Analyst

Computer Science (CS)

konsep computing dan pengembangan software

Computer Scientist

Information Technology (IT)

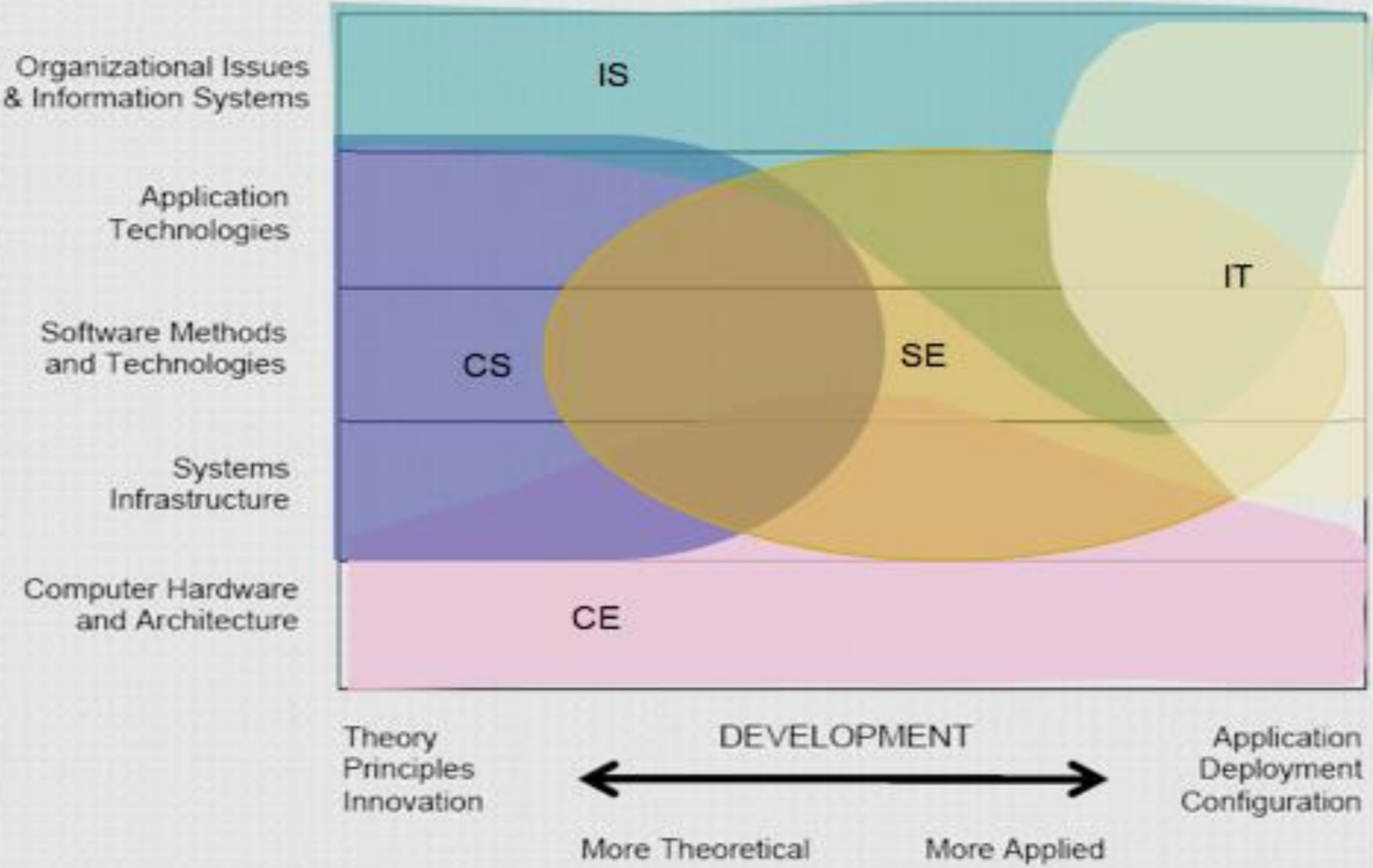
pengembangan dan pengelolaan infrastruktur IT

Infrastructure Engineer

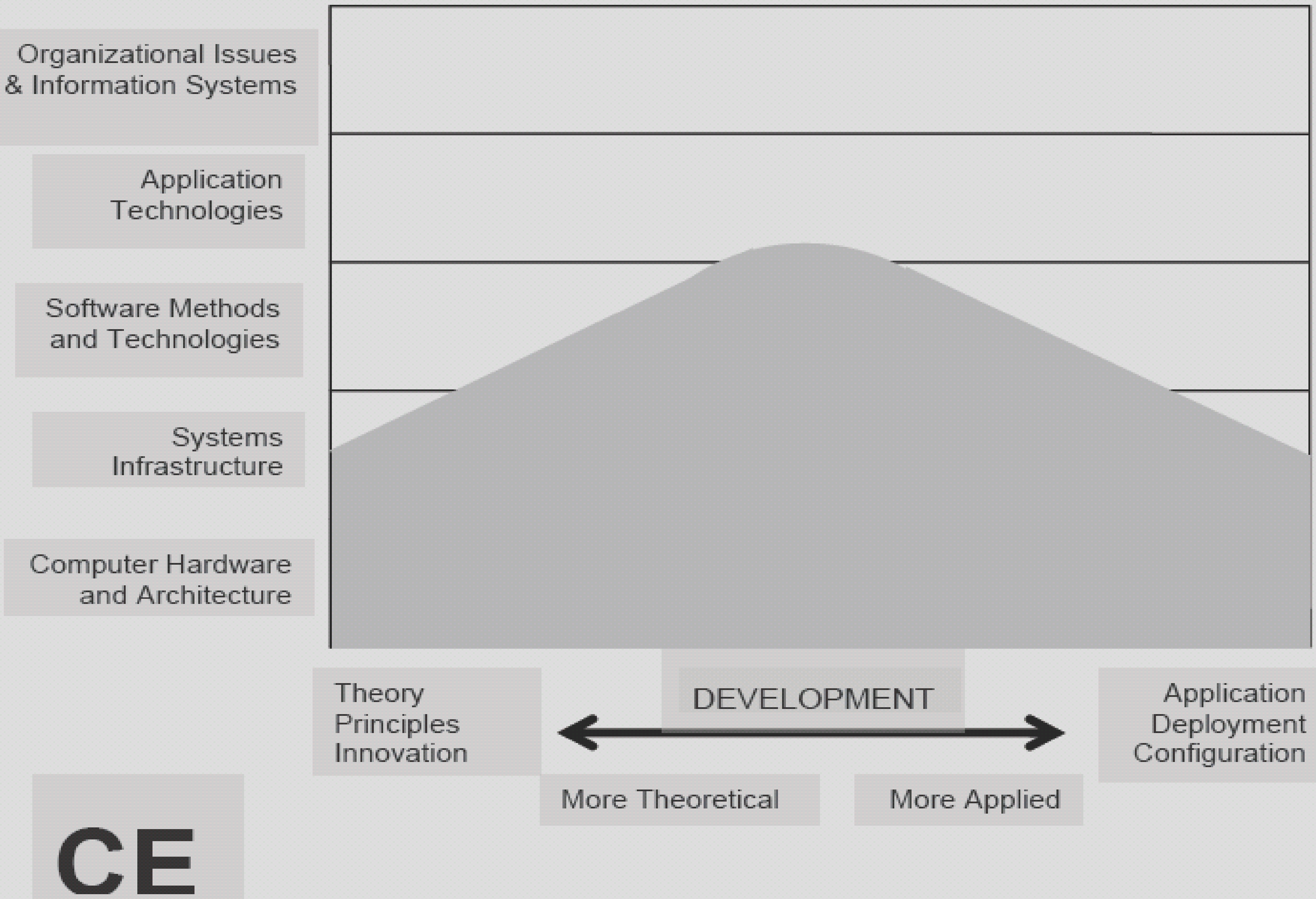
Software Engineering (SE)

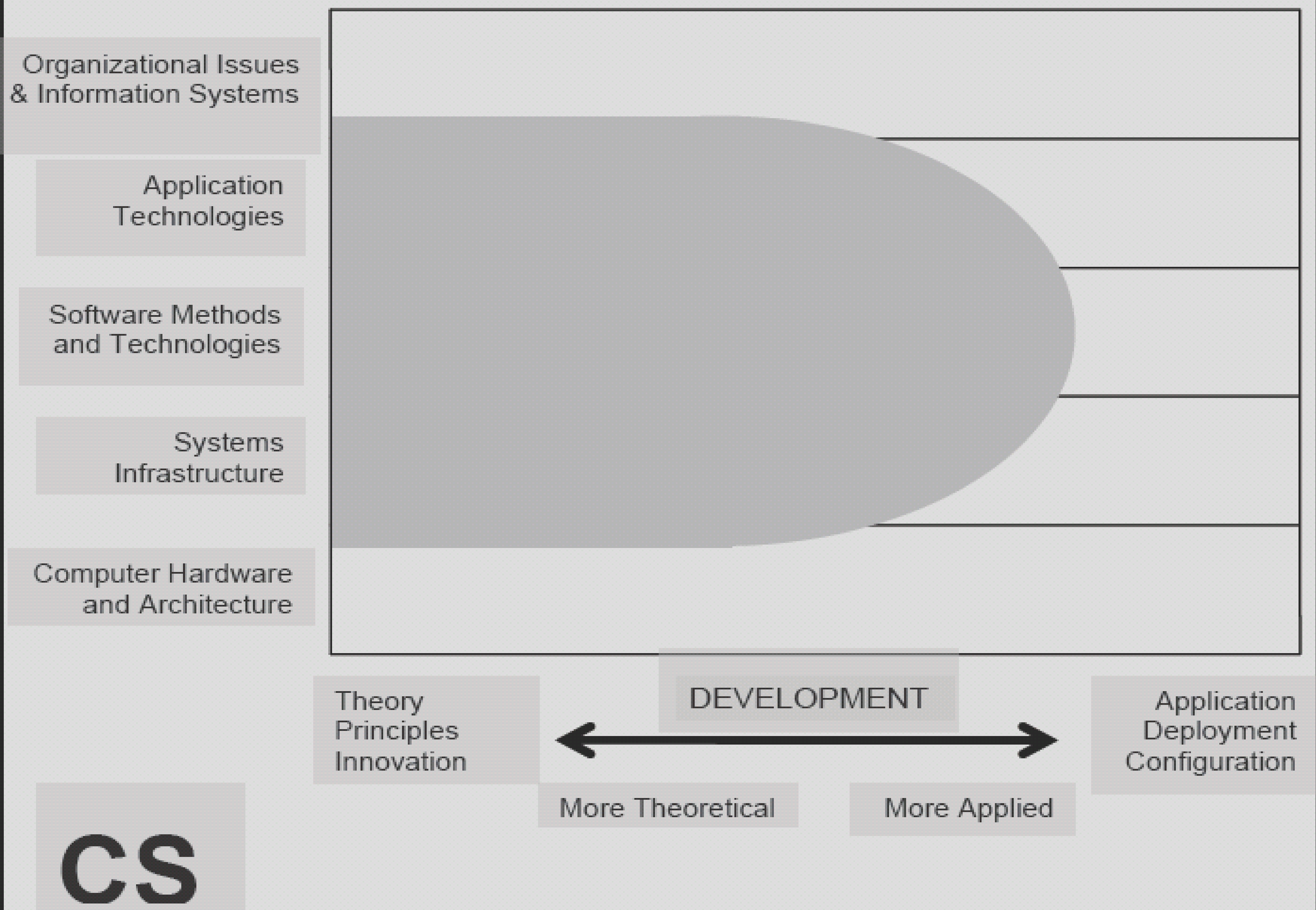
pengembangan software dan pengelolaan tahapan SDLC

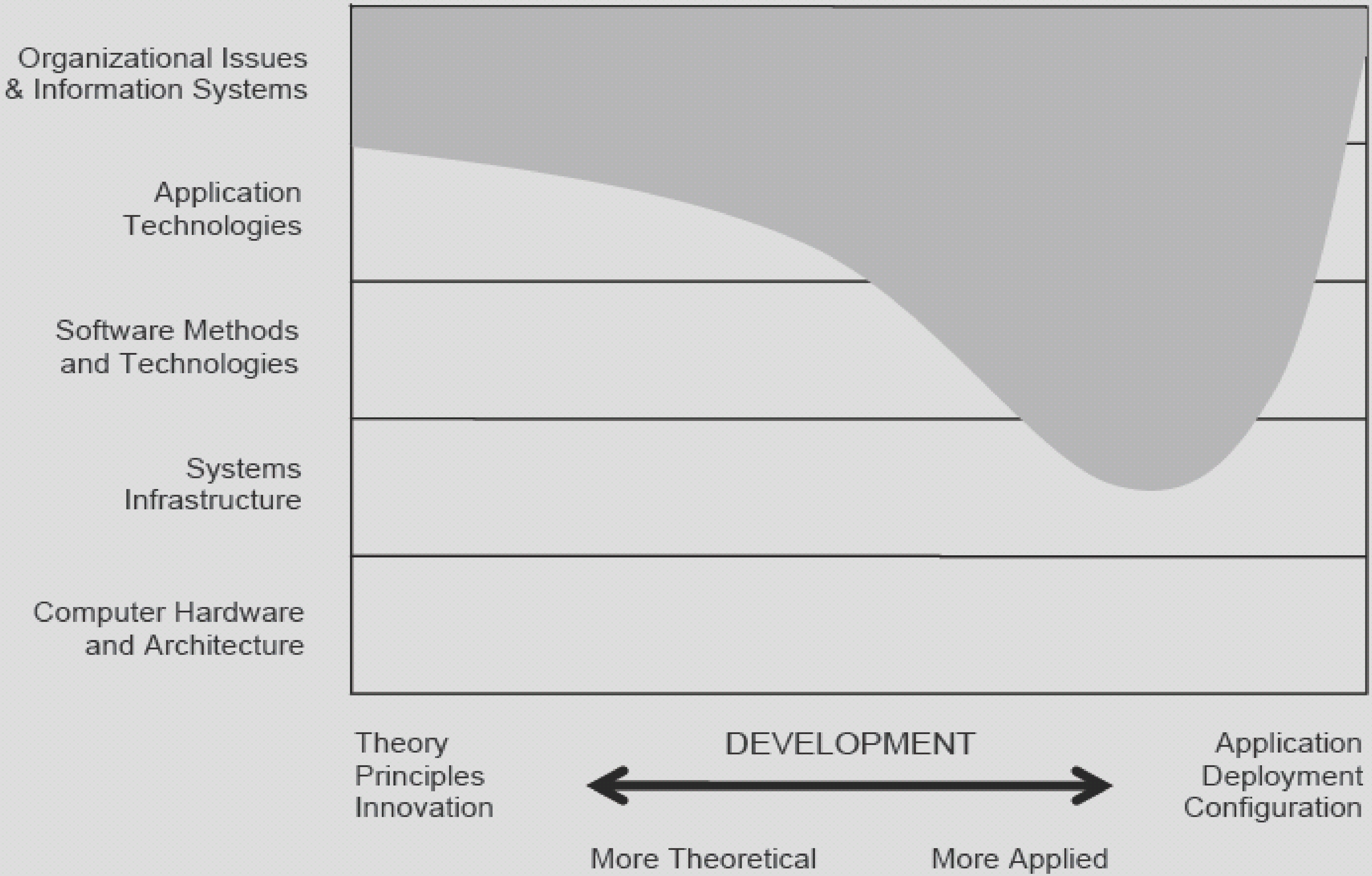
Software Engineer



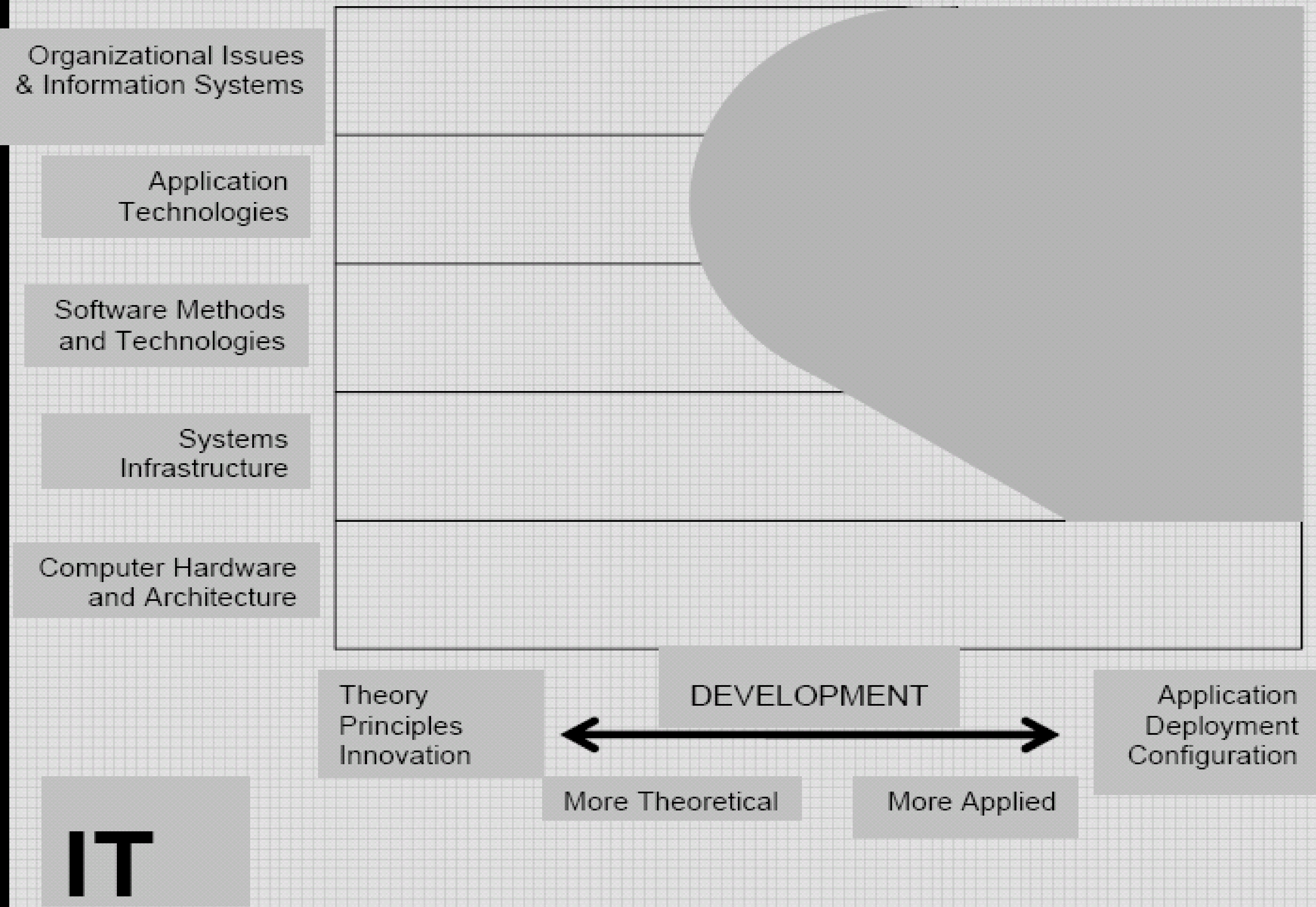
Computing Curricula 2005

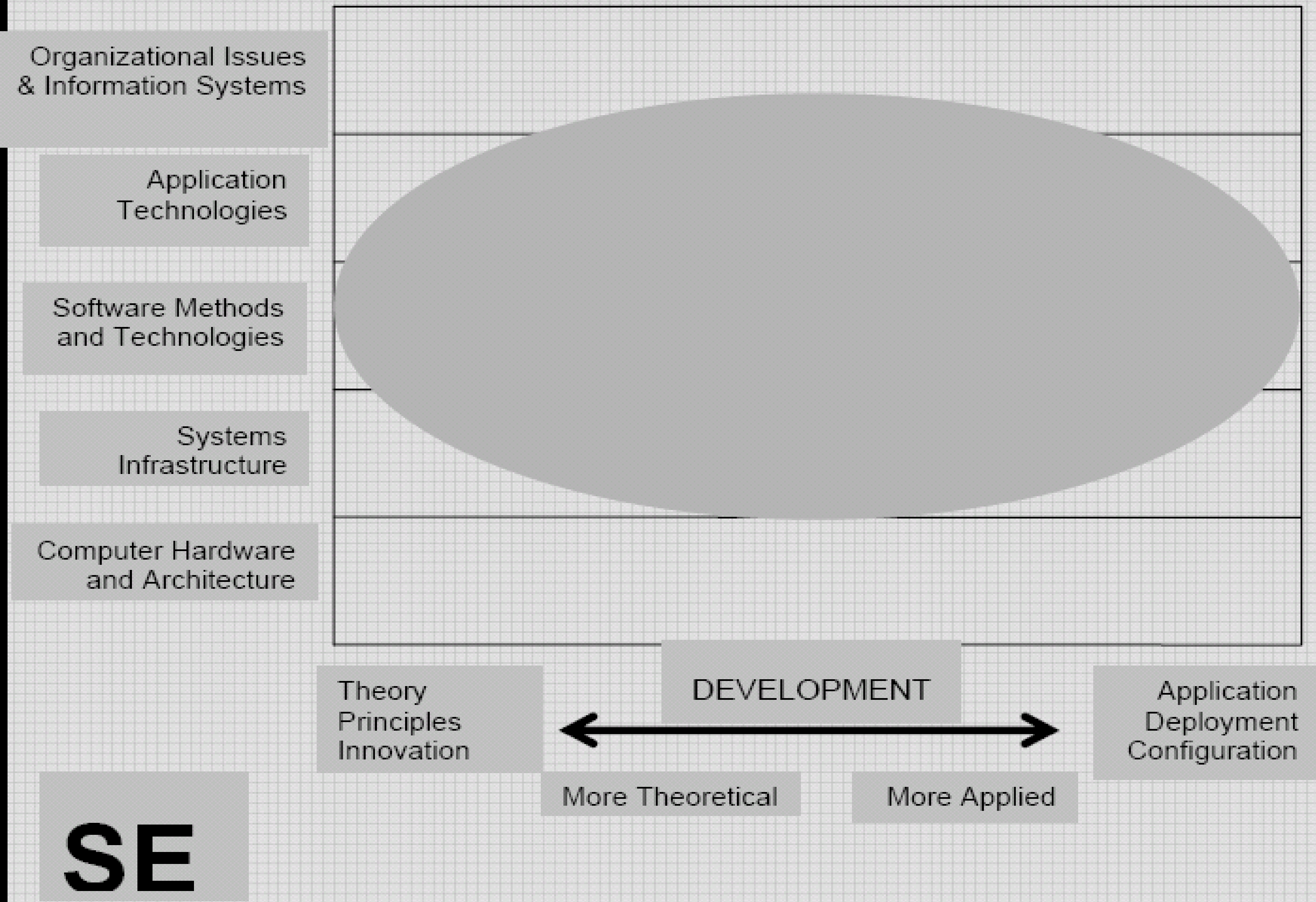

Organizational Issues & Information Systems
Application Technologies
Software Methods and Technologies
Systems Infrastructure
Computer Hardware and Architecture
Theory Principles Innovation
DEVELOPMENT
Application Deployment Configuration
More Theoretical
More Applied
CE


Organizational Issues & Information Systems
Application Technologies
Software Methods and Technologies
Systems Infrastructure
Computer Hardware and Architecture
DEVELOPMENT
Theory
Principles
Innovation
Application
Deployment
Configuration
More Theoretical
More Applied
CS


Organizational Issues
& Information Systems
Application
Technologies
Software Methods
and Technologies
Systems
Infrastructure
Computer Hardware
and Architecture
Theory
Principles
Innovation
DEVELOPMENT
Application
Deployment
Configuration
More Theoretical
More Applied

IS


Organizational Issues & Information Systems
Application Technologies
Software Methods and Technologies
Systems Infrastructure
Computer Hardware and Architecture
Theory Principles Innovation
DEVELOPMENT
Application Deployment Configuration
More Theoretical
More Applied
IT


Organizational Issues
& Information Systems
Application
Technologies
Software Methods
and Technologies
Systems
Infrastructure
Computer Hardware
and Architecture
Theory
Principles
Innovation
DEVELOPMENT
Application
Deployment
Configuration
More Theoretical
More Applied
SE

General Computing Courses -1-

Knowledge Area	CE		CS		IS		IT		SE	
	min	max	min	max	min	max	min	max	min	max
Programming Fundamentals	4	4	4	5	2	4	2	4	5	5
Integrative Programming	0	2	1	3	2	4	3	5	1	3
Algorithms and Complexity	2	4	4	5	1	2	1	2	3	4
Computer Architecture and Organization	5	5	2	4	1	2	1	2	2	4
Operating Systems Principles & Design	2	5	3	5	1	1	1	2	3	4
Operating Systems Configuration & Use	2	3	2	4	2	3	3	5	2	4
Net Centric Principles and Design	1	3	2	4	1	3	3	4	2	4
Net Centric Use and configuration	1	2	2	3	2	4	4	5	2	3
Platform technologies	0	1	0	2	1	3	2	4	0	3
Theory of Programming Languages	1	2	3	5	0	1	0	1	2	4
Human-Computer Interaction	2	5	2	4	2	5	4	5	3	5
Graphics and Visualization	1	3	1	5	1	1	0	1	1	3
Intelligent Systems (AI)	1	3	2	5	1	1	0	0	0	0
Information Management (DB) Theory	1	3	2	5	1	3	1	1	2	5
Information Management (DB) Practice	1	2	1	4	4	5	3	4	1	4
Scientific computing (Numerical mthds)	0	2	0	5	0	0	0	0	0	0
Legal / Professional / Ethics / Society	2	5	2	4	2	5	2	4	2	5
Information Systems Development	0	2	0	2	5	5	1	3	2	4
Analysis of Business Requirements	0	1	0	1	5	5	1	2	1	3
E-business	0	0	0	0	4	5	1	2	0	3

(Computing Curricula 2005: The Overview Report, ACM and IEEE CS, 2006)

General Computing Courses -2-

Knowledge Area	CE		CS		IS		IT		SE	
	min	max	min	max	min	max	min	max	min	max
Analysis of Technical Requirements	2	5	2	4	2	4	3	5	3	5
Engineering Foundations for SW	1	2	1	2	1	1	0	0	2	5
Engineering Economics for SW	1	3	0	1	1	2	0	1	2	3
Software Modeling and Analysis	1	3	2	3	3	3	1	3	4	5
Software Design	2	4	3	5	1	3	1	2	5	5
Software Verification and Validation	1	3	1	2	1	2	1	2	4	5
Software Evolution (maintenance)	1	3	1	1	1	2	1	2	2	4
Software Process	1	1	1	2	1	2	1	1	2	5
Software Quality	1	2	1	2	1	2	1	2	2	4
Comp Systems Engineering	5	5	1	2	0	0	0	0	2	3
Digital logic	5	5	2	3	1	1	1	1	0	3
Embedded Systems	2	5	0	3	0	0	0	1	0	4
Distributed Systems	3	5	1	3	2	4	1	3	2	4
Security: issues and principles	2	3	1	4	2	3	1	3	1	3
Security: implementation and mgt	1	2	1	3	1	3	3	5	1	3
Systems administration	1	2	1	1	1	3	3	5	1	2
Management of Info Systems Org.	0	0	0	0	3	5	0	0	0	0
Systems integration	1	4	1	2	1	4	4	5	1	4
Digital media development	0	2	0	1	1	2	3	5	0	1
Technical support	0	1	0	1	1	3	5	5	0	1

(Computing Curricula 2005: The Overview Report, ACM and IEEE CS, 2006)

General Non Computing Courses

Knowledge Area	CE		CS		IS		IT		SE	
	min	***max***	***min***	***max***	***min***	***max***	***min***	***max***	***min***	***max***
Organizational Theory	0	0	0	0	1	4	1	2	0	0
Decision Theory	0	0	0	0	3	3	0	1	0	0
Organizational Behavior	0	0	0	0	3	5	1	2	0	0
Organizational Change Management	0	0	0	0	2	2	1	2	0	0
General Systems Theory	0	0	0	0	2	2	1	2	0	0
Risk Management (Project, safety risk)	2	4	1	1	2	3	1	4	2	4
Project Management	2	4	1	2	3	5	2	3	4	5
Business Models	0	0	0	0	4	5	0	0	0	0
Functional Business Areas	0	0	0	0	4	5	0	0	0	0
Evaluation of Business Performance	0	0	0	0	4	5	0	0	0	0
Circuits and Systems	5	5	0	2	0	0	0	1	0	0
Electronics	5	5	0	0	0	0	0	1	0	0
Digital Signal Processing	3	5	0	2	0	0	0	0	0	2
VLSI design	2	5	0	1	0	0	0	0	0	1
HW testing and fault tolerance	3	5	0	0	0	0	0	2	0	0
Mathematical foundations	4	5	4	5	2	4	2	4	3	5
Interpersonal communication	3	4	1	4	3	5	3	4	3	4

(Computing Curricula 2005: The Overview Report, ACM and IEEE CS, 2006)

Information Systems vs Computer Science

Information Systems (IS): IS, IT

Computer Science (CS): CS, CE, SE

Information systems specialists focus on integrating information technology solutions and business processes to meet the information needs of businesses and other enterprises *(ACM CC 2005)*

Computer science spans a wide range, from its theoretical and algorithmic foundations to cutting-edge developments in robotics, computer vision, intelligent systems, bioinformatics, and other exciting areas *(ACM CC 2005)*

The Scope of Information Systems

Information Systems as a field of academic study encompasses the concepts, principles, and processes for two broad areas of activity within organizations:

1. Acquisition, deployment, management, and strategy for information technology resources and services
   → *the information systems function; IS strategy, management, and acquisition; IT infrastructure; enterprise architecture; data and information*
2. Packaged system acquisition or system development, operation, and evolution of infrastructure and systems for use in organizational processes
   → *project management, system acquisition, system development, system operation, and system maintenance)*

(Curriculum Guidelines for Undergraduate Degree Programs in Information Systems, ACM and AIS, 2010)

Information Systems Profession

- IS specialists focus on integrating information technology solutions and business processes to meet the information needs of businesses and other enterprises, enabling them to achieve their objectives in an effective, efficient way
- IS specialists concerned with the information that computer systems can provide to aid an enterprise in defining and achieving its goals, and the processes that an enterprise can implement or improve using information technology
- They must understand both technical and organizational factors, and they must be able to help an organization determine how information and technology-enabled business processes can provide a competitive advantage

(Computing Curricula 2005: The Overview Report, ACM and IEEE CS, 2006)

Computer Science Profession

1. They design and implement software. Computer scientists take on challenging programming jobs. They supervise other programmers, keeping them aware of new approaches (**Software Development)**
2. They devise new ways to use computers. Progress in the CS areas of networking, database, and human-computer-interface enabled the development of the World Wide Web. Now CS researchers are working with scientists from other fields to make robots become practical and intelligent aides, to use databases to create new knowledge, and to use computers to help decipher the secrets of our DNA **(Computing Algorithm Applications)**
3. They develop effective ways to solve computing problems. Computer scientists develop the best possible ways to store information in databases, send data over networks, and display complex images. Their theoretical background allows them to determine the best performance possible, and their study of algorithms helps them to develop new approaches that provide better performance **(Computing Algorithm Inventions)**

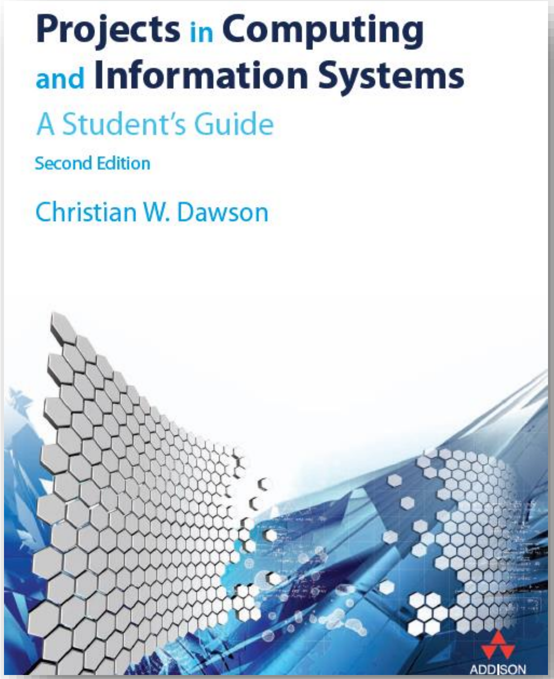
(Computing Curricula 2005: The Overview Report, ACM and IEEE CS, 2006)

1.4 Kontribusi dan Orisinalitas

Apa Yang Dikejar di Penelitian?

Research is a considered activity, which aims to make an original contribution to knowledge

(Dawson, 2009)

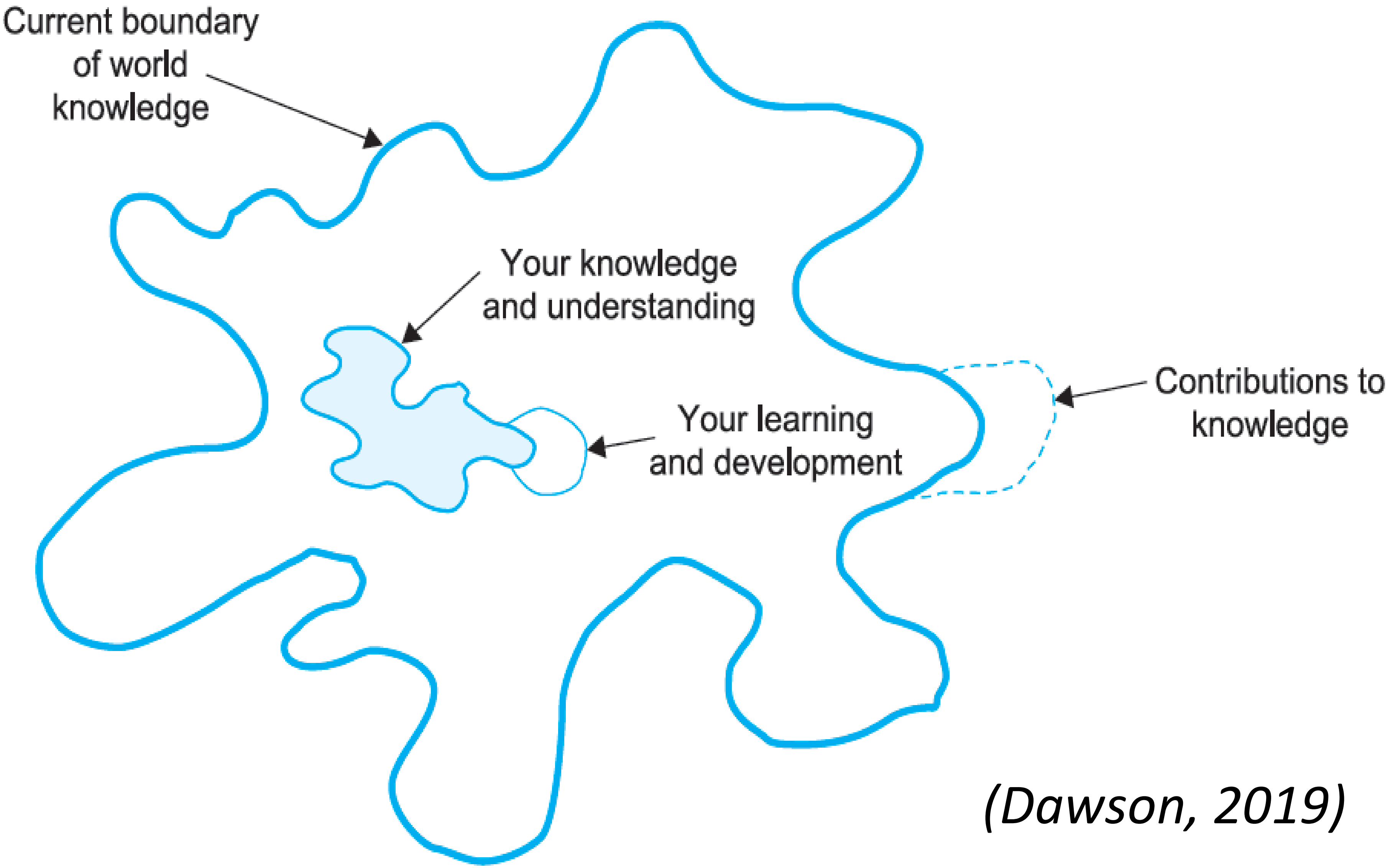


Bentuk Kontribusi ke Pengetahuan

Kegiatan penyelidikan dan investigasi terhadap suatu masalah yang dilakukan secara berulang-ulang dan sistematis, dengan tujuan untuk menemukan atau merevisi teori, metode, fakta, dan aplikasi

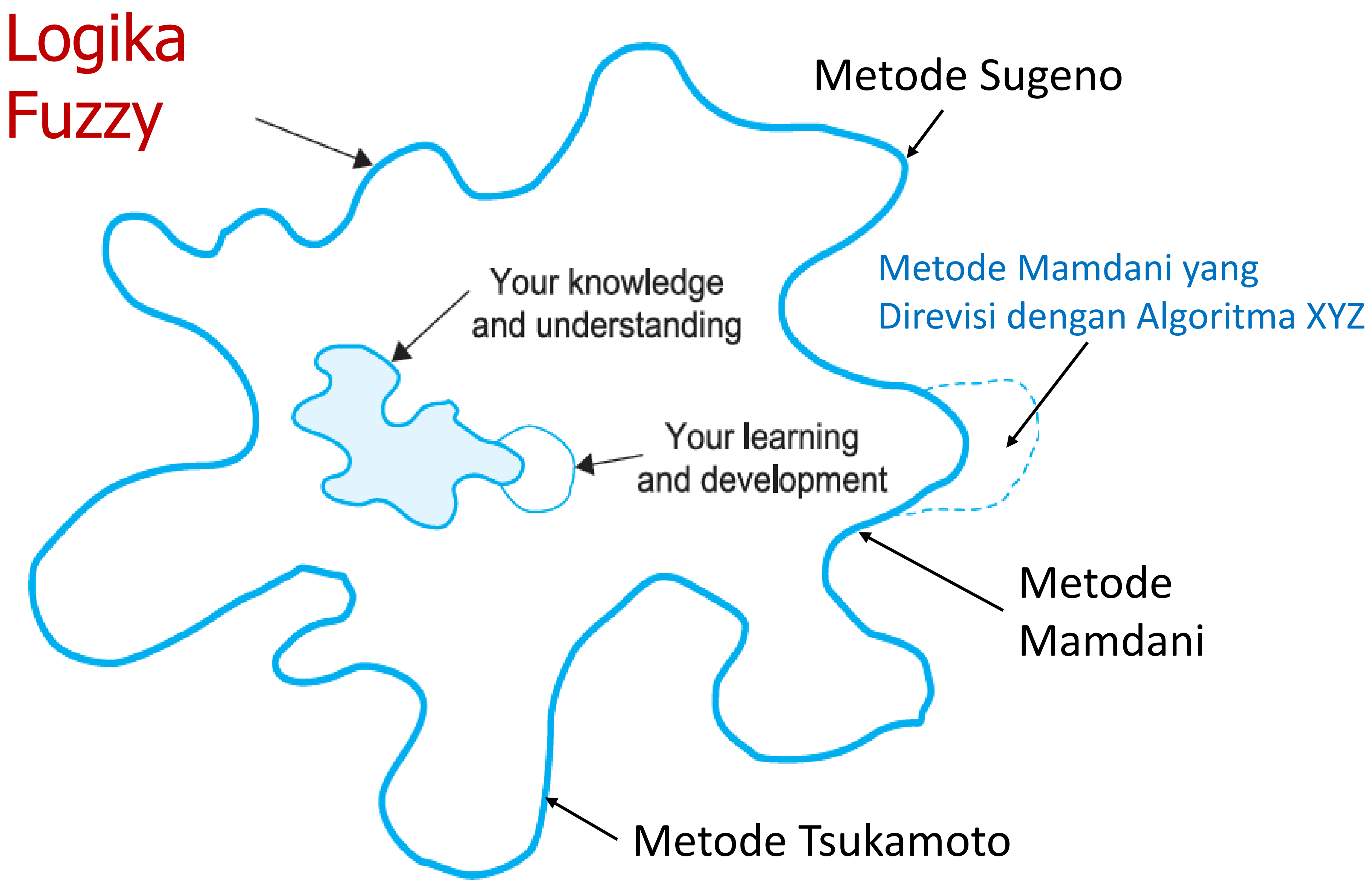
(Berndtsson et al., 2008)

Bentuk Kontribusi ke Pengetahuan

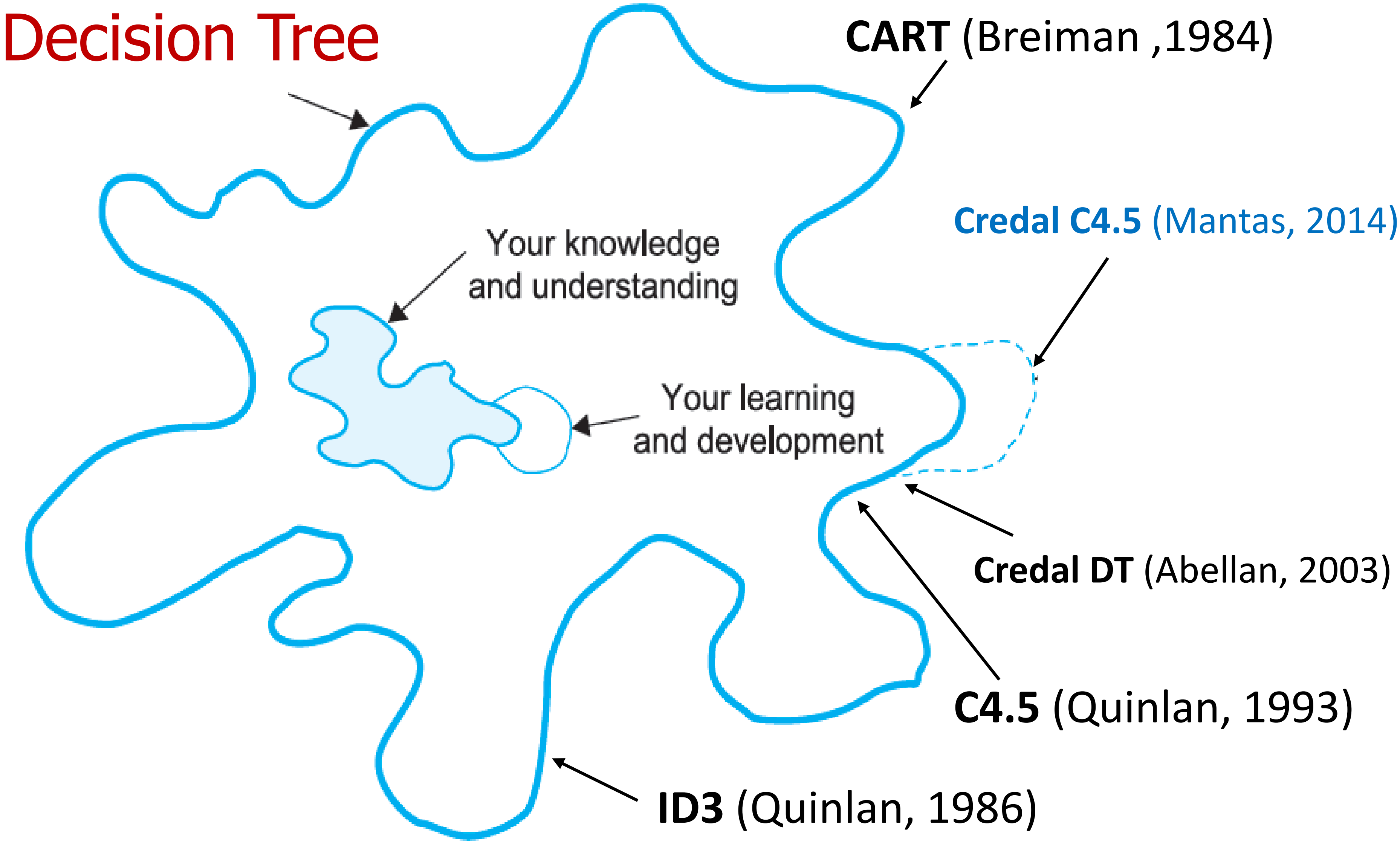


(Dawson, 2019)

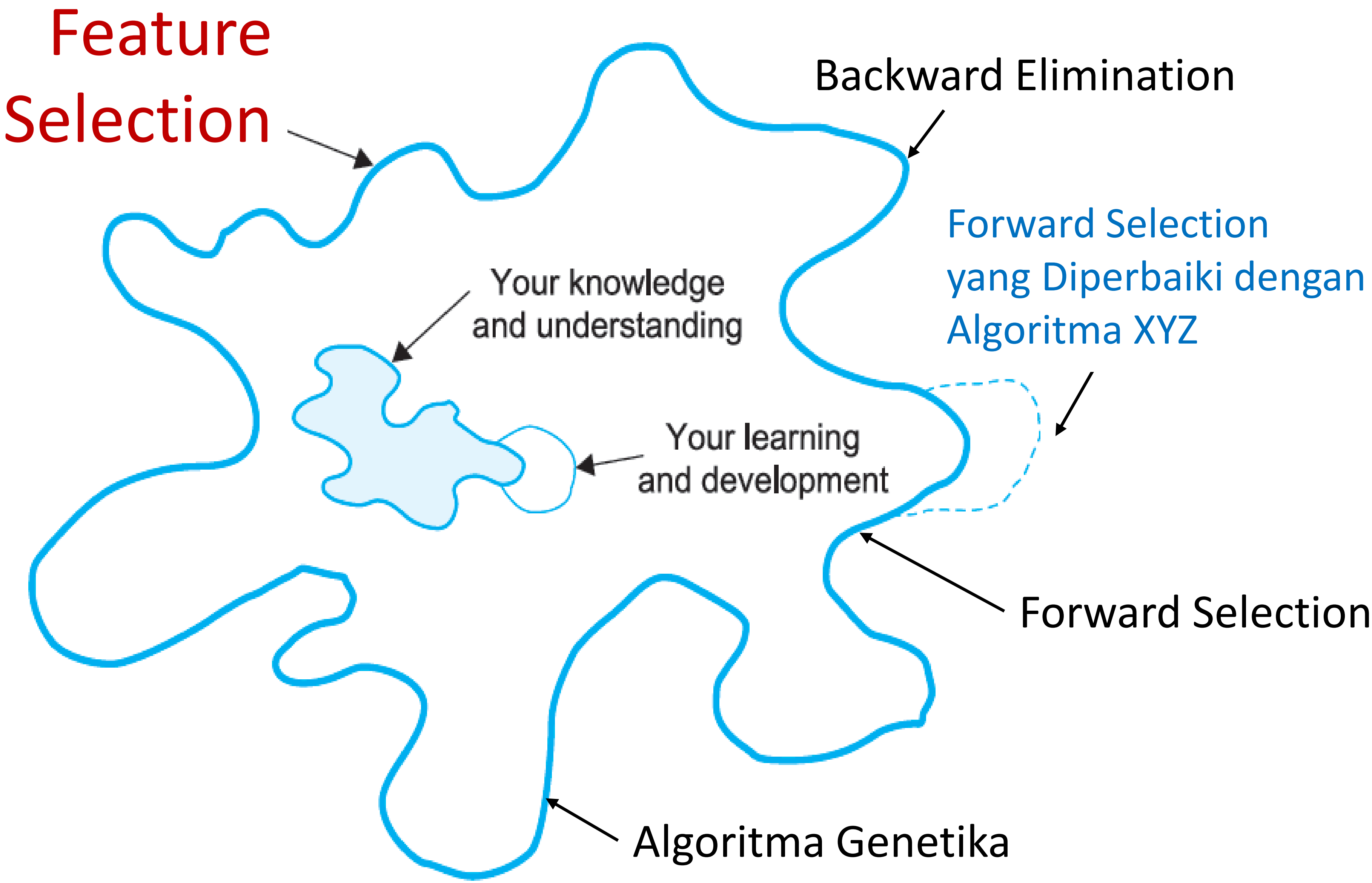
Bentuk Kontribusi ke Pengetahuan



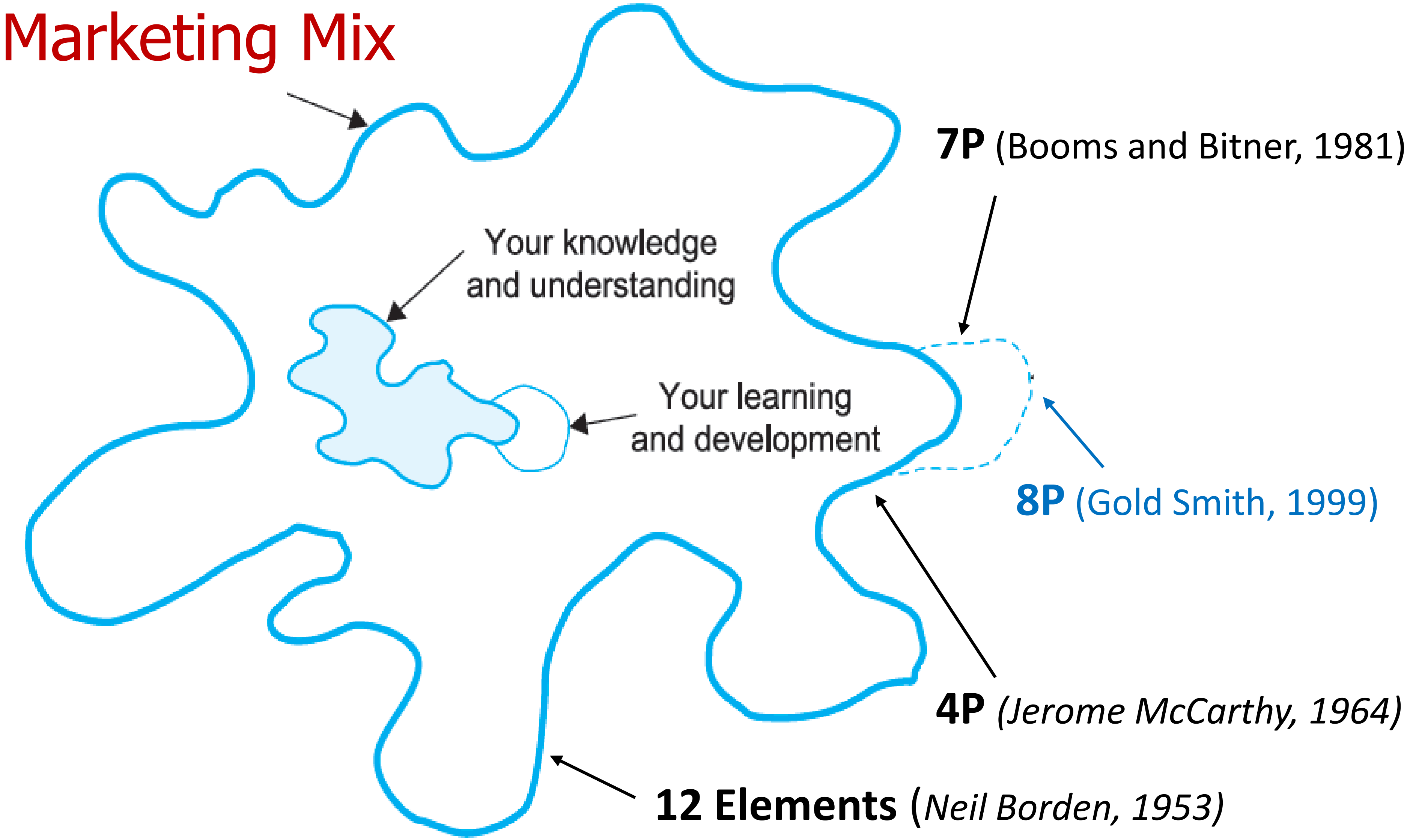
Bentuk Kontribusi ke Pengetahuan



Contoh Kontribusi ke Pengetahuan



Bentuk Kontribusi ke Pengetahuan



Orisinalitas Penelitian

1. Orisinalitas pada Metode:
   - Memecahkan masalah yang orang lain sudah pernah mengerjakan sebelumnya, tapi dengan metode yang berbeda
   - Model penelitian yang kontribusi ada pada method improvement

2. Orisinalitas pada Masalah:
   - Memecahkan suatu masalah yang orang lain belum pernah mengerjakan sebelumnya
   - Model penelitian yang kontribusi ada pada penemuan masalah baru sebagai obyek penerapan metode

(Dawson, 2009)

Contoh Tema Penelitian

Algoritma Genetika untuk Penentuan Desain Bendungan yang Paling Optimal

Contoh Kontribusi pada Metode

- Judul:

  Penerapan Metode XYZ untuk Pemecahan Masalah Konvergensi Prematur pada Algoritma Genetika untuk Penentuan Desain Bendungan

- Kontribusi: Menerapkan Metode XYZ yang sebelumnya tidak pernah digunakan orang untuk memecahkan masalah konvergensi premature pada Algoritma Genetika

Contoh Kontribusi pada Masalah

- Judul:

  Penerapan Algoritma Genetika untuk Penentuan Desain Bendungan dengan Tujuh Parameter

- Kontribusi: Penentuan Desain Bendungan dengan Tujuh Parameter (kebanyakan peneliti menggunakan tiga parameter)

Contoh Kontribusi pada Masalah dan Metode

- Judul:

  Penerapan Metode XYZ untuk Pemecahan Masalah Konvergensi Prematur pada Algoritma Genetika untuk Penentuan Desain Bendungan dengan Tujuh Parameter

- Kontribusi:

  1. Penerapan metode XYZ untuk memecahkan masalah konvergensi premature pada algoritma genetika
  2. Penentuan Desain Bendungan dengan Tujuh Parameter

Contoh Penelitian Tanpa Kontribusi

- Penerapan Algoritma Genetika untuk Penentuan Desain Bendungan di Bendungan Jatiluhur
- Penerapan Algoritma Genetika untuk Penentuan Desain Bendungan di Bendungan Gajah Mungkur
- Penerapan Algoritma Genetika untuk Penentuan Desain Bendungan di Bendungan Karang Kates

* banyak peneliti computing di Indonesia yang terjebak dengan penelitian tanpa kontribusi dan hanya mengganti obyek tempat, akhirnya ditolak ketika publikasi ke journal internasional terindeks

Komparasi Penelitian D3/D4 vs S1 vs S2 vs S3

Aspek	Tugas Akhir (D3/D4)	Skripsi (D4/S1)	Tesis (S2)	Disertasi (S3)
Level Kontribusi	Penguasaan Kemampuan Teknis	Pengujian Teori	Pengembangan Teori	Penemuan Teori Baru
Bentuk Kontribusi	Implementasi dan pengembangan	Implementasi dan pengembangan	Perbaikan Secara Inkremental dan Terus Menerus	Substansial dan Invention
Target Publikasi	-	Domestic Conference	International Conference	International Journal

(Permendikbud No 3 tahun 2020 tentang SNPT)

Komparasi Kontribusi Penelitian S1 vs S2 vs S3

- D3/D4:
  - Pengembangan Sistem Informasi Rumah Sakit untuk Rumah Sakit "Suka Sembuh"
  - *Karakter: menguasai skill teknis*
- S1:
  - Sistem Cerdas Berbasis Neural Network untuk Prediksi Harga Saham
  - *Karakter: menguji teori, ada software development*
- S2:
  - Penerapan Algoritma Genetika untuk Pemilihan Arsitektur Jaringan Secara Otomatis pada Neural Network untuk Prediksi Harga Saham
  - *Karakter: mengembangkan teori (perbaikan metode), ada kontribusi ke teori/metode meskipun specific obyek*
- S3:
  - Penerapan Algoritma XYZ untuk Pemilihan Arsitektur Jaringan Secara Otomatis pada Neural Network
  - *Karakter: menemukan teori (invensi metode), ada kontribusi ke teori/metode dengan generalisasi lebih luas*

Contents lists available at ScienceDirect
Expert Systems with Applications
journal homepage: www.elsevier.com/locate/eswa

Credal-C4.5: Decision tree based on imprecise probabilities to classify noisy data

Carlos J. Mantas, Joaquín Abellán *
Department of Computer Science & Artificial Intelligence, University of Granada, ETSI Informática, c/Periodista Daniel Saucedo Aranda s/n, 18071 Granada, Spain

Memperbaiki C4.5

Contents lists available at ScienceDirect
Information and Software Technology
journal homepage: www.elsevier.com/locate/infsof

Simplifying effort estimation based on Use Case Points ☆

M. Ochodek *, J. Nawrocki, K. Kwarciak
Poznan University of Technology, Institute of Computing Science, ul. Piotrowo 2, 60-965 Poznań, Poland

Memperbaiki Use Case Points

Genetic Algorithms With Guided and Local Search Strategies for University Course Timetabling

Shengxiang Yang, *Member, IEEE*, and Sadaf Naseem Jat

Abstract—**The university course timetabling problem (UCTP) is a combinatorial optimization problem, in which a set of events has to be scheduled into time slots and located into suitable rooms. The design of course timetables for academic institutions is a very difficult task because it is an NP-hard problem. This paper investigates genetic algorithms (GAs) with a guided search strategy and local search (LS) techniques for the UCTP. The guided search strategy is used to create offspring into the population based on a data structure that stores information extracted from good individu-**

The research on timetabling problems has a long history of more than 40 years, starting with Gotlieb in 1962 [22]. Researchers have proposed various timetabling approaches by using graph coloring methods, constraint-based methods, population-based approaches (e.g., genetic algorithms (GAs), ant-colony optimization, and memetic algorithms), metaheuristic methods (e.g., tabu search (TS), simulated annealing (SA), and great deluge), variable neighborhood search (VNS), hy-

Memperbaiki Genetic Algorithms

Komparasi Kontribusi Penelitian S1 vs S2 vs S3

- **S1**:
  - Pengaruh 4P Marketing Mix pada Peningkatan Penjualan Perusahaan XYZ
  - *Kontribusi: menguji dan menerapkan teori/hukum/model/metode*
- **S2**:
  - Pengaruh 4P+3C Marketing Mix pada Peningkatan Penjualan Perusahaan XYZ
  - *Kontribusi: mengembangkan dan memperbaiki teori/hukum/model/metode*
- **S3**:
  - Pengaruh ABCD Marketing Mix pada Peningkatan Penjualan Perusahaan
  - *Kontribusi: mengembangan dan menemukan (invention) teori/hukum/model/metode baru yang sifatnya lebih general*

Perbaikan 4P Menjadi 4E

Marketing Mix Baru Khusus untuk Industri Turisme

Marketing Mix Khusus untuk Private Labels Brand Equity

Standarisasi Marketing Mix untuk Generalisasi Lebih Luas

Contents lists available at ScienceDirect

Journal of Business Research

ELSEVIER

HOSTED BY ELSEVIER

CrossMark

Available online at www.sciencedirect.com

ScienceDirect

Future Business Journal 3 (2017) 47–69

European Journal of Management and Business Economics

European Academy of Management and Business Economics

www.elsevier.es/ejmbe

Article

Marketing mix effects on

Carmen Abril[a,*], Belén Rodriguez

[a] Facultad de CC Económicas y Empresariales, Universida

[b] Universidad Pontificia de Comillas, C/Alberto Aguilera 2

ARTICLE INFO

Article history:
Received 8 June 2016
Accepted 15 September 2016
Available online 25 October 2016

JEL classification:
M31
M37

Keywords:
Private labels
Store brands
Brand equity
In-store communication
Distribution

Introduction

Private label brands, also known as "store brands", were considered low-price, low-q decades ago; currently, however, they repre to manufacturer brands (Kapferer, 2008). than 40% of the market in six European c Manufacturers Association [PLMA], 2015). I refer to brands owned by the retailer or dist

Available online at www.sciencedirect.com

SCIENCE DIRECT®

ELSEVIER

International Business Review 13 (2004) 637–659

INTERNATIONAL BUSINESS REVIEW

www.elsevier.com/locate/ibusrev

Marketing mix standardization: a cross cultural study of four countries

Richard Alan Kustin

Southern Connecticut State University, New Haven, CT 06515, USA

Abstract

The study researched the possibility of standardizing the marketing mix by investigating the cross-cultural responses from the United States, Brazil, France and India. The study tested the premise of standardization by determining if respondents perceived specific attributes of a common non-durable consumer product the same or differently. The results indicate the opportunity for dynamic marketing standardization remains limited but applicable within specific cultural country markets. Several attribute perceptions between US and foreign respondents are found to be more similar than dissimilar suggesting advantages may exist for a limited implementation of marketing mix standardization as part of a global marketing strategy.

Penelitian Yang Memiliki Kontribusi?

- Penerapan algoritma genetika untuk penjadwalan mata kuliah
  - Penerapan algoritma genetika berbasis *guided local search strategies* untuk penjadwalan mata kuliah (Yang, 2011)
- Penerapan algoritma C4.5 untuk penentuan kelulusan mahasiswa tepat waktu: Studi Kasus STMIK XYZ
  - Penerapan algoritma C4.5 dengan penghitungan entropi berbasis metode ABC untuk penentuan kelulusan mahasiswa tepat waktu

Hanya penelitian dengan kontribusi ke pengetahuan yang bisa menembus jurnal-jurnal internasional terindeks

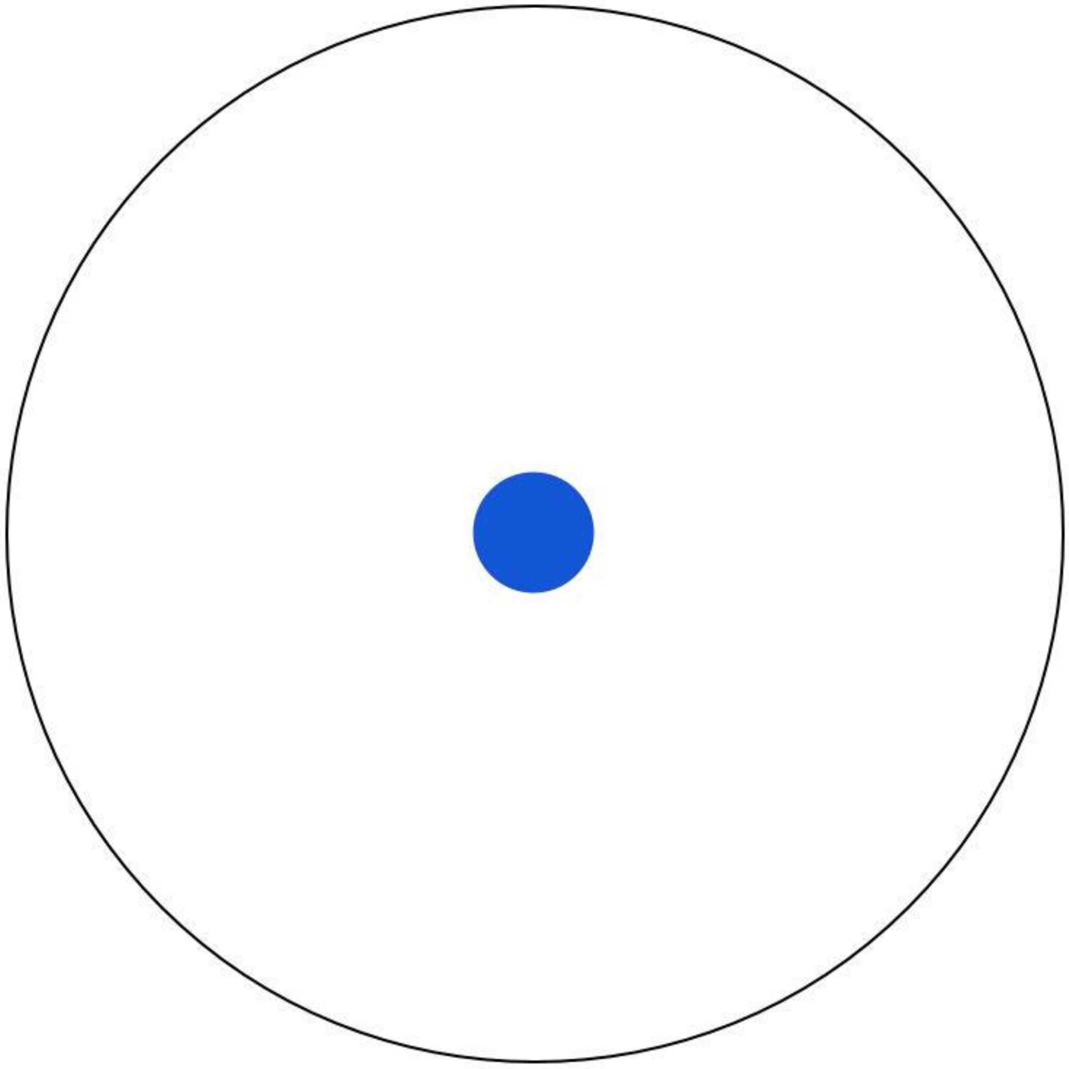
Penelitian Yang Memiliki Kontribusi?

No	Judul	
1	Penerapan Neural Network untuk Prediksi Harga Saham pada Perusahaan ABC	✗
2	Pemilihan Arsitektur Jaringan pada Neural Network Secara Otomatis dengan Menggunakan Algoritma Semut	✓
3	Modifikasi Penghitungan Gain dan Entropi untuk Peningkatan Akurasi pada Algoritma C4.5	✓
4	Penerapan Framework TOGAF untuk Pengembangan Enterprise Architecture pada Organisasi ABC	✗
5	Penerapan Framework TOGAF yang Dimodifikasi untuk Pengembangan Enterprise Architecture pada Perusahaan Skala Kecil dan Menengah	✓
6	Penerapan COBIT untuk Tata Kelola Organisasi ABC	✗
7	Integrasi COBIT dan TOGAF untuk Tata Kelola Organisasi ABC yang Lebih Komprehensif	✓
8	Penerapan algoritma genetika untuk penjadwalan mata kuliah: Studi Kasus STMIK ABC	✗

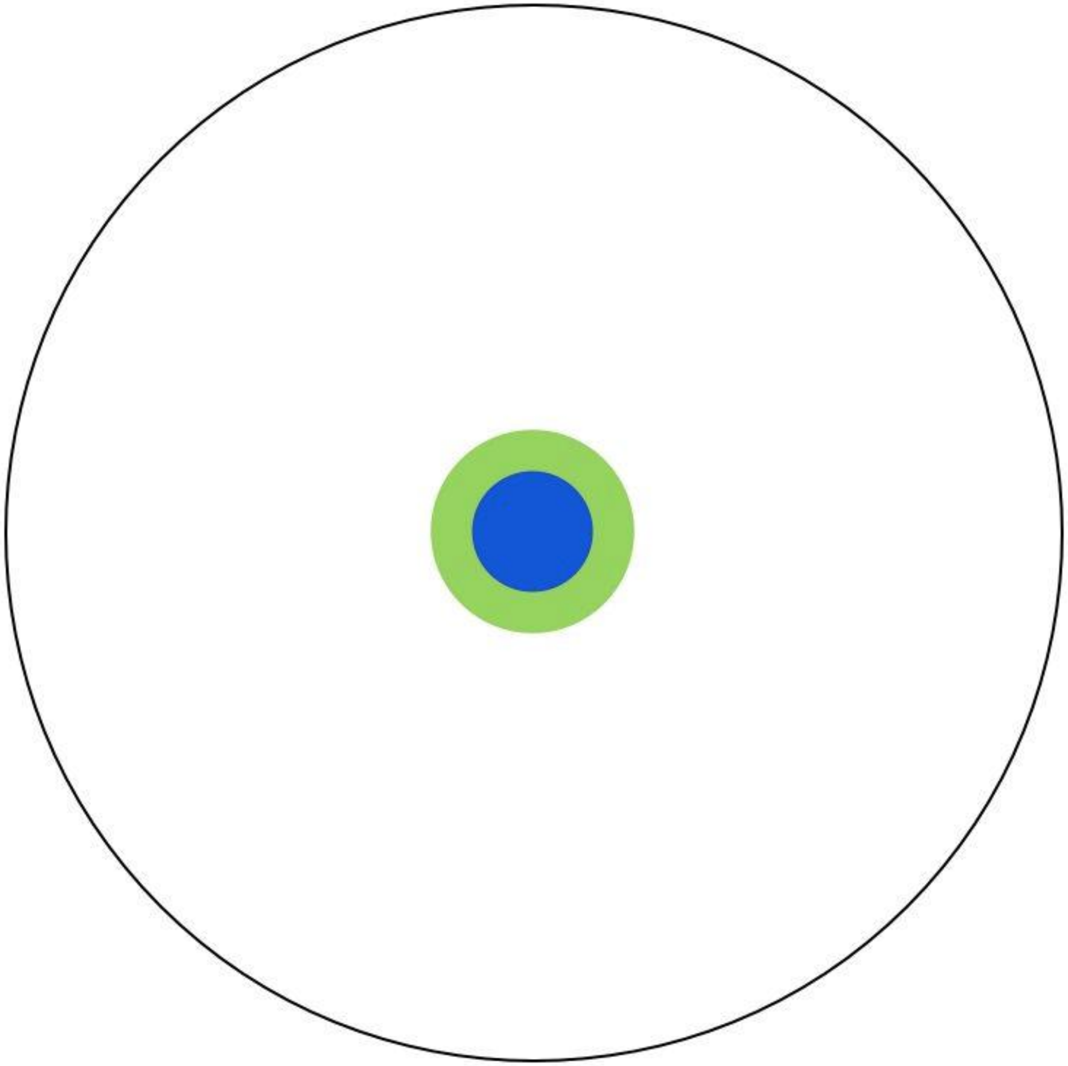
Penelitian yang Berkualitas Tinggi

Topik dan skalanya kecil, fokus, dalam, dan membawa pengaruh yang besar ke bidang penelitian kita

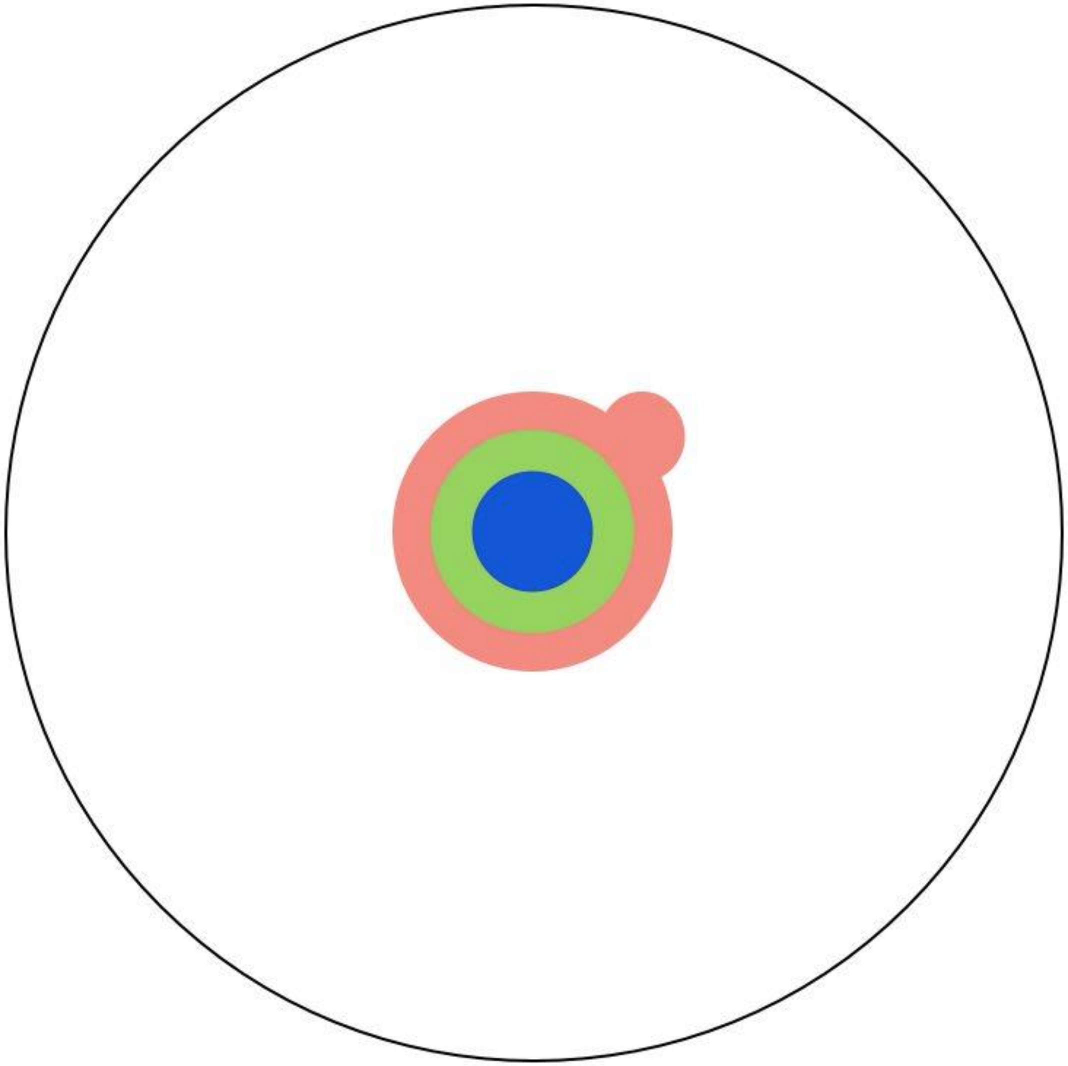
The Illustrated Guide to a Ph.D (Might, 2010)



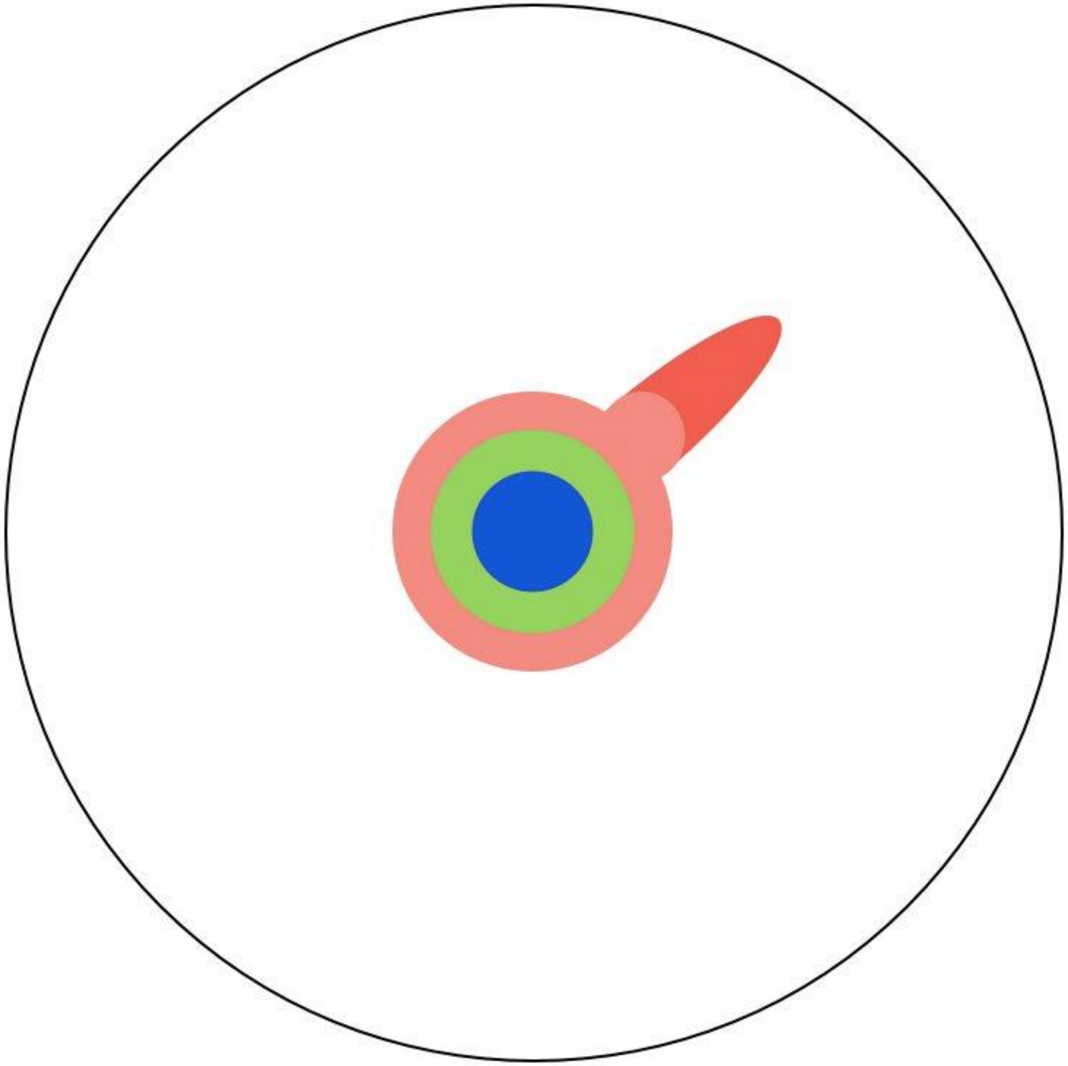
The Illustrated Guide to a Ph.D (Might, 2010)



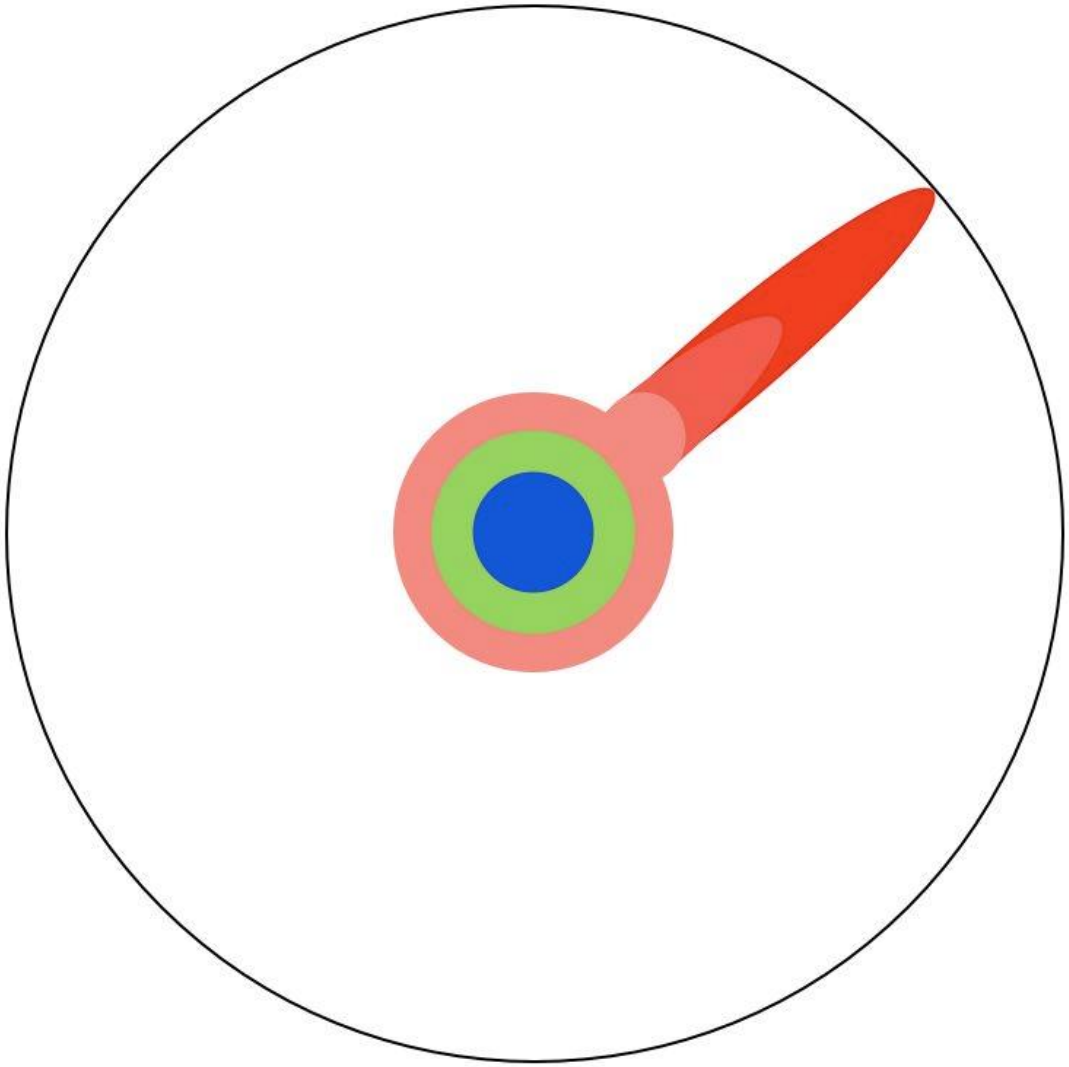
The Illustrated Guide to a Ph.D (Might, 2010)



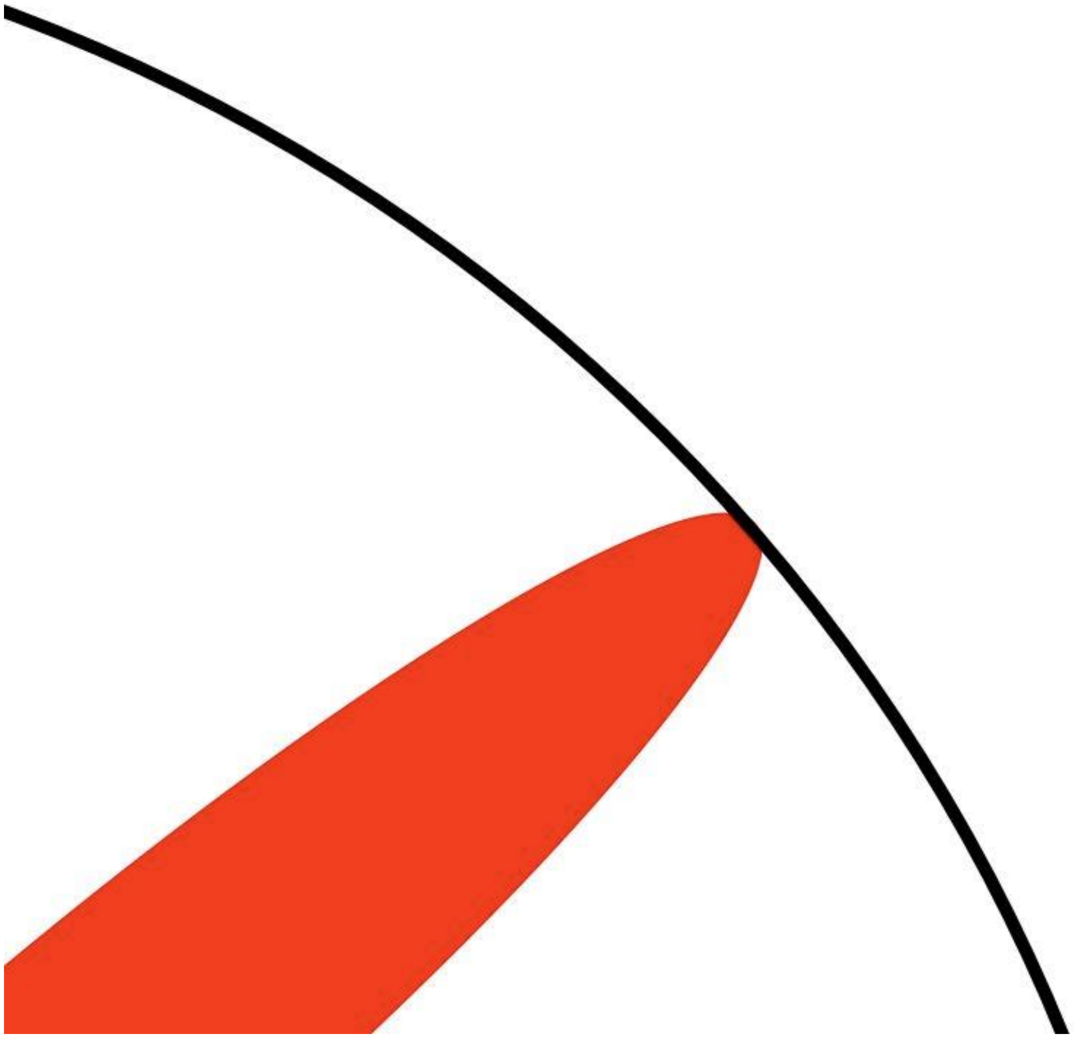
The Illustrated Guide to a Ph.D (Might, 2010)



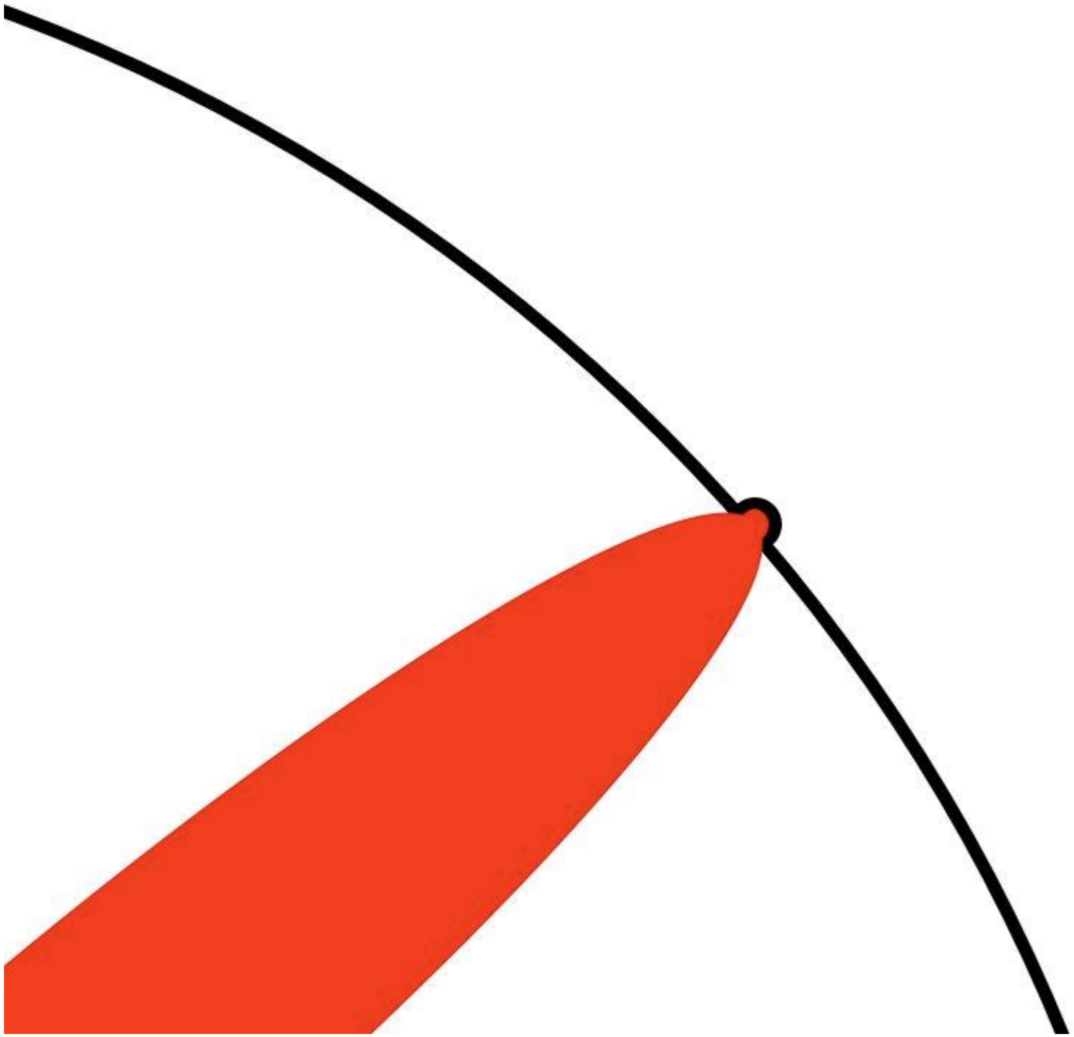
The Illustrated Guide to a Ph.D (Might, 2010)



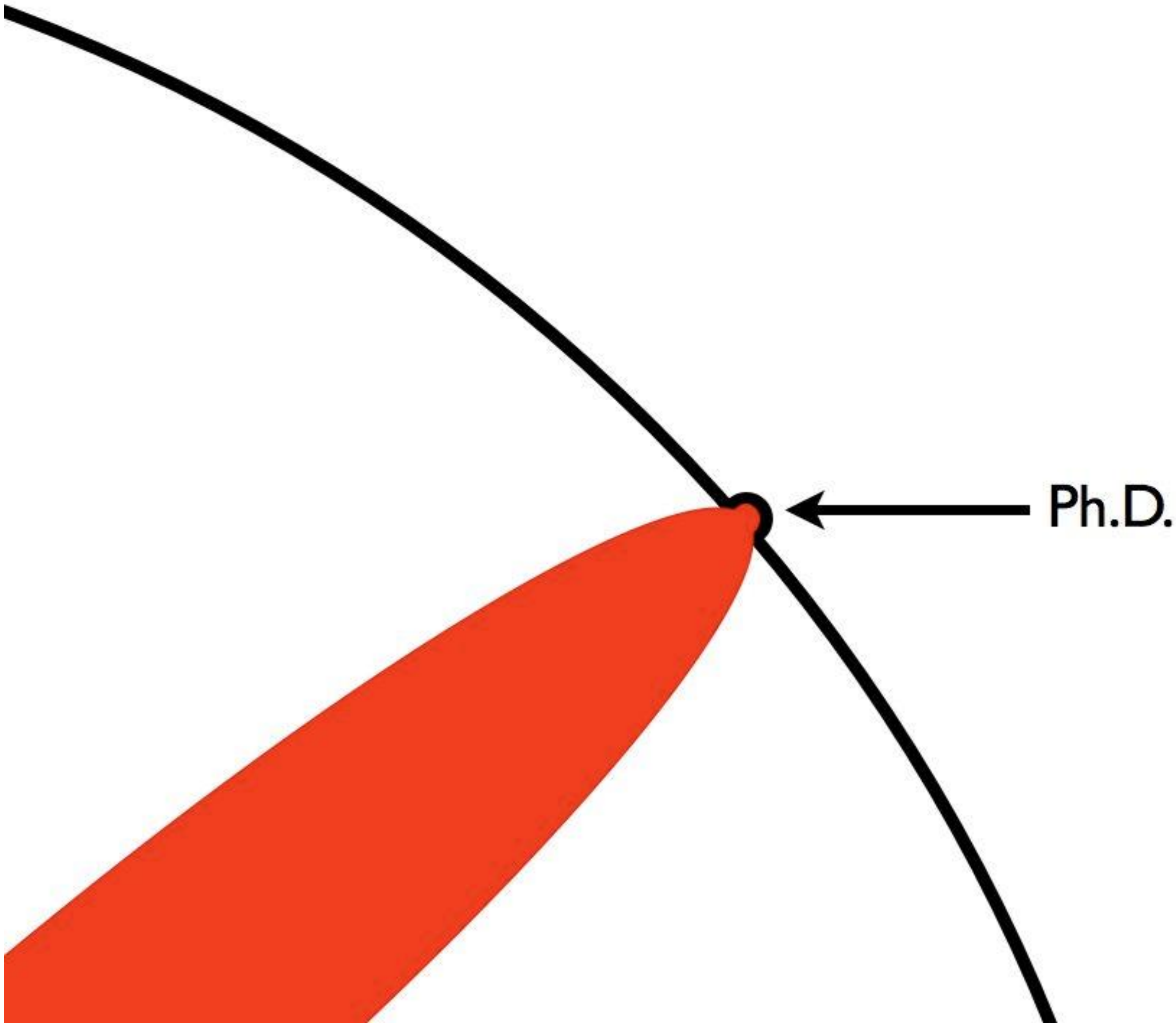
The Illustrated Guide to a Ph.D (Might, 2010)



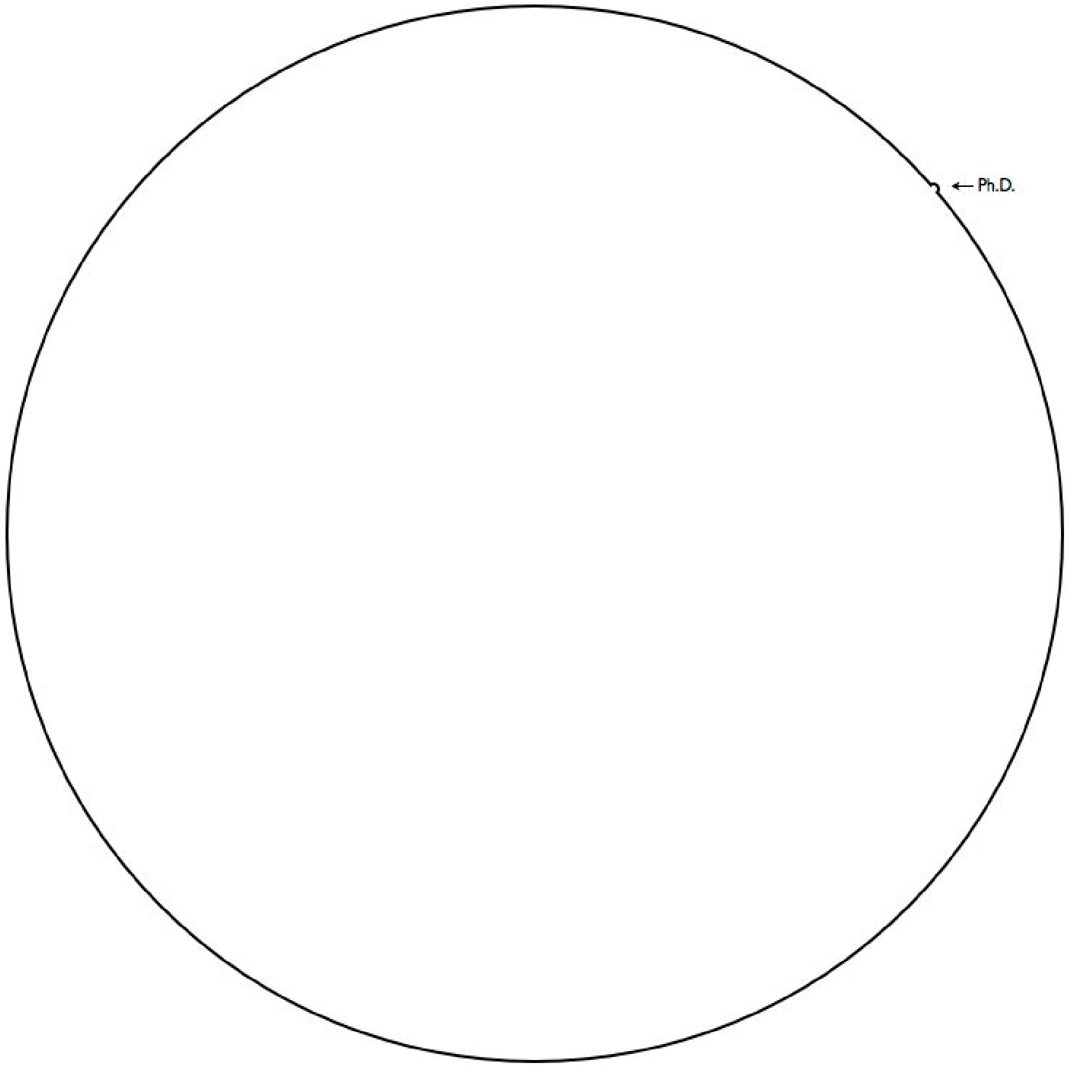
The Illustrated Guide to a Ph.D (Might, 2010)



The Illustrated Guide to a Ph.D (Might, 2010)



The Illustrated Guide to a Ph.D (Might, 2010)



Akademisi vs Technopreneur

Meja Indah

Meja Kuat

Meja Luas

- Technopreneur?
  1. Jual Produk
  2. Beri Nilai Tambah Produk
  3. Jadikan Aset, Jual Layanan

- Akademisi?
  - Pelajari, Preteli Komponen
  - Ciptakan Meja Baru yang Berbeda dengan 3 Meja Itu

Parameter Penelitian Yang Berkualitas

1. Penelitian yang dilakukan secara logis, sistematis, terencana, dan hasil penelitian divalidasi serta terukur *(Supino & Borer, 2012)*
2. Penelitian yang empiris, dilatarbelakangi oleh situasi yang riil, dengan data yang valid dan kongkrit *(Kothari, 2004)*
3. Penelitian yang memiliki kebaruan (*novelty*) yang bisa diwujudkan dalam berbagai bentuk *(Lichtfouse, 2013)*
4. Penelitian yang menghasilkan kontribusi ke pengetahuan yang memiliki orisinalitas yang tinggi *(Sahu, 2013)*
5. Penelitian yang menghasilkan kontribusi ke pengetahuan yang karakternya bisa digeneralisasi untuk obyek yang lain *(Dawson, 2009) (Supino & Borer, 2012)*
6. Penelitian yang bisa direplikasi oleh peneliti lain *(Kothari, 2004) (Runeson et al., 2012)*
7. Penelitian yang mendapatkan sitasi (citation) yang tinggi dari peneliti lain setelah dipublikasi dalam bentuk paper di jurnal ilmiah

2. Tahapan Penelitian

2.1 Tahapan Penelitian Umum

2.2 Tahapan Penelitian Computing

2.3 Tahapan Penelitian Computing Fokus Perbaikan Algoritma

2.1 Tahapan Penelitian Umum

Tahapan Penelitian Umum

1. Identifikasi Masalah
2. Perumusan Hipotesis
3. Pengujian Hipotesis dan Analisis
4. Kesimpulan

Tahapan Penelitian Umum vs Tesis

Tahapan Penelitian	Susunan Tesis
1. Identifikasi **Masalah**	1. Pendahuluan: - Latar Belakang - Rumusan Masalah - Tujuan Penelitian - Manfaat Penelitian
2. Perumusan **Hipotesis**	2. Landasan Teori: - Penelitian yang Berhubungan - Landasan Teori - Kerangka Pemikiran
3. **Pengujian Hipotesis** dan Analisis Hasil	3. Metodologi Penelitian: - Metode Penelitian - Metode Pengumpulan Data - Metode Analisis Data - Metode Pengukuran Penelitian
	4. Analisis Hasil dan Pembahasan
4. **Kesimpulan**	5. Kesimpulan dan Saran

2.2 Tahapan Penelitian Computing

Tahapan Penelitian Computing

Literature Review

1. Penentuan Bidang Penelitian (***Research Field***)
2. Penentuan Topik Penelitian (***Research Topic***)
3. Penentuan Masalah Penelitian (***Research Problem***)
4. Perangkuman Metode-Metode Yang Ada (***State-of-the-Art Methods***)
5. Penentuan Metode Yang Diusulkan (***Proposed Method***)
6. Evaluasi Metode Yang Diusulkan (***Evaluation***)
7. Penulisan Ilmiah dan Publikasi Hasil Penelitian (***Publications***)

***https://www.site.uottawa.ca/~bochmann/dsrg/how-to-do-good-research/*
***http://romisatriawahono.net/2013/01/23/tahapan-memulai-penelitian-untuk-mahasiswa-galau/*

1. Penentuan Bidang Penelitian

- Ingat kembali seluruh mata kuliah yang sudah kita terima di perkuliahan
- Bidang penelitian di disiplin computing:

Software Engineering	Data Mining
Image Processing	Computer Vision
Networking	Human Computer Interaction
Soft Computing	Information Retrieval
Bioinformatics	dsb

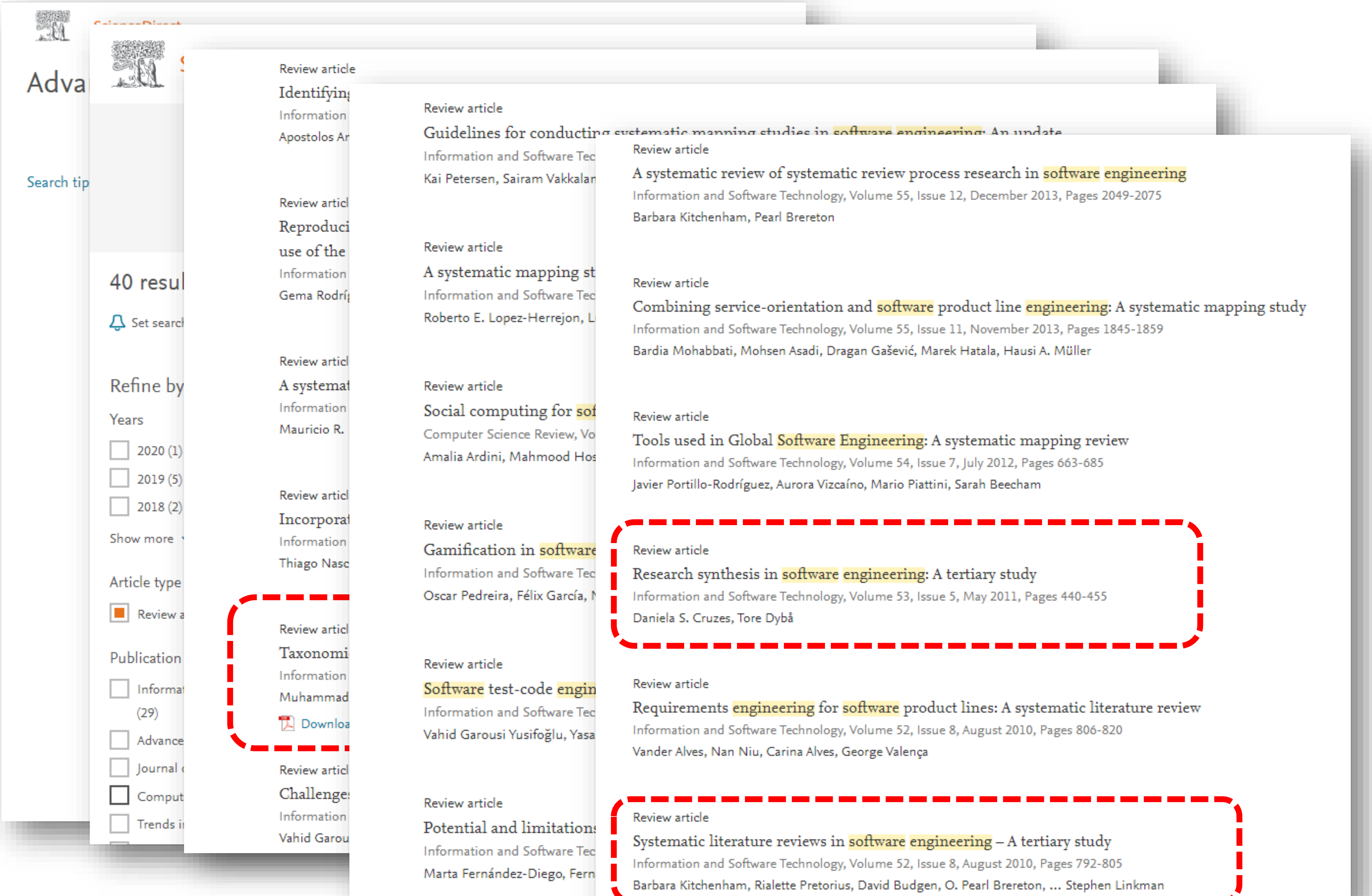
- Tentukan berdasarkan passion!

- Contoh: Saya memilih bidang Software Engineering (SE)

2. Penentuan Topik Penelitian

1. Searching di ScienceDirect.Com, Springerlink, IEEE Explore, Google (Scholar):
   - research trends challenge topics on NAMA BIDANG
2. Untuk mempercepat pembelajaran, temukan survey paper berbentuk Tertiery Study (SLR dari SLR), karena isinya sudah merangkumkan satu bidang penelitian
3. Lanjutkan penentuan topik penelitian dengan menemukan suvey/review paper (SLR, SMS), karena survey/review paper yang masuk jurnal terindeks pasti membahas satu topik penelitian

1. Cari Tertiery Study di Bidang Software Engineering


Review article
A systematic review of systematic review process research in software engineering
Information and Software Technology, Volume 55, Issue 12, December 2013, Pages 2049-2075
Barbara Kitchenham, Pearl Brereton

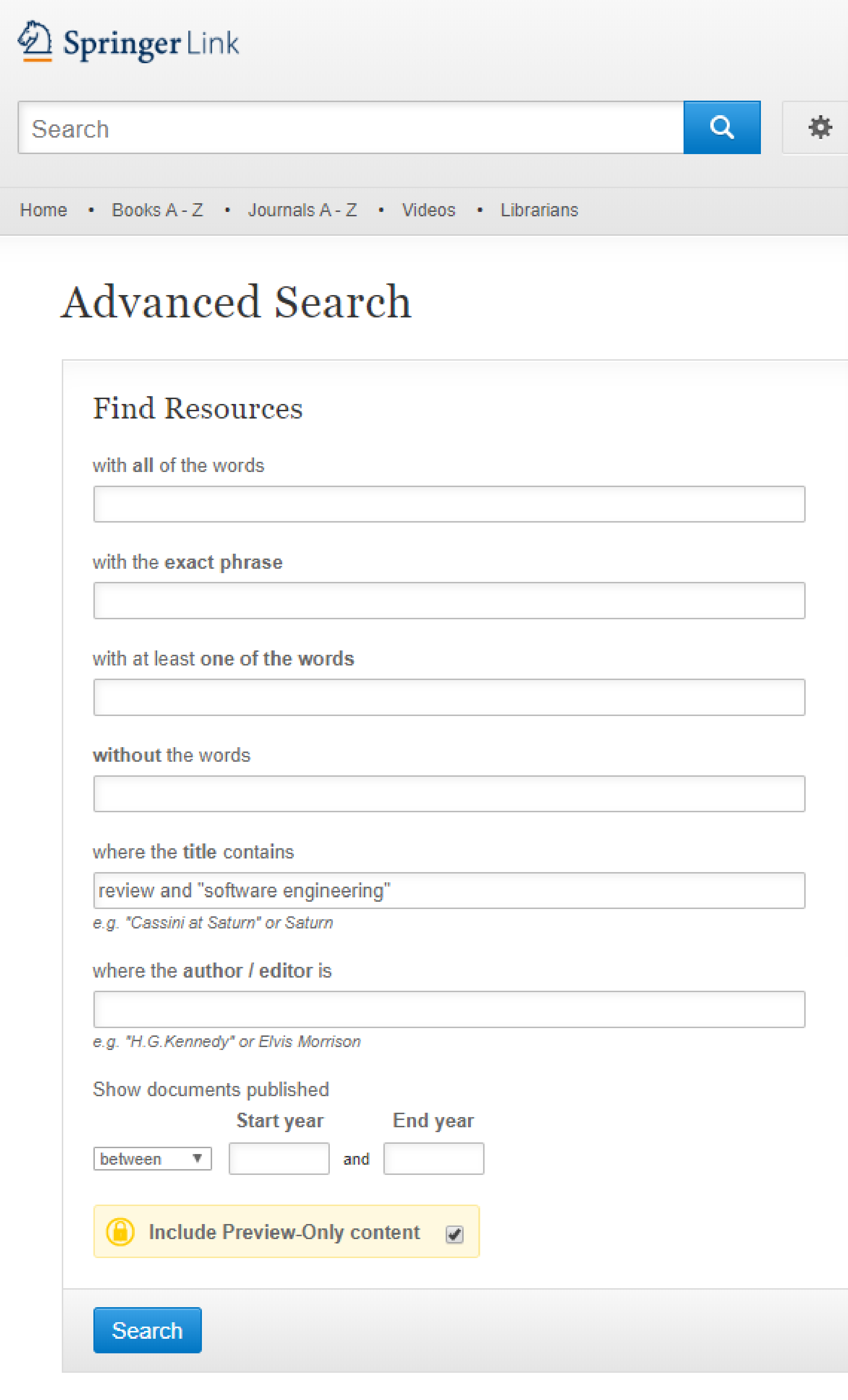
Review article
Combining service-orientation and software product line engineering: A systematic mapping study
Information and Software Technology, Volume 55, Issue 11, November 2013, Pages 1845-1859
Bardia Mohabbati, Mohsen Asadi, Dragan Gašević, Marek Hatala, Hausi A. Müller

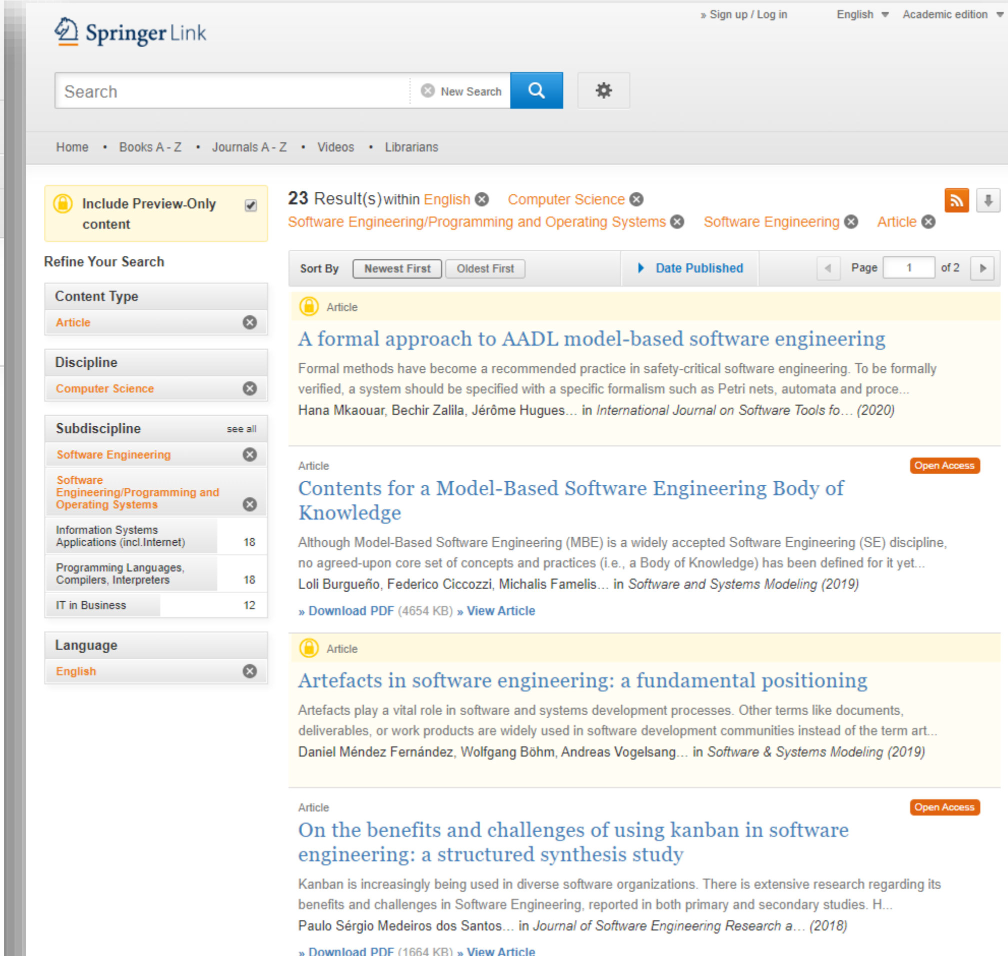
Review article
Tools used in Global Software Engineering: A systematic mapping review
Information and Software Technology, Volume 54, Issue 7, July 2012, Pages 663-685
Javier Portillo-Rodríguez, Aurora Vizcaíno, Mario Piattini, Sarah Beecham

Review article
Research synthesis in software engineering: A tertiary study
Information and Software Technology, Volume 53, Issue 5, May 2011, Pages 440-455
Daniela S. Cruzes, Tore Dybå

Review article
Requirements engineering for software product lines: A systematic literature review
Information and Software Technology, Volume 52, Issue 8, August 2010, Pages 806-820
Vander Alves, Nan Niu, Carina Alves, George Valença

Review article
Systematic literature reviews in software engineering – A tertiary study
Information and Software Technology, Volume 52, Issue 8, August 2010, Pages 792-805
Barbara Kitchenham, Rialette Pretorius, David Budgen, O. Pearl Brereton, ... Stephen Linkman


Springer Link
Search
Home • Books A - Z • Journals A - Z • Videos • Librarians
Advanced Search
Find Resources
with all of the words
with the exact phrase
with at least one of the words
without the words
where the title contains
review and "software engineering"
e.g. "Cassini at Saturn" or Saturn
where the author / editor is
e.g. "H.G.Kennedy" or Elvis Morrison
Show documents published
Start year
End year
between
and
Include Preview-Only content
Search


Springer Link
» Sign up / Log in
English
Academic edition
Search
New Search
Home • Books A - Z • Journals A - Z • Videos • Librarians
Include Preview-Only content
Refine Your Search
Content Type
Article
Discipline
Computer Science
Subdiscipline
see all
Software Engineering
Software Engineering/Programming and Operating Systems
Information Systems Applications (incl.Internet) 18
Programming Languages, Compilers, Interpreters 18
IT in Business 12
Language
English
23 Result(s) within English Computer Science Software Engineering/Programming and Operating Systems Software Engineering Article
Sort By Newest First Oldest First
Date Published
Page 1 of 2
Article
A formal approach to AADL model-based software engineering
Formal methods have become a recommended practice in safety-critical software engineering. To be formally verified, a system should be specified with a specific formalism such as Petri nets, automata and proce...
Hana Mkaouar, Bechir Zalila, Jérôme Hugues... in International Journal on Software Tools fo... (2020)
Article
Open Access
Contents for a Model-Based Software Engineering Body of Knowledge
Although Model-Based Software Engineering (MBE) is a widely accepted Software Engineering (SE) discipline, no agreed-upon core set of concepts and practices (i.e., a Body of Knowledge) has been defined for it yet...
Loli Burgueño, Federico Ciccozzi, Michalis Famelis... in Software and Systems Modeling (2019)
» Download PDF (4654 KB) » View Article
Article
Artefacts in software engineering: a fundamental positioning
Artefacts play a vital role in software and systems development processes. Other terms like documents, deliverables, or work products are widely used in software development communities instead of the term art...
Daniel Méndez Fernández, Wolfgang Böhm, Andreas Vogelsang... in Software & Systems Modeling (2019)
Article
Open Access
On the benefits and challenges of using kanban in software engineering: a structured synthesis study
Kanban is increasingly being used in diverse software organizations. There is extensive research regarding its benefits and challenges in Software Engineering, reported in both primary and secondary studies. H...
Paulo Sérgio Medeiros dos Santos... in Journal of Software Engineering Research a... (2018)
» Download PDF (1664 KB) » View Article

SLR in Software Engineering *(da Silva, 2011)*

3.1. Research questions

The five research questions (RQ) investigated in the SE were equivalent to the research questions used in the FE [19]. We performed minor adjustments and added subquestions as follows.

RQ1: How many SLRs were published between 1st January 2004 and 31st December 2009?
RQ1.1: How many SLRs were published between 1st January 2004 and 30th June 2008?
RQ1.2: How many SLRs were published between 1st July 2008 and 31st December 2009?

The subquestions of RQ1 investigate the development of SLRs in two separate periods. To answer RQ1.1, we used the results of OS/FE [18,19], whereas for RQ1.2, we performed the processes of search, selection, quality assessment, and data extraction defined in Sections 3.3–3.5. Similarly, we addressed the next questions considering the two time periods as we explicitly did for RQ1, searching for new evidence, combining with the results of the previous studies, and integrating all findings.

RQ2: What research topics are being addressed?
RQ3: Which individuals and organisations are most active in SLR-based research?

Information and Software Technology 53 (2011) 899–913

Contents lists available at ScienceDirect

Information and Software Technology

journal homepage: www.elsevier.com/locate/infsof

ELSEVIER

Six years of systematic literature reviews in software engineering: An updated tertiary study

Fabio Q.B. da Silva *, André L.M. Santos, Sérgio Soares, A. César C. França, Cleviton V.F. Monteiro, Felipe Farias Maciel

Centre for Informatics, UFPE, Cidade Universitária, 50.740-560 Recife, PE, Brazil

A R T I C L E I N F O

Article history:
Available online 23 April 2011

Keywords:
Systematic reviews
Mapping studies
Software engineering
Tertiary studies

A B S T R A C T

Context: Since the introduction of evidence-based software engineering in 2004, systematic literature review (SLR) has been increasingly used as a method for conducting secondary studies in software engineering. Two tertiary studies, published in 2009 and 2010, identified and analysed 54 SLRs published in journals and conferences in the period between 1st January 2004 and 30th June 2008.
Objective: In this article, our goal was to extend and update the two previous tertiary studies to cover the period between 1st July 2008 and 31st December 2009. We analysed the quality, coverage of software engineering topics, and potential impact of published SLRs for education and practice.
Method: We performed automatic and manual searches for SLRs published in journals and conference proceedings, analysed the relevant studies, and compared and integrated our findings with the two previous tertiary studies.
Results: We found 67 new SLRs addressing 24 software engineering topics. Among these studies, 15 were considered relevant to the undergraduate educational curriculum, and 40 appeared of possible interest to practitioners. We found that the number of SLRs in software engineering is increasing, the overall quality of the studies is improving, and the number of researchers and research organisations worldwide that are

Table 1
Manual search sources.

ACM Computer Surveys
ACM Transactions on Software Engineering Methodologies
Communications of the ACM
Empirical Software Engineering Journal
Evaluation and Assessment of Software Engineering
IEE Proceedings Software (now IET Software)
IEEE Software
IEEE Transactions on Software Engineering
Software Practice and Experience
Information and Software Technology
Int. Conference on Software Engineering
Int. Symposium on Empirical Software Engineering and Measurement
Journal of Systems and Software

SLR in Software Engineering *(da Silva, 2011)*

Table 7
Authors with three or more studies.

Authors	Country
Jørgensen	Norway
Guilherme Travassos	Brazil
Shepperd	UK
Tore Dyba	Norway
Muhammad Ali Babar	Ireland
Hannay	Norway
Sarah Beecham	UK
Sjøberg	Spain
Tony Gorschek	Sweden
Ambrosio Toval	Spain
Helen Sharp	UK
Hugh Robinson	UK
Juristo	Spain
Kampenes	Norway
Kitchenham	UK
Maya Daneva	The Netherlands
Moløkken-Østvold	Norway
Moreno	Spain
Nathan Baddoo	UK
Thelin	Sweden
Tracy Hall	UK

Table 2
Systematic literature reviews in software engineering between July 2008 and December 2009.

Study Ref. (*N* = 67)	Year	Quality score	Review type	Review focus	Review topic	Cited EBSE paper	Cited guidelines	Number primary studies	Practitioners guidelines	Paper type
[SE01]	2008	4	MS	SERT	Human Aspects	N	Y[d]	92	N	J
[SE02]	2008	4	SLR	RT	Knowledge Management	Y[a,b]	Y[d]	68	Y	J
[SE03]	2008	1.5	MS	RT	Research Topics in Software Engineering	N	N	691	N	J
[SE04]	2008	1	MS	SERT	Software Project Management	N	N	48	N	C
[SE05]	2008	4	MS	SERT	Agile Software Development	N	Y[f]	36	Y	J
[SE08]	2008	2	MS	SERT	Software Testing	N	Y[h]	14	Y	C
[SE09]	2008	2	MS	SERT	Requirements Engineering	N	Y[e]	240	N	C
[SE10]	2008	1	MS	SERT	Usability	N	Y[f]	51	Y	C
[SE11]	2008	2.5	MS	SERT	Software Process Improvement	N	Y[f]	50	Y	C
[SE12]	2008	1.5	MS	SERT	UML	N	Y[f]	33	N	C[i]
[SE13]	2008	1	SLR	RT	Distributed Software Development	N	N	12	N	C
[SE14]	2008	3	SLR	RQ	Usability	N	Y[e,h]	63	Y	J
[SE18]	2009	4	MS	SERT	Software Testing	N	Y[f]	35	N	J
[SE19]	2009	3.5	SLR	RT	Software Testing	Y[b]	Y[f,h]	64	N	J
[SE20]	2009	2.5	MS	SERT	Software Maintenance and Evolution	N	Y[e,f]	34	N	J
[SE21]	2009	2.5	MS	SERT	Requirements Engineering	N	Y[d]	58	N	C
[SE22]	2009	2	SLR	RQ	Agile Software Development	Y	Y[d,f]	9	N	C
[SE23]	2009	2	MS	RQ	Design Patterns	Y	Y[f]	4	N	C
[SE24]	2009	2	MS	SERT	Software Maintenance and Evolution	N	Y[d]	12	Y	C
[SE25]	2009	2	MS	SERT	Risk Management	N	Y[e,g]	80	N	J
[SE26]	2009	2	MS	SERT	Software Fault Prediction	N	N	74	N	J
[SE27]	2009	3	MS	SERT	Software Product Line	N	Y[f]	34	N	C
[SE28]	2009	3	MS	SERT	Software Product Line	Y	Y[f]	97	N	C
[SE29]	2009	1.5	MS	SERT	Requirements Engineering	N	N	46	N	C[i]
[SE30]	2009	3	MS	SERT	Software Maintenance and Evolution	N	Y[d]	176	N	J
[SE32]	2009	1.5	SLR	RT	Empirical Research Methods	Y	Y[d]	16	N	C[i]
[SE33]	2009	1.5	SLR	RQ	Software Security	N	Y[f]	64	N	C
[SE34]	2009	2.5	MS	SERT	Empirical Research Methods	N	N	8	N	J
[SE35]	2008	3	SLR	RQ	Software Testing	N	Y[f,h]	28	N	C
[SE36]	2009	3.5	MS	SERT	Human Aspects	Y	Y[d]	92	N	J
[SE37]	2009	3	MA	RQ	Agile Software Development	Y	Y[f]	18	Y	J
[SE38]	2009	3	MS	SERT	Context Aware Systems	N	N	237	N	J
[SE39]	2009	3.5	SLR	RQ	Software Maintenance and Evolution	N	Y[d]	18	Y	C
[SE40]	2009	4	MS	SERT	Distributed Software Development	N	Y[f]	20	Y	C
[SE42]	2009	2.5	SLR	RT	Requirements Engineering	N	Y[f]	97	Y	J
[SE43]	2009	2.5	MS	SERT	Distributed Software Development	N	Y[f]	78	Y	J
[SE44]	2009	2	MS	SERT	Distributed Software Development	N	Y[f]	98	Y	C
[SE45]	2009	2	MS	SERT	Distributed Software Development	N	Y[d]	122	Y	C
[SE46]	2009	4	MS	SERT	Software Product Line	N	Y[e]	89	N	J
[SE47]	2009	3	MS	SERT	Software Product Line	N	Y[d]	23	N	C

SLR in Software Engineering *(Kitchenham, 2010)*

Information and Software Technology 52 (2010) 792–805
Contents lists available at ScienceDirect

Table 1
Additional software engineering SLRs published from 1st January 2004 to 30th June 2008 (studies above the double lines were published before July 1st 2007, studies below the double lines were published after June 30th 2007).

Study ref.	Review focus	Quality total score	Year	Cited EBSE paper	Cited guidelines	Paper type	Number primary studies	Practitioner guidelines	Review topic
[32]	RQ	2.5	2005	No	Yes	Conference	8	N	Cost estimation – impact of clients on estimate accuracy
[33]	RQ	2	2005	No	No	Journal	70	Y	Cost estimation – Guidelines for estimating uncertainty
[36]	RT	2.5	2005	No	Yes[a]	Workshop	50	N	Cost estimation – data sets used to evaluate models
[39]	RT	1.5	2005	No	No	Workshop	119	N	Evidence produced by empirical software engineers
[40]	RT	2	2005	No	Yes	Conference	13	N	Classifying context in SE experiments
[41]	SERT	2.5	2005	No	No	Conference	105	N	Mobile systems development
[34]	RQ	3.5	2006	Yes	Yes	Conference	26	Y	Requirements elicitation techniques
[37]	SERT	1.5	2006	No	No	Workshop	57	N	Conceptual model of outsourcing
[42]	SERT	1	2006	No	No	Technical report	750	N	Software architecture
[35]	SERT	1.5	2007	Yes	No	Workshop	653	N	Cost estimation challenges
[38]	SERT	1.5	2007	No	No	Journal	80	N	Approaches for mining software repositories in the context of evolution
[43]	SERT	1.5	2007	No	No	Book chapter (working conference)	4089	N	Requirements Engineering publications
[44]	RT	1.5	2007	No	No	Book chapter (workshop)	133	N	Evidence produced by empirical software engineers
[45]	SERT	2	2007	No	No	Book chapter (working conference)	155	N	Developing open source software – analysis of research
[11]	RT	2.5	2007	No	Yes	Journal	103	N	Empirical software engineering – effect size
[16]	SERT	2.5	2007	No	Yes	Conference	138	No	Software design – Object-oriented
[17]	RQ	4	2007	No	Yes	Conference	10	No	Cost estimation-local vs. global estimation models
[19]	RQ	3.5	2007	No	Yes	Book chapter (conference)	5	N	Software development process – tailoring and introduction of rational unified process
[20]	RQ	2.5	2007	No	Yes	Journal	11	N	Reuse – economic benefits
[21]	SERT	1	2007	No	No	Journal	137	N	Tool integration – a research agenda
[24]	SERT	3.5	2007	No	Yes	Conference	53	N	Web application development – design for accessibility
[10]	RQ	2.5	2008	No	Yes	Journal	103	N	Empirical software engineering – the value of laboratory experiments
[15]	SERT	2.5	2008	No	Yes	Conference	28	No	Re-engineering – multi-channel access
[18]	RQ	1.5	2008	No	No	Book chapter (conference)	21	N	Metrics – measurement programme

SLR in Software Engineering *(Kitchenham, 2010)*

Table 5
Distribution of SLRs over the SE sections of the University curriculum guidelines.

Section	Number of sub-sections	Number of sub-topics	January 1st 2004 to June 30th 2007 manual search [12]		January 1st 2004 to June 30th 2007 additional papers (broad search)		July 1st 2007 to June 30th		Total SLRs	Total sub-topics
			SLRs	Num subs addressed	SLRs	Num subs addressed	SLRs	Num subs addressed		
Software modeling and analysis	7	41			[34]	1	[22,25]	2	3	3
Software design	7	37	[56]	0	[42]	0	[23,24]	0	4	0[a]
Software validation and verification	40	5	[61,65,66][b,c]	5			[31]	0	5	5
Software evolution	2	13								
Software process	2	14			[37]	1	[19]	1	2	2
Software quality	5	28	[57]	1				–	1	1
Software management	5	32	[58–60,62–64]	1	[32,33]	1	[17,18,48]	2	11	2
Computing essentials	4	41	[67]	1					1	1
Mathematical and engineering fundamentals	3	22					[22,48]	2	2	2
System and application specialties	42				[41]	1	[15]	1	2	1
Total		233	12	8	6	4	11[d]	8	29	17

[a] The papers addressed a general topic, not a specific technique.
[b] This paper addressed four sub-topics of the subsection testing.
[c] This paper addressed testing methods and inspection methods.
[d] Paper [48] addressed two topics.

SLR in Software Engineering (*Zhang, 2013*)

- RQ1. What is the value of SLR for SE? Why did (or did not) SE researchers do SLRs?
- RQ2. What SE topics have been addressed by what types of SLRs? What has the influence of SLRs been in SE research?
- RQ3. How did SE researchers perform SLRs (in terms of, for example, rigour and effort)?

Information and Software Technology 55 (2013) 1341–1354

Contents lists available at SciVerse ScienceDirect

Information and Software Technology

journal homepage: www.elsevier.com/locate/infsof

ELSEVIER

Systematic reviews in software engineering: An empirical investigation

He Zhang [a,*], Muhammad Ali Babar [b]

[a] University of New South Wales, Australia
[b] IT University of Copenhagen, Denmark

ARTICLE INFO

Article history:
Available online 4 October 2012

Keywords:
Systematic (literature) reviews
Evidence-based software engineering
Research methodology
Methodology adoption
Mixed-methods research
Tertiary study

Background: Systematic Literature Reviews (SLRs) have gained significant popularity among Software Engineering (SE) researchers since 2004. Several researchers have also been working on improving the scientific and methodological infrastructure to support SLRs in SE. We argue that there is also an apparent and essential need for evidence-based body of knowledge about different aspects of the adoption of SLRs in SE.
Objective: The main objective of this research is to empirically investigate the adoption, value, and use of SLRs in SE research from various perspectives.
Method: We used mixed-methods approach (systematically integrating tertiary literature review, semi-structured interviews and questionnaire-based survey) as it is based on a combination of complementary research methods which are expected to compensate each others' limitations.
Results: A large majority of the participants are convinced of the value of using a rigourous and systematic methodology for literature reviews in SE research. However, there are concerns about the required time and resources for SLRs. One of the most important motivators for performing SLRs is new findings and inception of innovative ideas for further research. The reported SLRs are more influential compared to the traditional literature reviews in terms of number of citations. One of the main challenges of conducting SLRs is drawing a balance between methodological rigour and required effort.

SLR in Software Engineering (*Zhang, 2013)*

Table 4
Overview of SLR studies and publications in software engineering research by topic and year.

Rank	Review topics	2004	2005	2006	2007	2008	2009	2010	Sum
1	Global development					◊◊◊	◆[a]◊◊▽	◆[a] ◆[a]◊◊◊◊◊◊◊▽	13/16
1	Cost estimation	◆	◆◊◊▽	◆◊	◆◆◆[a]◊◊		◊▽		13/14
3	Requirement engineering			◊	◊	◊	◆◆◆◊◊[a]▽▽	◆◆◆	12/13
4	Empirical methods		◆◊▽	◆	◆◆▽	◆▽	◆▽		11/11
5	Agile development					◆	◆◊◊	◆◊◊◊▽▽	10/10
6	Inspection and testing	◆		◆	▽	◆[a]	◆◆[a]◊◊	◆◆◊	9/11
6	Software product lines					◊▽	◆◊◊◊◊◊	◆	9/9
8	Software process improvement				◆◊	◆◆◊	◊	◆◊	8/8
9	Software architecture			▽		▽	◆[a]▽	◆◊	5/6
10	Software process modelling			▽		◊	◊	◊◊[a]	4/5
10	Open source development						▽◊[a]	◆◊◊	4/5
10	Software measurement			◊		◆◊		◆	4/4
10	Program analysis						◆◆◊	◊	4/4
14	Model-based development					◊	◆[a]	◊▽	3/4
14	Tertiary study						◆	◆[a]◊▽	3/4
14	Software maintenance				◆	▽	◊		3/3
14	Software tools				◆▽		◆		3/3
14	Software security					◆◊		◆	3/3
14	Web engineering		◊		◊	▽			3/3
14	Software outsourcing						◆◊◊		3/3
14	Human-aspects (e.g., motivations)					◆	◆◆		3/3
22	Software design				◊		◊		2/2
22	Unified modelling language					▽	◆		2/2
22	Software evolution				◆			◊	2/2
22	Aspect-oriented programming						◊	◆	2/2
22	Business process			◊				◆	2/2
22	SE research in general	◆				◆			2/2
	Other topics		◊		◆	◆	◆	◆◆	6/6
Total	(*new/all SLR reports*)	3/3	9/9	9/9	18/19	25/26	44/50	40/44	**148/160**

◆ journal paper or book chapter, ◊ full conference paper, ▽ workshop or short paper.
[a] Update/extension of previous SLR report.

Research Synthesis in Software Engineering (Cruzes, 2011)

Information and Software Technology 53 (2011) 440–455

Contents lists available at ScienceDirect

oftware Technology

.elsevier.com/locate/infsof

ng: A tertiary study

ntrasting evidence from multiple studies is necessary to build knowledge and
e empirical support for a phenomenon. Therefore, research synthesis is at the
erprise in the software engineering discipline.
this article is to contribute to a better understanding of the challenges in syn-
ring research and their implications for the progress of research and practice.
f journal articles and full proceedings papers from the inception of evidence-
; was performed to assess the types and methods of research synthesis in sys-
e engineering.
f the 49 reviews included in the study did not contain any synthesis. Of the
ynthesis, two thirds performed a narrative or a thematic synthesis. Only a
nonstrated a robust, academic approach to research synthesis.
that, despite the focus on systematic reviews, there is limited attention paid to
vare engineering. This trend needs to change and a repertoire of synthesis
tegral part of systematic reviews to increase their significance and utility for

Table 6
Main topic areas in SE systematic reviews.

Main topic area	Studies
Agile software development	S5, S12
Aspect-oriented programming	S45
Distributed software development	S15, S19, S47
Domain analysis	S22
Estimation models	S16, S21, S35, S43
Experimental methods in SE	S6, S17, S18, S38, S44
Global software engineering	S31
Knowledge management in SE	S3
Motivation in SE	S2, S28
Product lines software development	S20, S25, S33
Requirements engineering	S4, S14, S26, S29, S32, S48
Reuse	S36
Software design	S8, S23, S30, S39, S42
Computer games	S49
Software maintainability	S34
Software measurement	S9, S40
Software process	S27, S41
Technology acceptance model	S46
Testing	S1, S7
Theory use in SE	S10, S11
Web development	S13, S24, S37

Citation & Topics in Software Engineering (Garousi, 2016)

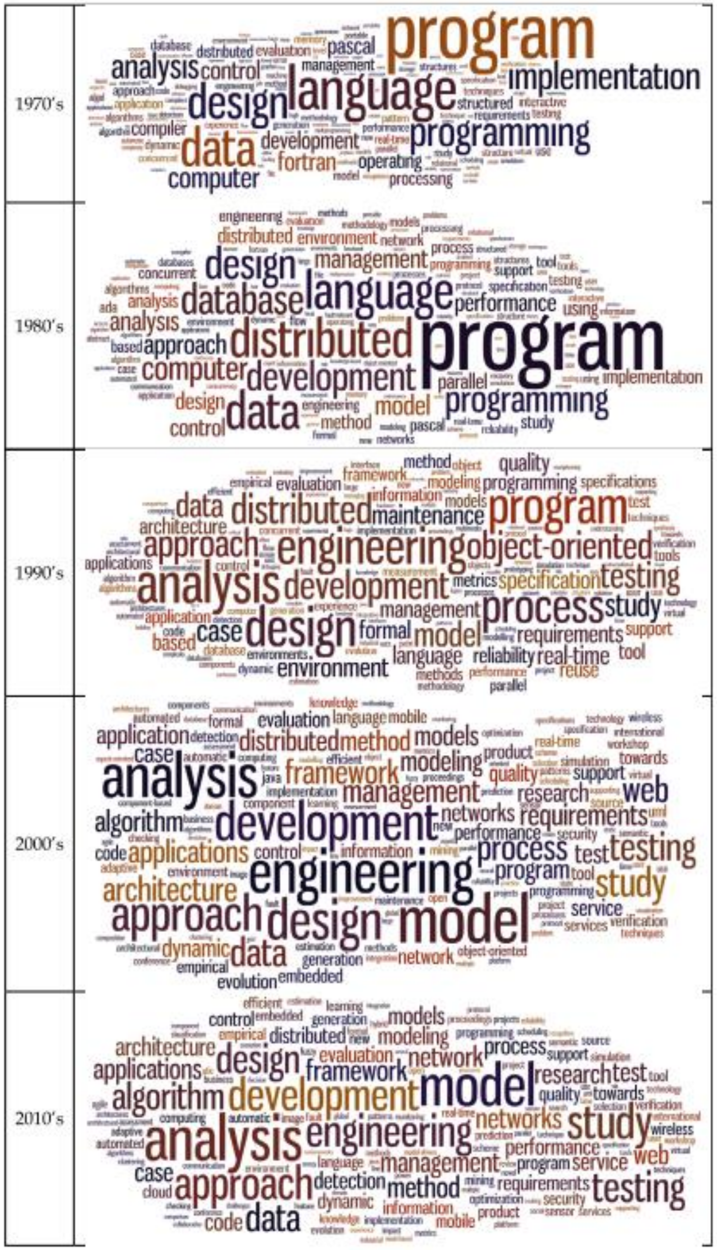


Fig. 10 – Focus areas of SE papers in each decade.

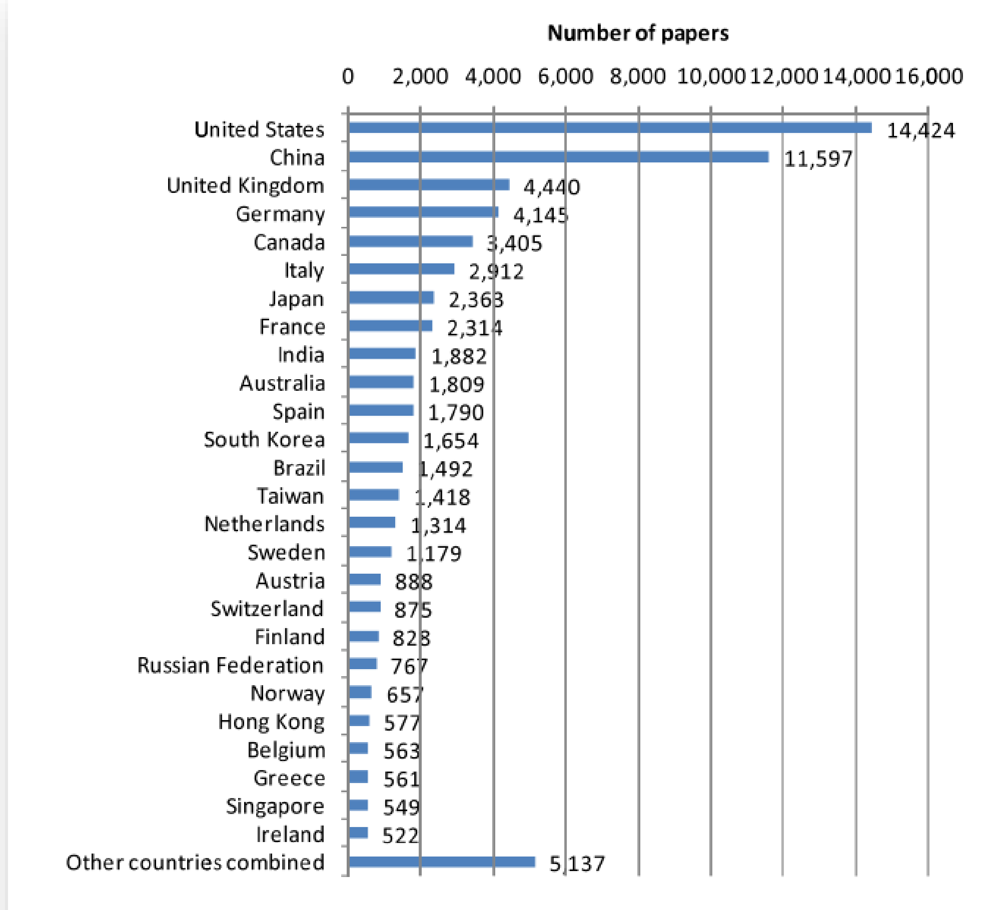


Fig. 13 – Ranking of the countries with more than 500 contributions.

10 Most Probable Terms in the Topics

Table 10 – Ten most probable terms in the topics with the highest number of papers.

Given topic name	Requirements engineering ($n = 3096$)	Database ($n = 2601$)	Software effort estimation ($n = 2601$)	Design patterns ($n = 2172$)	Incoherent topic ($n = 2155$)
1	"requirements"	"database"	"software"	"design"	"using"
2	"engineering"	"data"	"estimation"	"patterns"	"analysis"
3	"elicitation"	"processing"	"effort"	"implementation"	"time"
4	"nonfunctional"	"databases"	"cost"	"architectural"	"model"
5	"role"	"xml"	"measurement"	"pattern"	"real"
6	"tracing"	"relational"	"models"	"matching"	"behavior"
7	"goals"	"query"	"functional"	"style"	"functions"
8	"managing"	"objects"	"function"	"decisions"	"world"
9	"legal"	"efficient"	"size"	"principles"	"structure"
10	"goal-oriented"	"spatial"	"point"	"rationale"	"paper"

Table 11 – Ten most probable terms in the topics with the highest amount of citations per paper per year.

Given topic name	Model checking ($n = 686$)	Test generation (=923)	Source code ($n = 988$)	Automated testing ($n = 785$)	Systematic reviews ($n = 653$)
1	"model"	"test"	"code"	"testing"	"software"
2	"checking"	"generation"	"source"	"automated"	"systematic"
3	"transformation"	"automatic"	"open"	"model based"	"review"
4	"driven"	"coverage"	"projects"	"regression"	"challenges"
5	"transformations"	"automated"	"changes"	"mutation"	"survey"
6	"probabilistic"	"selection"	"usage"	"random"	"future"
7	"Markov"	"generating"	"documentation"	"strategies"	"issues"
8	"bounded"	"cases"	"API"	"GUI"	"mapping"
9	"graph"	"suite"	"detecting"	"conformance"	"results"
10	"properties"	"tests"	"clones"	"techniques"	"approaches"

2. Cari SLR dari Topik Penelitian yang Dipilih

- Setelah kita paham beberapa topik penelitian di bidang software engineering dari *Tertiery Study (SLR dari SLR)*
- Langkah berikutnya, kita kumpulkan seluruh SLR dengan keyword topik seperti di paper *Tertiery Study (SLR dari SLR)*
- Lanjutkan dengan mengejar seluruh SLR dari topik yang kita akan angkat pada penelitian kita

Keyword harus masuk Title Paper, Pilih Review, dan Journal di Bidang Computing

- Global Software Engineering
- Requirement Engineering
- Self Adaptive Systems
- Service Oriented Architecture
- Software Architecture
- Software Construction
- Software Cost Effort Estimation
- Software Defect Prediction
  - Arisholm - fault prediction models - 2010
  - Catal - A systematic review of software fault prediction studies
  - Catal - Software fault prediction A literature review and curren
  - Hall - Fault Prediction Performance in Software Engineering -
  - mahmood - reproducibbility of SDP - 2018
  - Radjenovic - Software fault prediction metrics - 2013
  - Shepperd - SLR of Unsupervised Learning for SDP - 2020
  - Strate - Software Defect Reporting - 2013
  - Wahono - SLR of Software Defect Prediction - 2015

Title

Requirement Engineering

Volume(s) | Issue(s) | Page(s) | ISSN or ISBN

References

Article types ?

- [x] Review artic
- [ ] Research ar
- [ ] Encyclopedi
- [ ] Book chapte
- [ ] Conference

6 results

Set search alert

Refine by:

Years
- [ ] 2016 (1)
- [ ] 2015 (1)
- [ ] 2010 (3)
- [ ] 2009 (1)

Custom range

Show less

Article type
- [x] Review articles (6)

Publication title
- [x] Information and Software Technology (3)
- [x] Computer Standards & Interfaces (2)
- [x] Computers in Human Behavior (1)
- [ ] Biomaterials (1)
- [ ] Biocatalysis and Agricultural Biotechnology (1)
- [ ] Journal of Pharmaceutical Sciences (1)

Show less

Review article

Requirements engineering for safety-critical systems: A systematic literature review

Information and Software Technology, Volume 75, July 2016, Pages 71-89

Luiz Eduardo G. Martins, Tony Gorschek

Review article

A systematic literature review on agile requirements engineering practices and challenges

Computers in Human Behavior, Volume 51, Part B, October 2015, Pages 915-929

Irum Inayat, Siti Salwah Salim, Sabrina Marczak, Maya Daneva, Shahaboddin Shamshirband

Want a richer search experience?

Sign in for additional filter options, multiple article downloads, and more.

Sign in >

Review article

Requirements engineering for software product lines: A systematic literature review

Information and Software Technology, Volume 52, Issue 8, August 2010, Pages 806-820

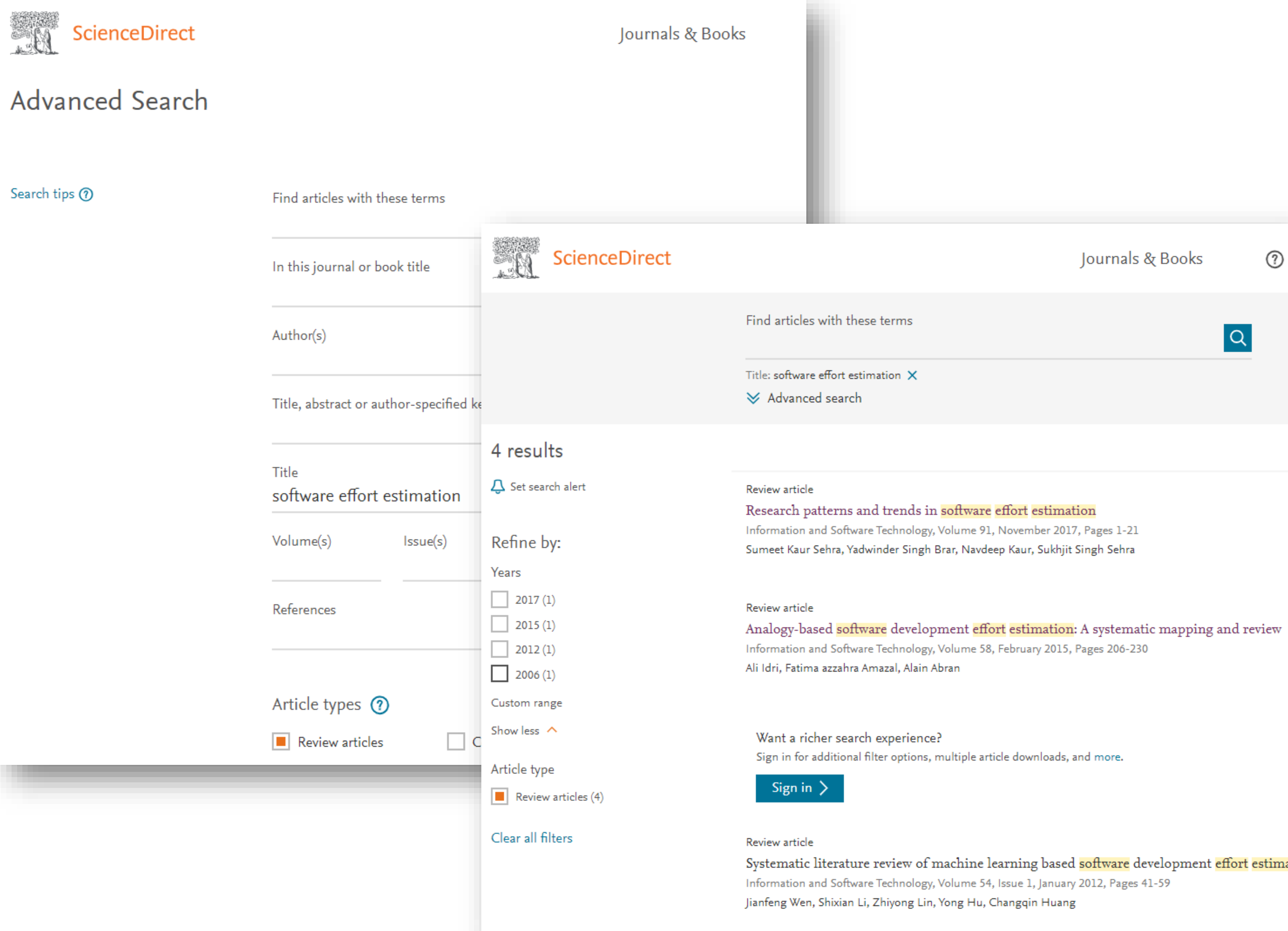
Vander Alves, Nan Niu, Carina Alves, George Valença

Review article

A systematic review of security requirements engineering

Computer Standards & Interfaces, Volume 32, Issue 4, June 2010, Pages 153-165

Daniel Mellado, Carlos Blanco, Luis E. Sánchez, Eduardo Fernández-Medina


ScienceDirect
Journals & Books
Advanced Search
Search tips
Find articles with these terms
In this journal or book title
Author(s)
Title, abstract or author-specified ke
Title
software effort estimation
Volume(s)
Issue(s)
References
Article types
Review articles
ScienceDirect
Journals & Books
Find articles with these terms
Title: software effort estimation
Advanced search
4 results
Set search alert
Refine by:
Years
2017 (1)
2015 (1)
2012 (1)
2006 (1)
Custom range
Show less
Article type
Review articles (4)
Clear all filters
Review article
Research patterns and trends in software effort estimation
Information and Software Technology, Volume 91, November 2017, Pages 1-21
Sumeet Kaur Sehra, Yadwinder Singh Brar, Navdeep Kaur, Sukhjit Singh Sehra
Review article
Analogy-based software development effort estimation: A systematic mapping and review
Information and Software Technology, Volume 58, February 2015, Pages 206-230
Ali Idri, Fatima azzahra Amazal, Alain Abran
Want a richer search experience?
Sign in for additional filter options, multiple article downloads, and more.
Sign in
Review article
Systematic literature review of machine learning based software development effort estima
Information and Software Technology, Volume 54, Issue 1, January 2012, Pages 41-59
Jianfeng Wen, Shixian Li, Zhiyong Lin, Yong Hu, Changqin Huang

Are You Sure This is a Topic?

1276 IEEE TRANSACTIONS ON SOFTWARE ENGINEERING, VOL. 38, NO. 6, NOVEMBER/DECEMBER 2012

A Systematic Literature Review on Fault Prediction Performanc in Software Engineering

Tracy Hall, Sarah Beecham, David Bowes, David Gray, and Stev

Abstract—*Background*: The accurate prediction of where faults are likely to occur in code can help direct te improve the quality of software. *Objective*: We investigate how the context of models, the independent va modeling techniques applied influence the performance of fault prediction models. *Method*: We used a sys identify 208 fault prediction studies published from January 2000 to December 2010. We synthesize the results of 36 studies which report sufficient contextual and methodological information according to the crit *Results:* The models that perform well tend to be based on simple modeling techniques such as Naive Ba Combinations of independent variables have been used by models that perform well. Feature selection h combinations when models are performing particularly well. *Conclusion:* The methodology used to build mo to predictive performance. Although there are a set of fault prediction studies in which confidence is possibl that use a reliable methodology and which report their context, methodology, and performance comprehe

Index Terms—Systematic literature review, software fault prediction

1 INTRODUCTION

THIS Systematic Literature Review (SLR) aims to identify and analyze the models used to predict faults in source code in 208 studies published between January 2000 and December 2010. Our analysis investigates how model performance is affected by the context in which the model was developed, the independent variables used in the model, and the technique on which the model was built. Our results enable researchers to develop prediction models based on best knowledge and practice across many previous studies. Our results also help practitioners to make effective decisions on prediction models most suited to their context.

Fault[1] prediction modeling is an important area of research and the subject of many previous studies. These studies typically produce fault prediction models which allow software engineers to focus development activities on fault-prone code, thereby improving software quality and

making better use of resources models published are complex date comprehensive picture o prediction exists. Two previou been performed in [1] and [2] these reviews in the following

- *Timeframes*. Our review because it includes stu 2010. Fenton and Neil co software fault predictio Catal and Diri's [2] rev between 1990 and 2007.
- *Systematic approach*. W Charters [3] original and rigorous procedures for conducting systematic reviews. Catal and Diri did not report on how they sourced their studies, stating that they adapted Jørgensen and Shepperd's [4]

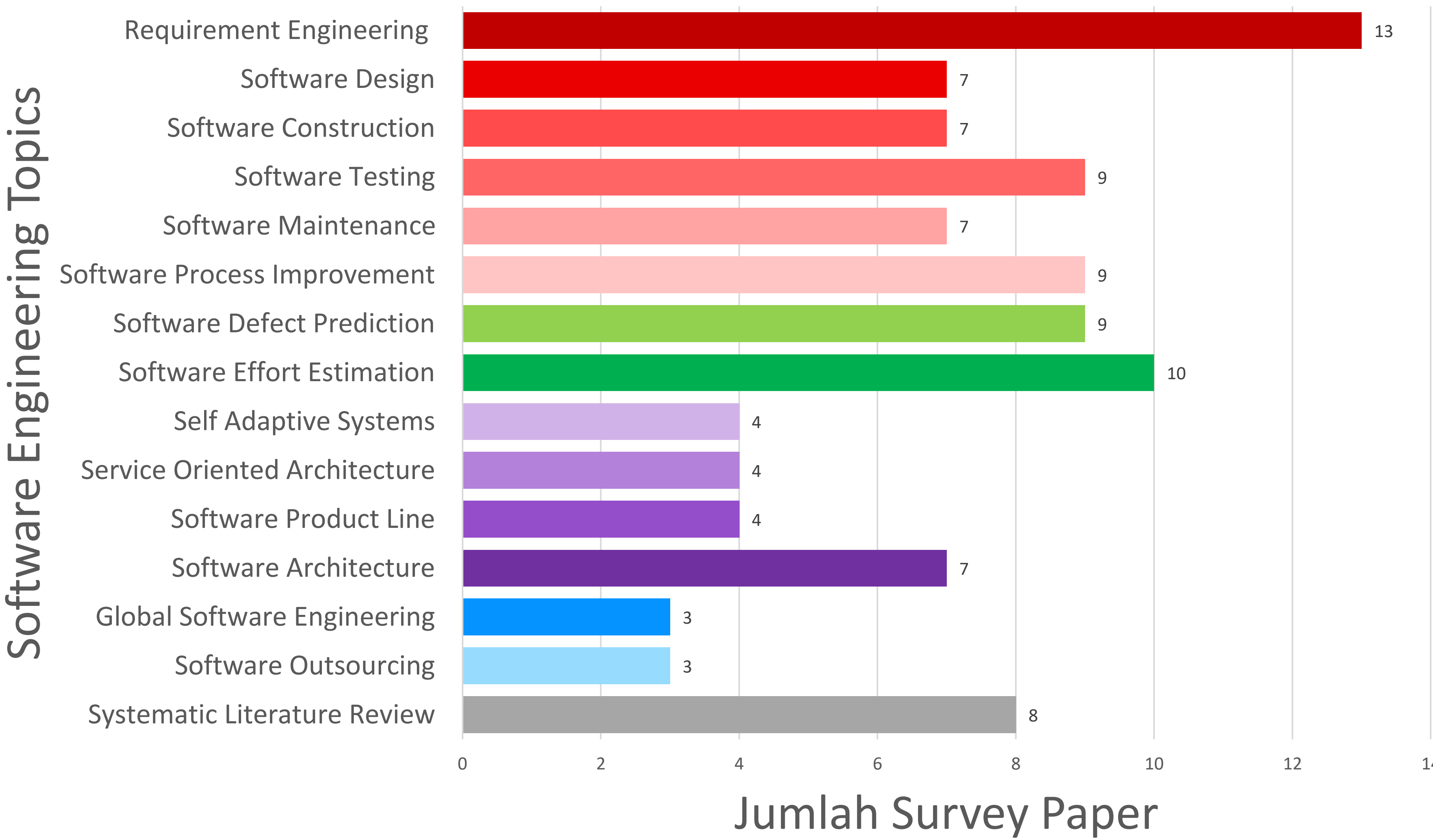
1. Introduction

Global Software Engineering (GSE) has become a growing area of research, apart from being an expanding trend in the Information Technology (IT) industry [3]. GSE requires software tools (management tools, development tools, etc.) to support the special characteristics that this environment has, and which have principally come about as a result of the distance factor (temporal, geographic and socio-cultural distance) [4].

Modern software development, such as globally dispersed teams, creates specific challenges and risks (in spite of the benefits that can be obtained) for the software industry, which need to be considered [5]. In fact, developing software systems through collaboration with other partners and in different geographical locations is a great challenge for organizations [6,7].

Software tools for GSE should therefore help to alleviate problems such as: (a) *Geographic Dispersion*, which sometimes causes a loss of synchronous communication or team interactions, since the sites are in different time zones; (b) *Control and Coordination Breakdown*, owing to the difficulties created by a distributed environment; (c) *Loss of Communication*; this is the case in this type of environment, if we consider that the richest communication medium is face-to-face communication; (d) *Loss of Team Spirit* and trust among team members [8] and (e) *Cultural Differences* which occur when people from different cultures work together in a global environment [9].

Software Engineering Research Trends



Resources: Survey Papers from ScienceDirect, SpringerLink, and IEEE Explore (2011-2020)

Research Topics	Description
Global Software Engineering	Metode dan teknik pengembangan dan pelayanan software dengan environment dan sumber daya tersebar di berbagai negara
Requirement Engineering	Metode dan teknik pengumpulan kebutuhan dalam proses pengembangan software
Self Adaptive Systems	Software yang berkarakter autonomous dan bisa memperbaiki diri sendiri
Software Architecture	Metode dan teknik pengembangan arsitektur software untuk mengurangi kompleksitas: arsitektur model-view-controller, enterprise architecture, etc
Service Oriented Architecture	Metode dan teknik pengembangan dan pelayanan software sebagai sebuah service (software as a services (SaaS) serta proses deliverynya ke pengguna
Software Construction	Metode dan teknik konstuksi software, termasuk: programming paradigm, code programming, refactoring, clone detection, code convention, etc
Software Cost Estimation	Estimasi effort atau cost (berapa orang dan bulan) dari pengembangan software, termasuk: function points, use case points, atau dengan metode machine learning
Software Defect Prediction	Prediksi bug software dengan menggunakan pendekatan machine learning
Software Design	Metode dan teknik perancangan software, termasuk: design pattern, modelling language, forward and reverse engineering, model driven development, etc
Software Maintenance	Metode dan teknik perawatan software setelah software dikembangkan
Software Outsourcing	Metode dan teknik outsourcing dan offshoring pengembangan serta pelayanan software, termasuk: strategi dan parameter dalam pemilihan vendor, etc
Software Process Improvement	Perbaikan proses, siklus, metodologi, dan pengukuran maturity model dari proses pengembangan software
Software Product Line	Metode dan teknik pengembangan dan pengklasifikasian software produk yang memiliki kesamaan karakter dan tujuannya
Software Testing	Metode dan teknik pengujian software untuk berbagai jenis pengujian dan platform
Systematic Literature Review	Penelitian survey yang membahas satu topik penelitian bidang software engineering

Kiat Memilih Topik Penelitian

- Pilih topik bukan karena pekerjaan kita sekarang, tapi karena topiknya menarik (ada passion) dan secara penelitian dapat kita lakukan (tidak mission impossible)
- Usahakan cari penelitian yang membuat kita bisa konsentrasi penuh ke method improvement, tidak harus pontang-panting menjelaskan tentang obyek organisasi, mencari dataset, dsb
- Pilih topik yang dataset sudah tersedia secara public, jadi tidak perlu kita repot mencari dataset untuk eksperimen kita
- Pilih topik yang mudah secara pengukuran penelitian dan bila memungkinkan pengukuran cukup dengan komputer
  - Penelitian requirement engineering, termasuk yang rumit pengukuran penelitiannya, melibatkan manusia dan organisasi sebagai obyek
- Pilih topik sesuai kapasitas dan kapabilitas
  - Kita tidak mungkin penelitian tentang software process improvement apabila tidak tersedia organisasi sebagai testbed yang menerapkan metodologi yang kita kembangkan
- Pilih topik yang memungkinkan kita lakukan dengan laptop kita yang kita miliki sekarang, kecuali kita mendapatkan grant research besar yang memungkinkan pembelian infrastruktur penelitian
  - Penelitian global software engineering, software outsourcing, product line, relative agak perlu biaya lebih besar dan kompleks

Kumpulan Survey Paper di Berbagai Bidang Penelitian Computing

1. Baca artikel tentang tahapan memulai penelitian untuk mahasiswa galau di romisatriawahono.net
2. Tentukan topic penelitian yang kira-kira kita inginkan lewat paper-paper di: http://romisatriawahono.net/lecture/rm/survey/
3. Baca paper survey dan rangkumkan dalam bentuk slide

3. Penentuan Masalah Penelitian

- Searching di google, google scholar, ScienceDirect.Com:
  - Survey review on NAMA TOPIK
  - Research problem challenge on NAMA TOPIK
- Dari “survey paper” yang ditemukan, kejar sampai dapat semua “technical paper” yang ada di daftar referensinya
- Dari puluhan/ratusan/ribuan paper yang didapat lakukan scanning, pilih paper journal yang terindeks SCOPUS/ISI, 3 tahun terakhir, dan peta kan masalah penelitian yang ada di paper-paper itu
- Gunakan Mendeley untuk mempermudah pekerjaan kita
- Pilih satu atau dua masalah penelitian yang kita anggap menarik dan menantang, dan jadikan itu masalah penelitian kita

Susun Research Problem dan Landasan

Masalah Penelitian	**Landasan Literatur**
Data set pada prediksi cacat software berdimensi tinggi, memiliki atribut yang bersifat noisy, dan classnya bersifat tidak seimbang, menyebabkan penurunan akurasi pada prediksi cacat software	There are noisy data points in the software defect data sets that can not be confidently assumed to be erroneous using such simple method *(Gray, Bowes, Davey, & Christianson, 2011)*
	The performances of software defect prediction improved when irrelevant and redundant attributes are removed *(Wang, Khoshgoftaar, & Napolitano, 2010)*
	The software defect prediction performance decreases significantly because the dataset contains noisy attributes *(Kim, Zhang, Wu, & Gong, 2011)*
	Software defect datasets have an imbalanced nature with very few defective modules compared to defect-free ones *(Tosun, Bener, Turhan, & Menzies, 2010)*
	Imbalance can lead to a model that is not practical in software defect prediction, because most instances will be predicted as non-defect prone *(Khoshgoftaar, Van Hulse, & Napolitano, 2011)*
	Software fault prediction data sets are often highly imbalanced *(Zhang & Zhang, 2007)*

4. Perangkuman Metode Yang Ada

- Pahami semua paper penelitian yang tujuannya memecahkan masalah yang sama dengan yang kita pilih
- Pahami metode/algoritma terkini yang mereka gunakan untuk memecahkan masalah penelitian mereka. Ini yang disebut dengan state-of-the-art method
- Dalam bidang computing, metode biasanya berupa algoritma yang secara sistematis, logis dan matematis menyelesaikan masalah

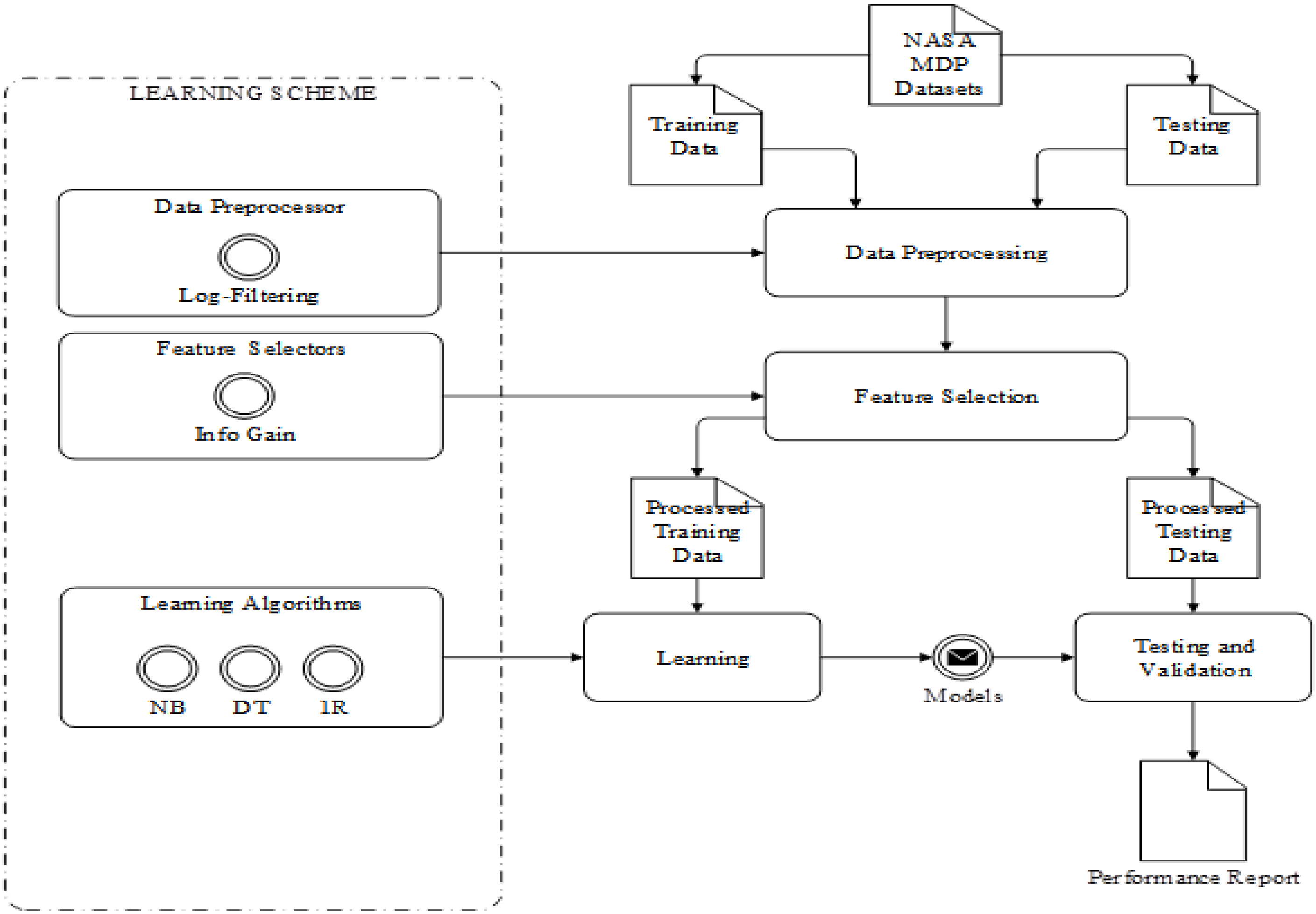
The State-of-the-Art Method

- The highest level of development, as of a device, technique, or scientific field, achieved at a particular time
- The level of development (as of a device, procedure, process, technique, or science) reached at any particular time usually as a result of modern methods (*Merriam Webster Dictionary*)
  - This machine is an example of state-of-the-art technology
  - The state of the art in this field is mostly related to the ABC technology
- A concept used in the process of assessing and asserting novelty and inventive step (*European Patent Convention (EPC)*)

State-of-the-Art Frameworks in Software Defect Prediction

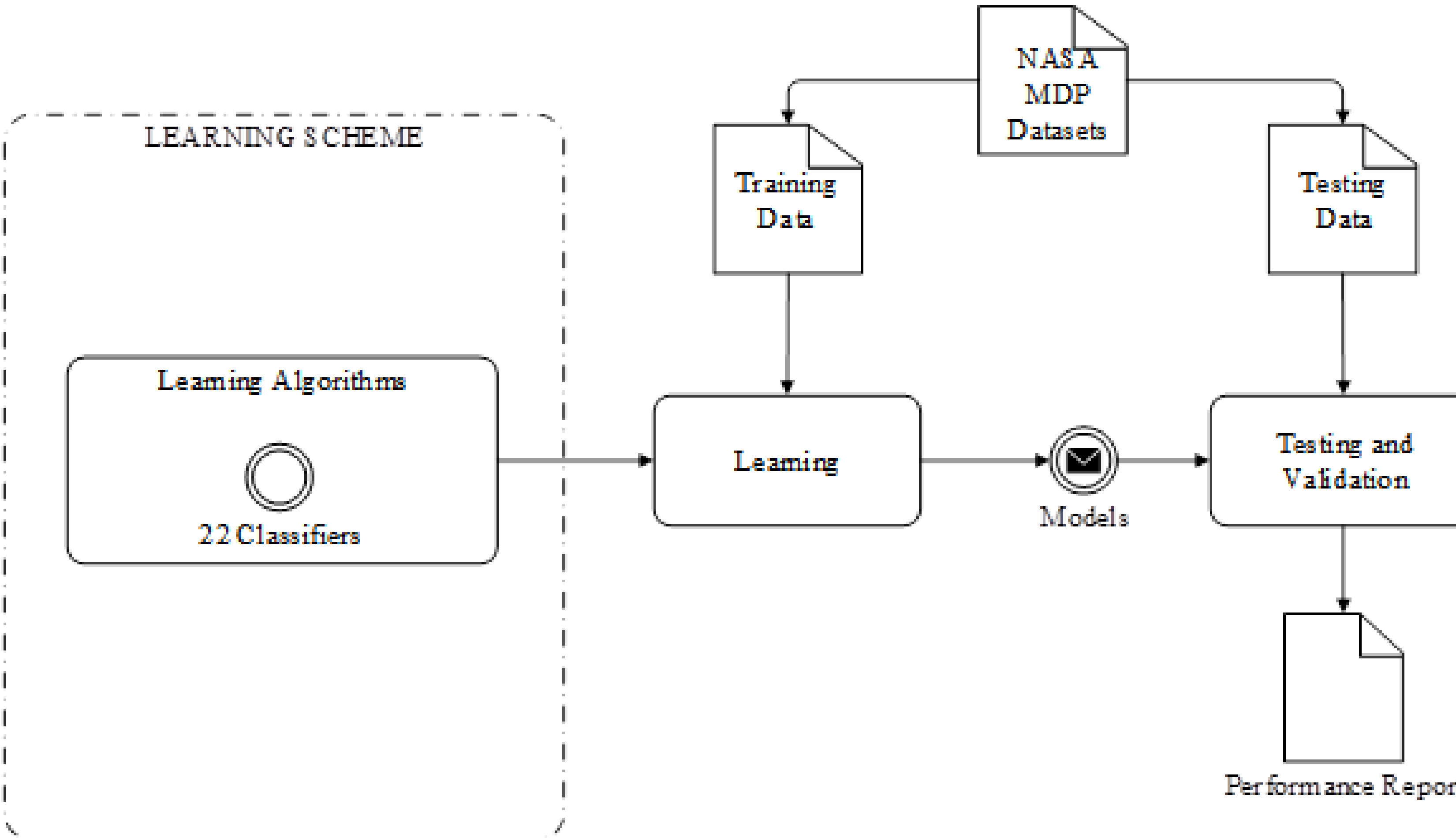
Menzies Framework	**Lessmann** Framework	**Song** Framework
(Menzies et al. 2007)	(Lessmann et al. 2008)	(Song et al. 2011)

Menzies Framework (Menzies et al. 2007)



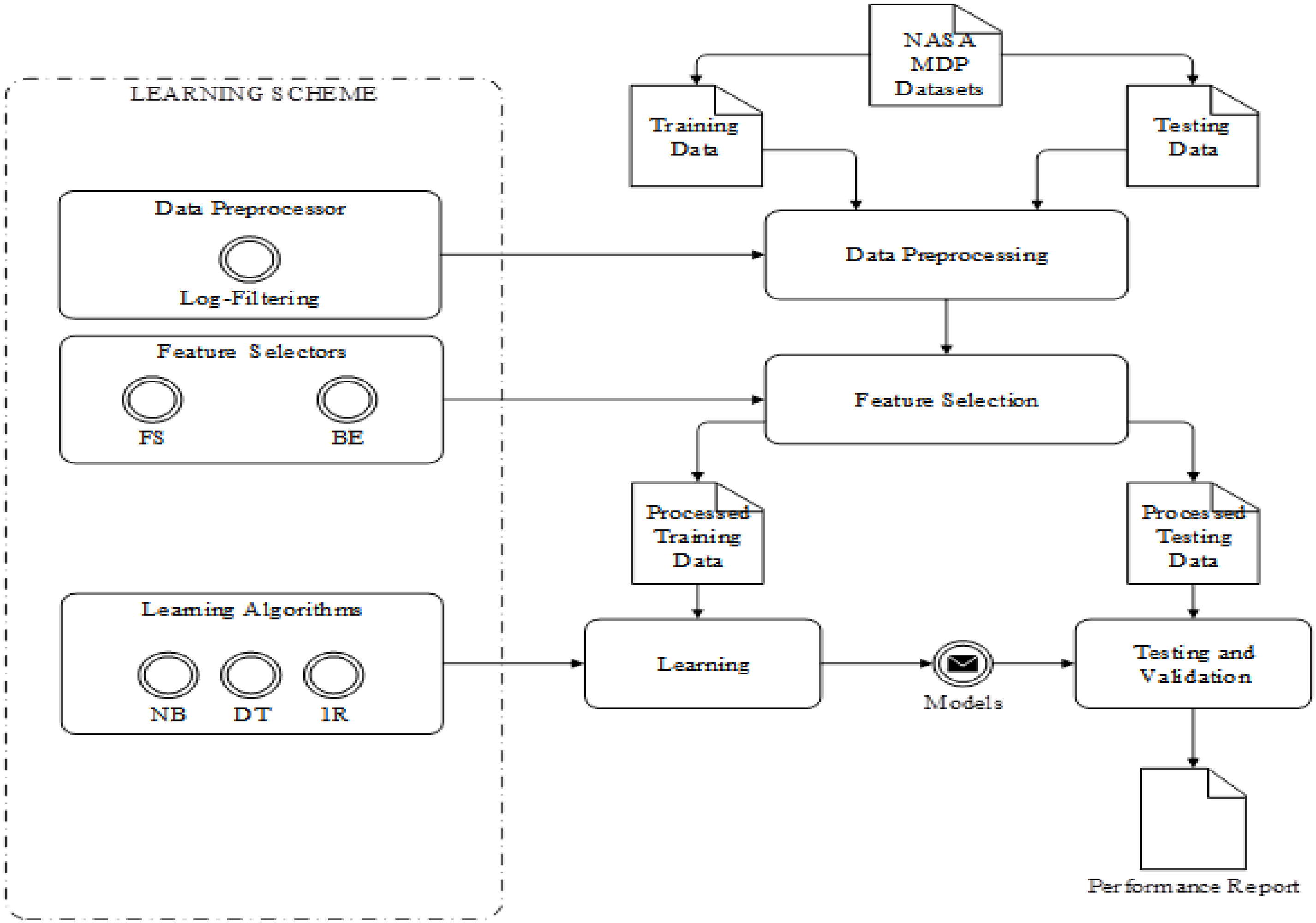
Framework	Dataset	Data Preprocessor	Feature Selectors	Meta-learning	Classifiers	Parameter Selectors	Validation Methods	Evaluation Methods
(Menzies et al. 2007)	NASA MDP	**Log Filtering**	**Info Gain**	-	**3 algorithms (DT, 1R, NB)**	-	10-Fold X Validation	ROC Curve (AUC)

Lessmann Framework (Lessmann et al. 2008)



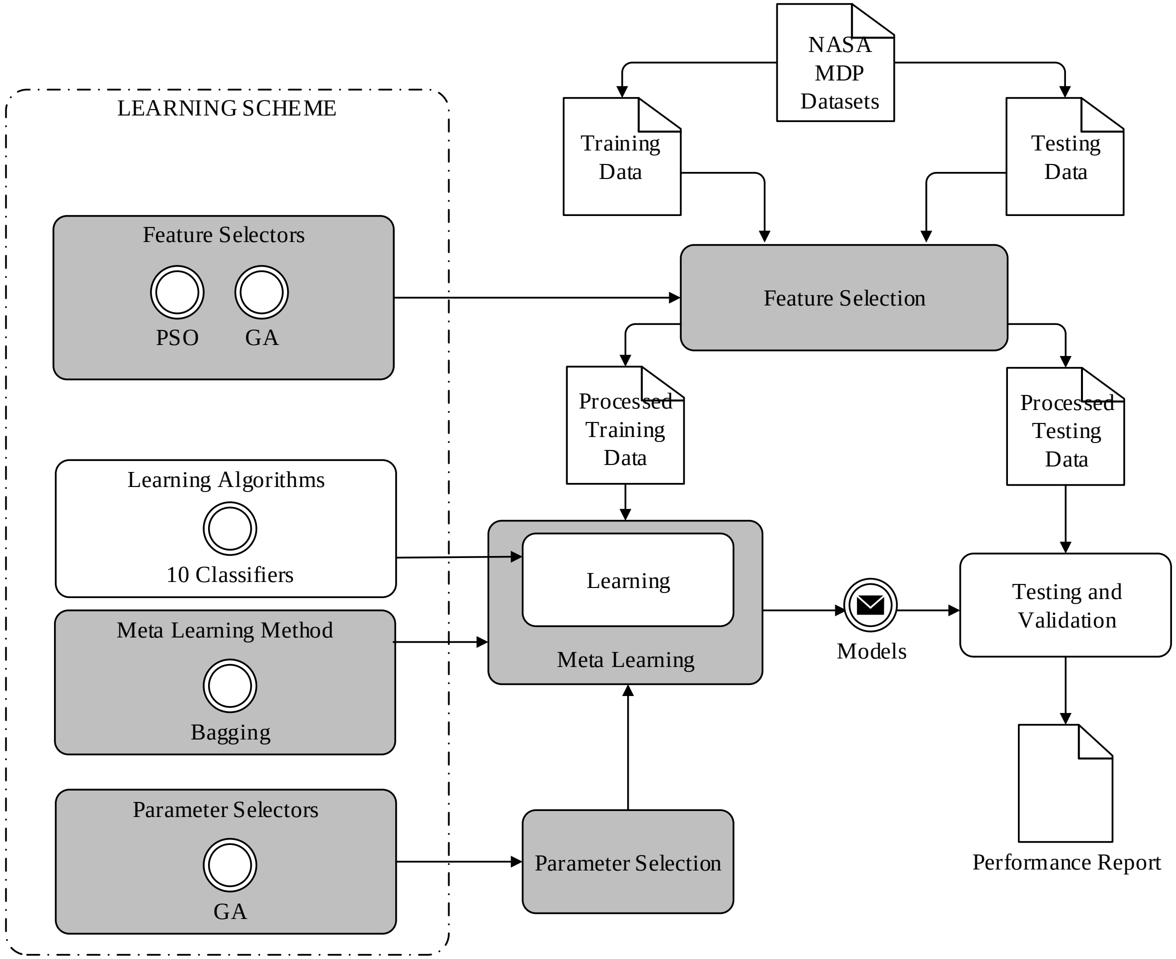
Framework	Dataset	Data Preprocessor	Feature Selectors	Meta-learning	Classifiers	Parameter Selectors	Validation Methods	Evaluation Methods
(Lessman et al. 2008)	NASA MDP	-	-	-	**22 algorithms**	-	10-Fold X Validation	ROC Curve (AUC)

Song Framework (Song et al. 2011)



Framework	Dataset	Data Preprocessor	Feature Selectors	Meta-learning	Classifiers	Parameter Selectors	Validation Methods	Evaluation Methods
(Song et al. 2011)	NASA MDP	**Log Filtering**	**FS, BE**	**-**	**3 algorithms (DT, 1R, NB)**	**-**	10-Fold X Validation	ROC Curve (AUC)

Proposed Framework



Framework	Dataset	Data Preprocessor	Feature Selectors	Meta-Learning	Classifiers	Parameter Selectors	Validation Methods	Evaluation Methods
(Menzies et al. 2007)	NASA MDP	Log Filtering	Info Gain		3 algorithm (DT, 1R, NB)	-	10-Fold X Validation	ROC Curve (AUC)
(Lessman et al. 2008)	NASA MDP	-	-		22 algorithm	-	10-Fold X Validation	ROC Curve (AUC)
(Song et al. 2011)	NASA MDP	Log Filtering	FS, BE		3 algorithm (DT, 1R, NB)	-	10-Fold X Validation	ROC Curve (AUC)
Proposed Framework	NASA MDP	-	**PSO, GA**	**Bagging**	**10 algorithms**	**GA**	10-Fold X Validation	ROC Curve (AUC)

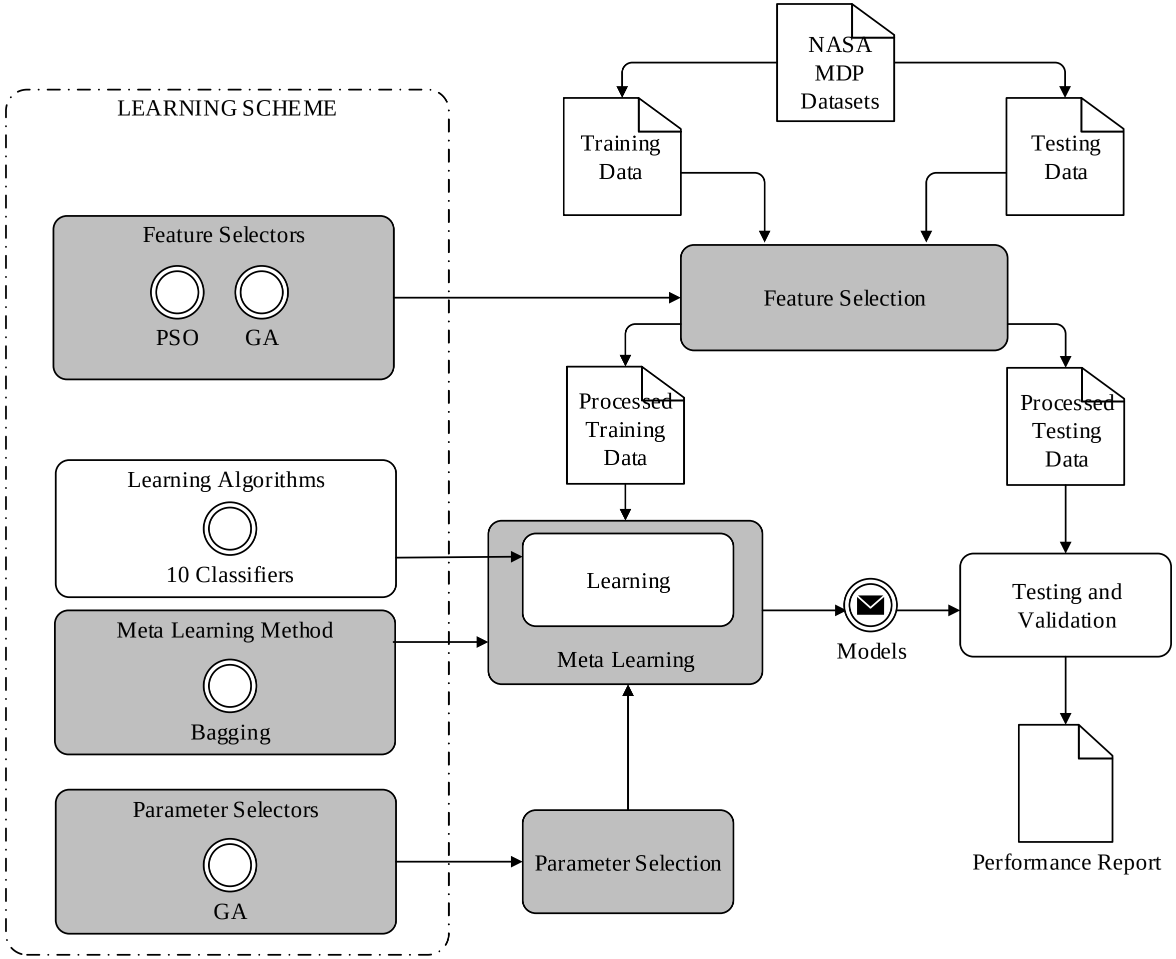
5. Penentuan Metode Yang Diusulkan

- Kita harus membangun dan mengusulkan suatu metode (*proposed method*), yg lebih baik bila dibandingkan dengan metode-metode yang ada saat ini
- Keunggulan metode yang kita usulkan harus dilandasi (*reference*), dibuktikan secara matematis dan empiris lewat hasil eksperimen dan perbandingan dengan metode yang ada
- Metode yang kita usulkan itu bisa saja dari *state-of-the-art methods*, kita kemudian "menambahkan" sesuatu (algoritma, koefisien, formula, dsb), yang akhirnya ketika kita bandingkan dengan metode original, metode kita lebih baik (lebih cepat, lebih akurat, lebih konsisten, dsb).
- "Penambahan" yang kita lakukan dan akhirnya membuat pemecahan masalah menjadi lebih baik itulah yang disebut dengan kontribusi ke pengetahuan (*contribution to knowledge*) (Dawson, 2009)

Susun RP-RQ-RO

Research Problem (RP)	Research Question (RQ)	Research Objective (RO)
RP1. Data set pada prediksi cacat software berdimensi tinggi, memiliki atribut yang bersifat noisy, dan classnya bersifat tidak seimbang, menyebabkan penurunan akurasi pada prediksi cacat software	RQ1. Algoritma pemilihan fitur apa yang performanya terbaik untuk meyelesaikan masalah atribut yang noisy pada prediksi cacat software?	RO1. Mengidentifikasi algoritma pemilihan fitur apa yang memiliki performa terbaik apabila digunakan untuk menyelesaikan masalah atribut yang noisy pada prediksi cacat software
	RQ2. Algoritma meta learning apa yang performanya terbaik untuk menyelesaikan masalah class imbalance pada prediksi cacat software?	RO2. Mengidentifikasi algoritma meta learning apa yang memiliki performa terbaik apabila digunakan untuk menyelesaikan masalah class imbalance pada prediksi cacat software
	RQ3. Bagaimana pengaruh penggabungan algoritma pemilihan fitur dan metode meta learning terbaik pada peningkatan akurasi prediksi cacat software?	RO3. Mengembangkan algoritma baru yang menggabungkan algoritma pemilihan fitur dan meta learning terbaik untuk meningkatkan akurasi pada prediksi cacat software

Proposed Framework



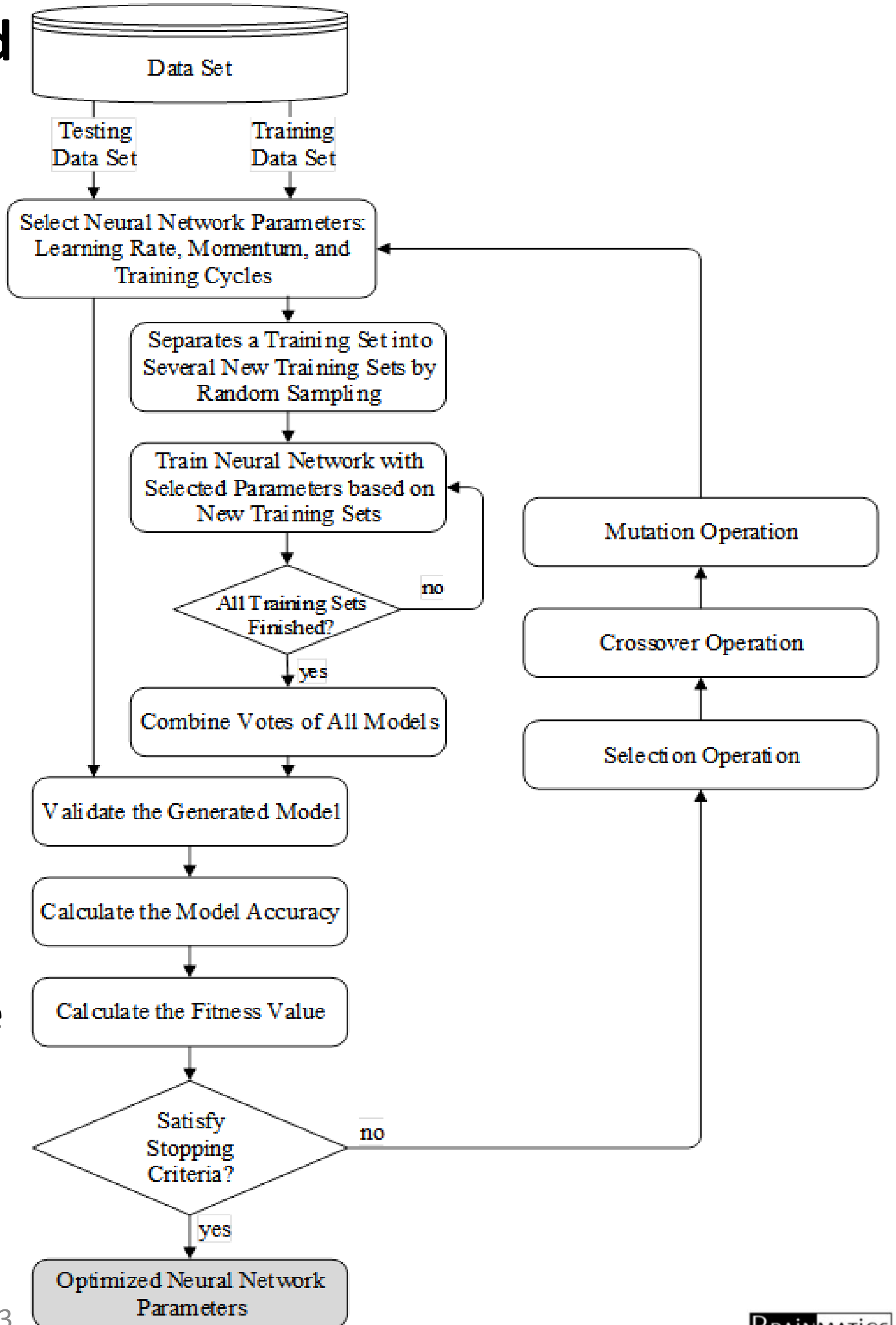
Framework	Dataset	Data Preprocessor	Feature Selectors	Meta-Learning	Classifiers	Parameter Selectors	Validation Methods	Evaluation Methods
(Menzies et al. 2007)	NASA MDP	Log Filtering	Info Gain		3 algorithm (DT, 1R, NB)	-	10-Fold X Validation	ROC Curve (AUC)
(Lessman et al. 2008)	NASA MDP	-	-		22 algorithm	-	10-Fold X Validation	ROC Curve (AUC)
(Song et al. 2011)	NASA MDP	Log Filtering	FS, BE		3 algorithm (DT, 1R, NB)	-	10-Fold X Validation	ROC Curve (AUC)
Proposed Framework	NASA MDP	-	**PSO, GA**	**Bagging**	**10 algorithms**	**GA**	10-Fold X Validation	ROC Curve (AUC)

A Hybrid Genetic Algorithm based Neural Network Parameter Optimization and Bagging Technique for Software Defect Prediction (NN GAPO+B)

- Every chromosome is evaluated by the fitness function Equation

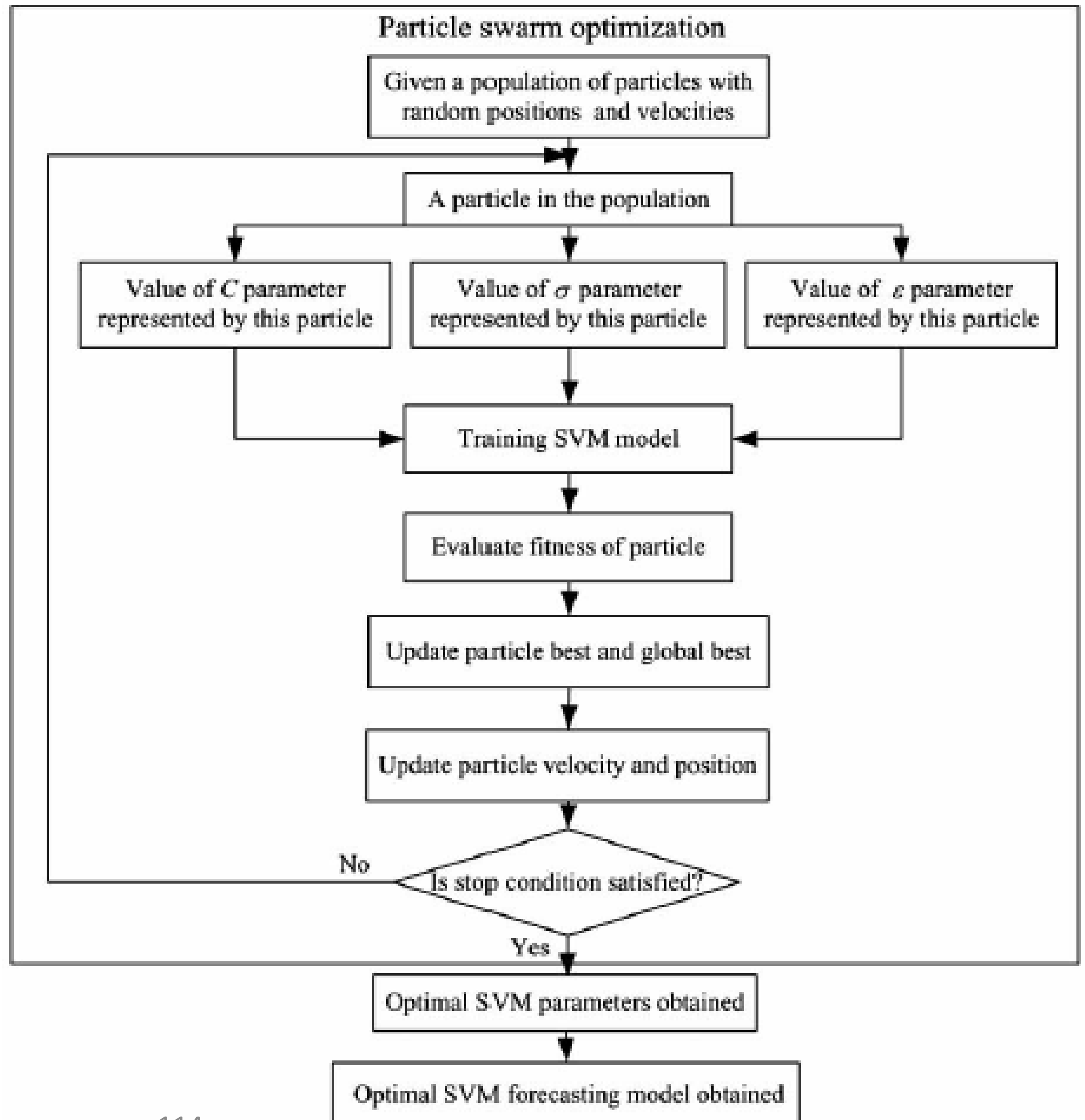
$$fitness = W_A \times A + W_P \times \left(S + \left(\sum_{i=1}^{n} C_i \times P_i \right) \right)^{-1}$$

- Where
  - A: classification accuracy
  - P_i: parameter value
  - W_A: weight of classification accuracy
  - W_p: parameter weight
  - C_i: feature cost
  - S: setting constant
- When ending condition is satisfied, the operation ends and the optimized NN parameters are produced. Otherwise, the process will continue with the next generation operation



Contoh Proposed Method

Metode yang diusulkan adalah metode SVM dengan pemilihan parameter C, Gamma dan Epsilon diotomatisasi menggunakan PSO



6. Evaluasi Metode Yang Diusulkan

- Metode yang diusulkan harus divalidasi dan dievaluasi dengan metode pengukuran standard dan disepakati para peneliti di bidang penelitian yang kita lakukan
- Pengukuran metode disesuaikan dengan masalah dan tujuan penelitian:
  - Masalahnya rendahnya akurasi → pengukurannya akurasi
  - Masalah rendahnya efisiensi → pengukurannya waktu

Evaluasi pada Penelitian Data Mining

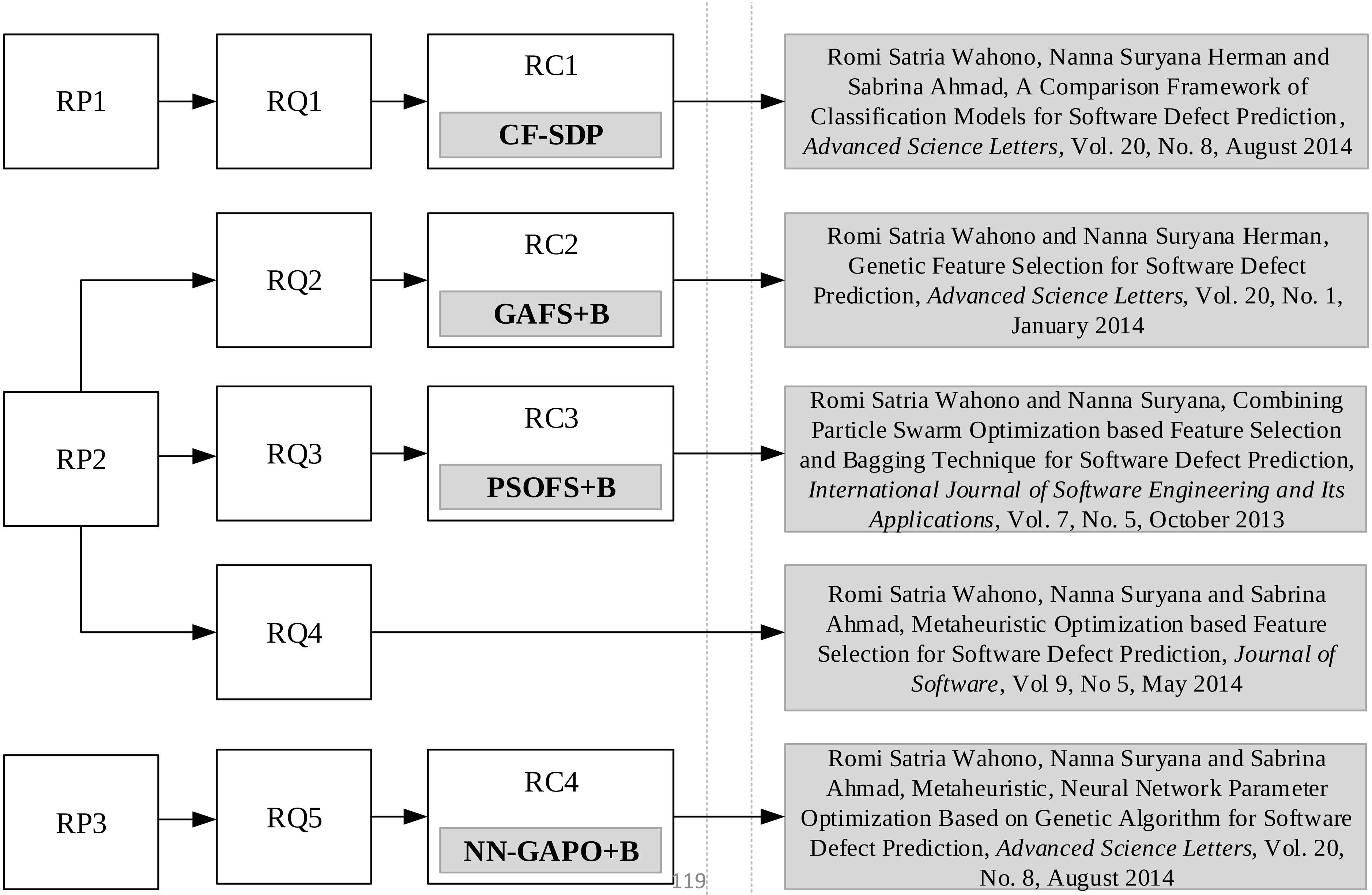
1. Estimation:
   - Error: Root Mean Square Error (RMSE), MSE, MAPE, etc
2. Prediction/Forecasting (Prediksi/Peramalan):
   - Error: Root Mean Square Error (RMSE) , MSE, MAPE, etc
3. Classification:
   - Confusion Matrix: Accuracy
   - ROC Curve: Area Under Curve (AUC)
4. Clustering:
   - Internal Evaluation: Davies–Bouldin index, Dunn index,
   - External Evaluation: Rand measure, F-measure, Jaccard index, Fowlkes–Mallows index, Confusion matrix
5. Association:
   - Lift Charts: Lift Ratio
   - Precision and Recall (F-measure)

7. Penulisan Ilmiah dan Publikasi Hasil Penelitian

- Lakukan pendataan journal-journal yang ada di bidang kita, urutkan berdasarkan rangking SJR atau JIF
- Pilih target journal untuk tempat publikasi hasil penelitian kita
- Publikasikan hasil penelitian ke journal yang sesuai dengan kualitas kontribusi penelitian yang kita lakukan
- A paper is an organized description of hypotheses, data and conclusions, intended to instruct the reader. If your research does not generate papers, it might just as well not have been done (Whitesides 2004)

No	Journal Publications	SJR	Q Category
1	IEEE Transactions on Software Engineering	3.39	Q1 in Software
2	Information Sciences	2.96	Q1 in Information Systems
3	IEEE Transactions on Systems, Man, and Cybernetics	2.76	Q1 in Artificial Intelligence
4	IEEE Transactions on Knowledge and Data Engineering	2.68	Q1 in Information Systems
5	Empirical Software Engineering	2.32	Q1 in Software
6	Information and Software Technology	1.95	Q1 in Information Systems
7	Automated Software Engineering	1.78	Q1 in Software
8	IEEE Transactions on Reliability	1.43	Q1 in Software
9	Expert Systems with Applications	1.36	Q2 in Computer Science
10	Journal of Systems and Software	1.09	Q2 in Software
11	Software Quality Journal	0.83	Q2 in Software
12	IET Software	0.55	Q2 in Software
13	Advanced Science Letters	0.24	Q3 in Computer Science
14	Journal of Software	0.23	Q3 in Software
15	International Journal of Software Engineering and Its Application	0.14	Q4 in Software

RP – RQ – RC dan Publikasi Penelitian



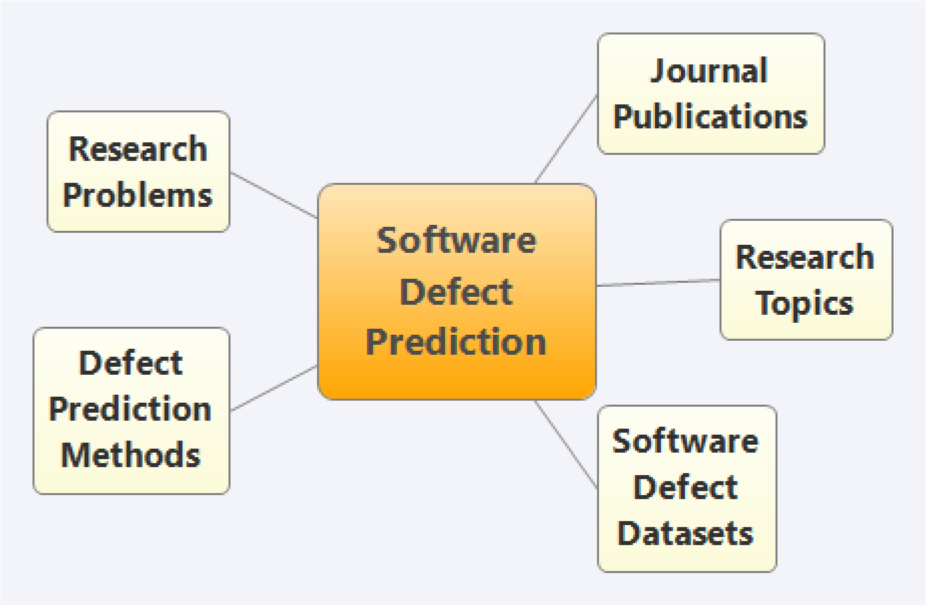
Tugas Menentukan Bidang Penelitian

1. Analisis kembali berbagai mata kuliah yang telah diajarkan, serta paper dan buku yang telah dibaca
2. Tentukan bidang dan sub bidang (field dan subfield) penelitian yang kita tertarik untuk melakukannya
3. Baca artikel tentang tahapan memulai penelitian utk mahasiswa galau di *http://romisatriawahono.net*

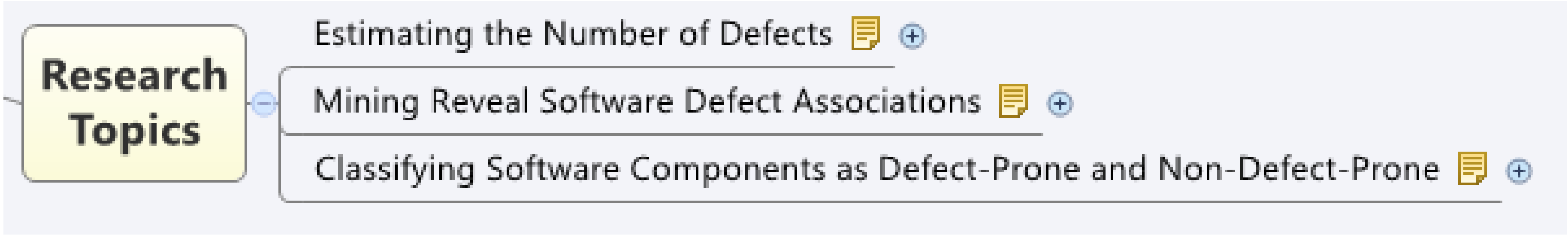
Tugas Menentukan Topik Penelitian

1. Tentukan topik penelitian dari bidang penelitian
2. Studi literatur tentang topik tersebut (state-of-the-art, research direction/challenge) dari paper journal (review paper)
3. Rangkumkan topic penelitian yang kita pilih secara mendetail dalam bentuk mindmap dengan software XMind
4. Ikuti artikel di
   http://romisatriawahono.net/2014/02/28/mind-map-untuk-memahami-topik-penelitian/

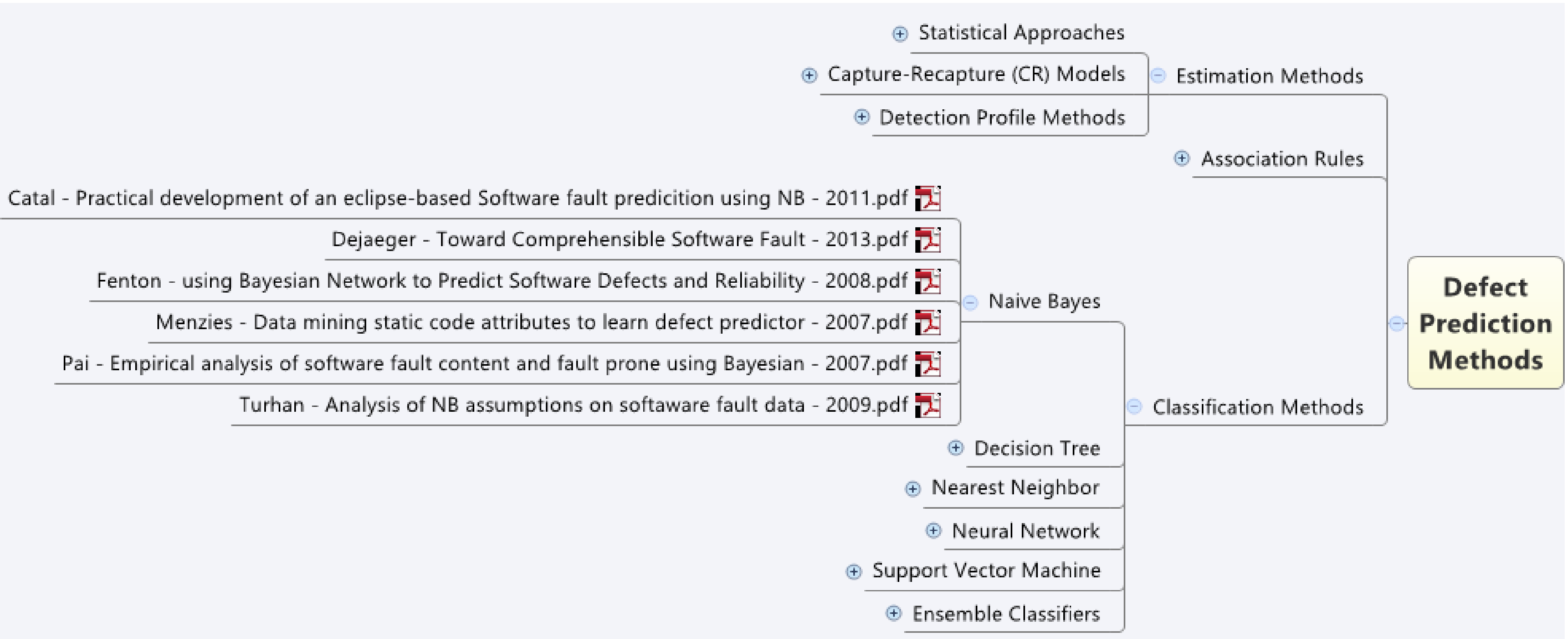
Contoh Mindmap untuk Topik "Software Defect Prediction"



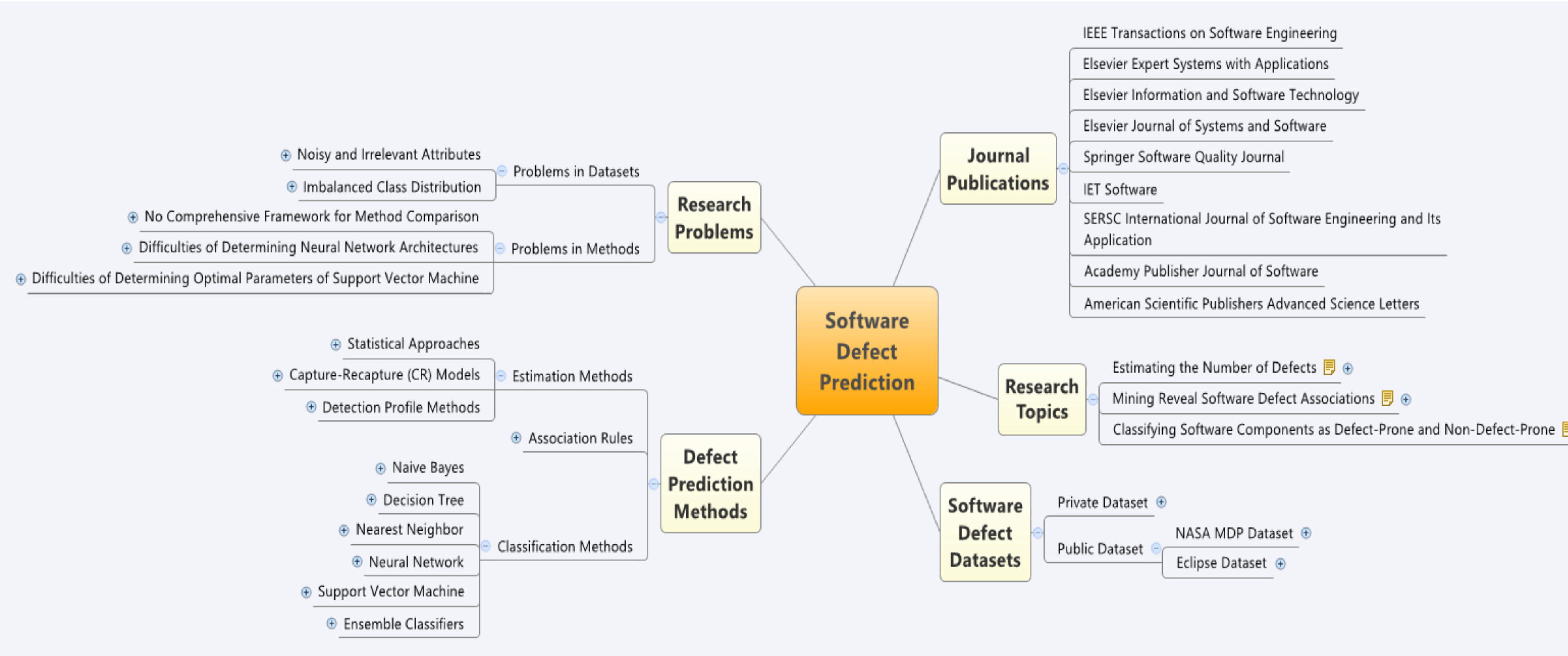
Contoh Mindmap untuk Topik "Software Defect Prediction"



Contoh Mindmap untuk Topik "Software Defect Prediction"



Contoh Mindmap untuk Topik "Software Defect Prediction"



Tugas Menentukan Masalah Penelitian

1. Dari topik penelitian yang dipilih
2. Tentukan beberapa masalah penelitian yang muncul di survey paper atau technical paper

Tugas Literature Review

- Tentukan topik penelitian yang diinginkan
- Baca beberapa survey paper yang ada di folder dan cari paper survey lain yang relevan dan lengkap tentang topik tersebut (scopus indexed journal)
- Rangkumkan ke dalam bentuk slide, dengan menggunakan bahasa manusia yang baik dan benar:
    1. Mengapa topik penelitian itu penting, susun argumentasi, landasan dan analisis ekonomi berhubungan dengan topik penelitian tersebut
    2. Sub topik apa saja yang ada pada penelitian tersebut, dan siapa yang mengatakan (landasi) bahwa itu sub topik yang baik
    3. Siapa saja nama peneliti yang banyak menulis paper di topik penelitian itu
    4. Metode apa saja yang telah diusulkan berhubungan dengan topik tersebut
    5. Dataset apa saja yang biasa digunakan pada topik penelitian tersebut
    6. Masalah penelitian apa saja yang biasanya diangkat oleh para peneliti di topik tersebut

2.3 Tahapan Penelitian Computing Fokus Perbaikan Algoritma

Tahapan Penelitian Computing Fokus Perbaikan Algoritma

Literature Review

1. Pemilihan Satu Algoritma yang Menarik (***Research Field***)
2. Pencarian Paper Journal yang Melakukan Perbaikan Algoritma Tersebut (***Research Topic***)
3. Penentuan Masalah Penelitian (***Research Problem***)
4. Perangkuman Metode-Metode Yang Ada (***State-of-the-Art Methods***)
5. Penentuan Metode Yang Diusulkan (***Proposed Method***)
6. Evaluasi Metode Yang Diusulkan (***Evaluation***)
7. Penulisan Ilmiah dan Publikasi Hasil Penelitian (***Publications***)

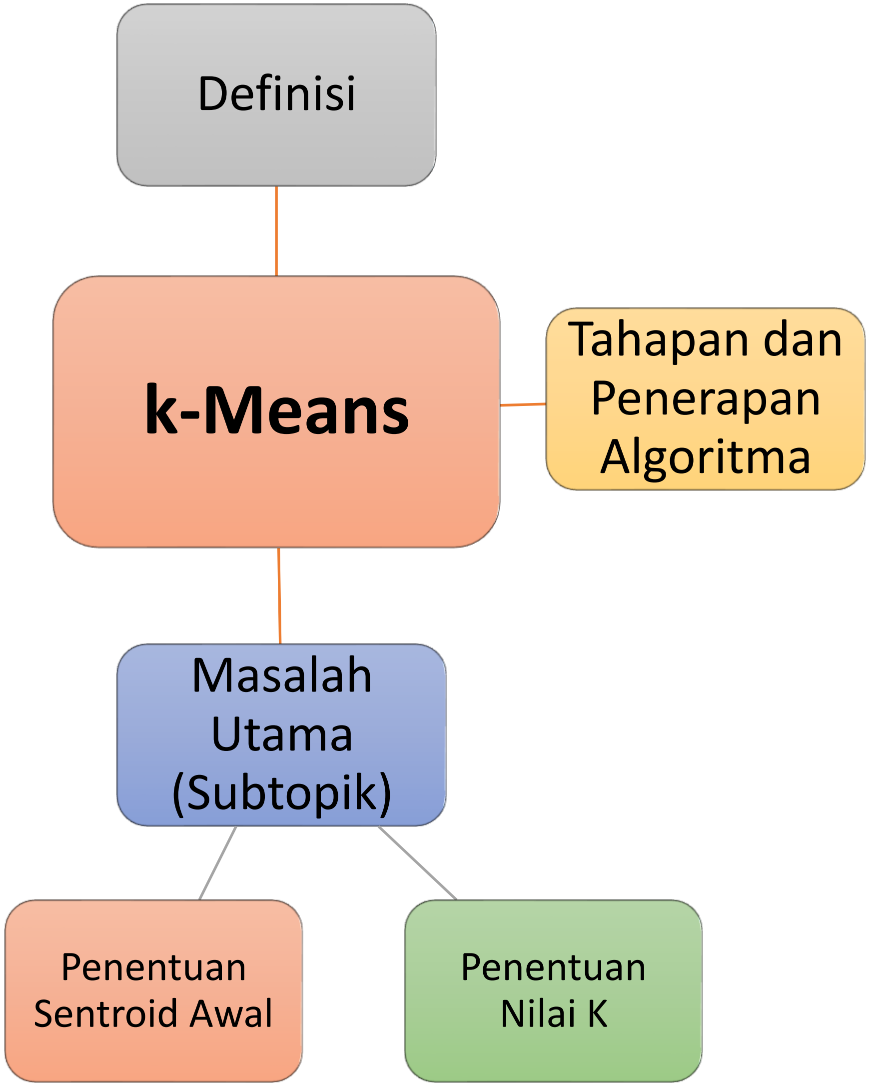
**http://romisatriawahono.net/2013/01/23/tahapan-memulai-penelitian-untuk-mahasiswa-galau/*

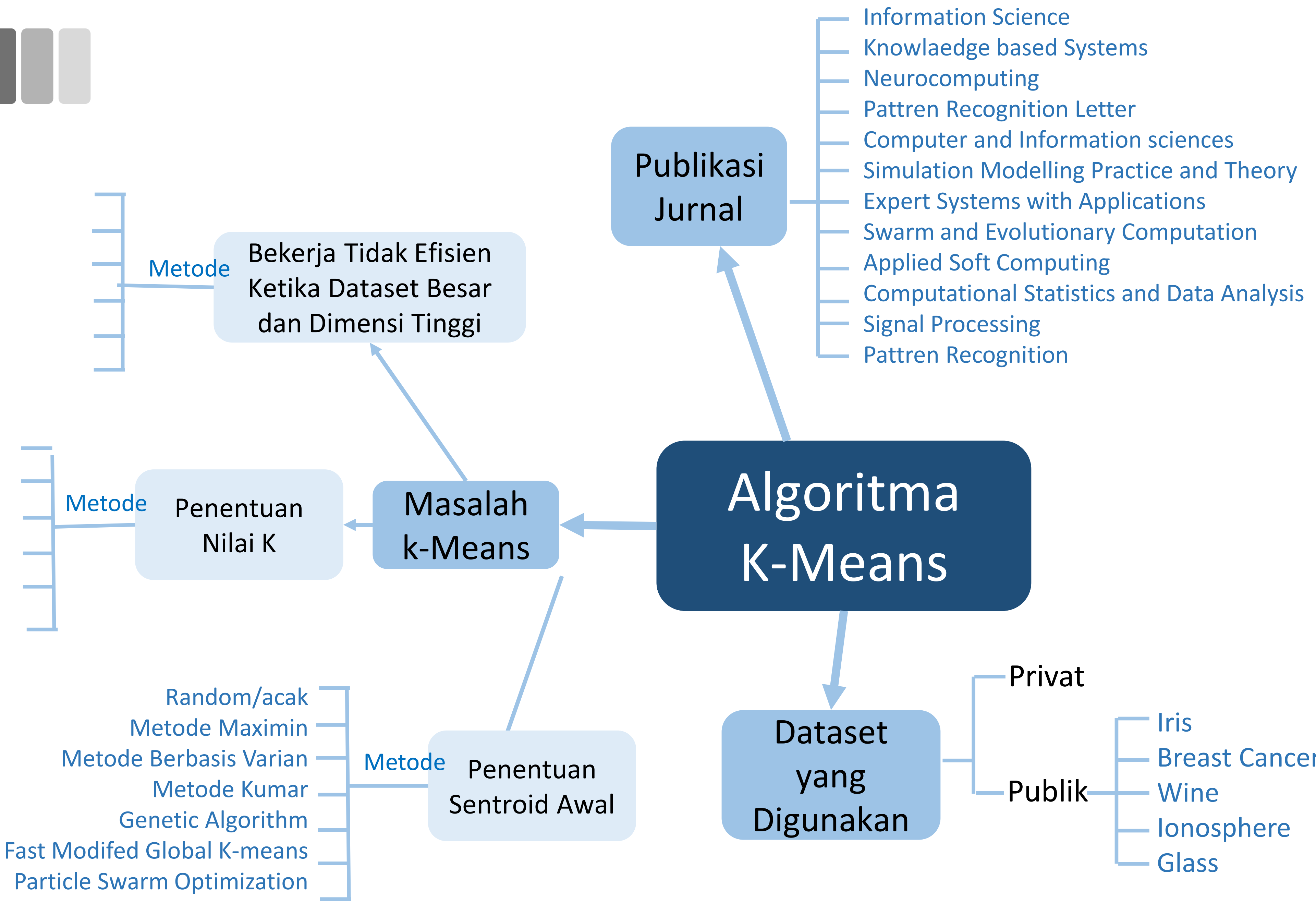
Tugas

1. Tentukan algoritma yang ingin kita perbaiki pada penelitian
2. Lakukan pencarian dan kumpulkan paper yang membahas algoritma yang kita pilih (boleh technical atau survey paper)
3. Konsentrasi ke journal Q2 dan Q1, cek dengan ScimagoJr.Com
4. Baca abstract, introduction dan related research dari paper tersebut
5. Pahami masalah dan kontribusi penelitian dari paper-paper yang didownload dan susun dalam bentuk tabel

k-Means

- Definisi k-Means (Textbooks)
- Tahapan dan Penghitungan k-Means (Textbooks)
- Masalah Algoritma k-Means
  - Penentuan Nilai K
    - Metode Joko (Joko, 2017)
    - Metode Budi (Budi, 2016)
    - Metode Wati (Wati, 2018)
    - ....
  - Penentuan Sentroid Awal
    - Metode Karno (Karno, 2014)
    - Metode Kartini (Kartini, 2013)
    - Metode Subagyo (Subagyo, 2016)
    - ....





Algoritma K-Means
Publikasi Jurnal
Information Science
Knowlaedge based Systems
Neurocomputing
Pattren Recognition Letter
Computer and Information sciences
Simulation Modelling Practice and Theory
Expert Systems with Applications
Swarm and Evolutionary Computation
Applied Soft Computing
Computational Statistics and Data Analysis
Signal Processing
Pattren Recognition
Masalah k-Means
Bekerja Tidak Efisien Ketika Dataset Besar dan Dimensi Tinggi
Metode
Penentuan Nilai K
Metode
Penentuan Sentroid Awal
Metode
Random/acak
Metode Maximin
Metode Berbasis Varian
Metode Kumar
Genetic Algorithm
Fast Modifed Global K-means
Particle Swarm Optimization
Dataset yang Digunakan
Privat
Publik
Iris
Breast Cancer
Wine
Ionosphere
Glass

3. Literature Review

3.1 Literatur Ilmiah

3.2 Teknik Mengelola Paper

3.3 Teknik Mereview Paper

3.1 Literatur Ilmiah

Tahapan Penelitian Computing

Literature Review

1. Penentuan Bidang Penelitian (***Research Field***)
2. Penentuan Topik Penelitian (***Research Topic***)
3. Penentuan Masalah Penelitian (***Research Problem***)
4. Perangkuman Metode-Metode Yang Ada (***State-of-the-Art Methods***)
5. Penentuan Metode Yang Diusulkan (***Proposed Method***)
6. Evaluasi Metode Yang Diusulkan (***Evaluation***)
7. Penulisan Ilmiah dan Publikasi Hasil Penelitian (***Publications***)

**https://www.site.uottawa.ca/~bochmann/dsrg/how-to-do-good-research/*
**http://romisatriawahono.net/2013/01/23/tahapan-memulai-penelitian-untuk-mahasiswa-galau/*

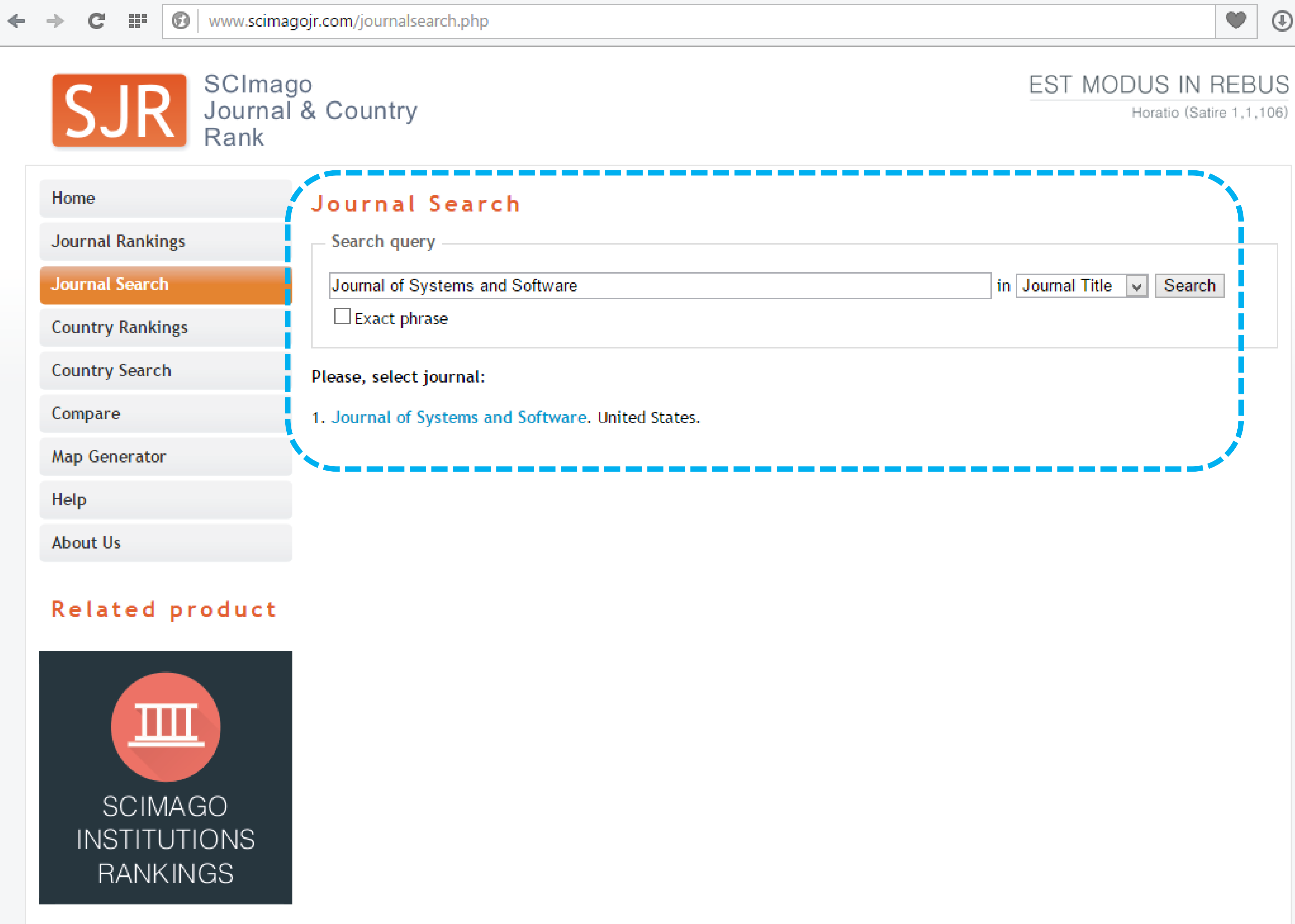
Manfaat Mereview Literatur

- Memperdalam pengetahuan tentang bidang dan topik yang diteliti (*Textbooks*)
- Memperdalam pengetahuan tentang topik lebih detail yang diteliti (*Survey Paper*)
- Mengetahui hasil penelitian yang berhubungan dan yang sudah pernah dilaksanakan (Related Research) (*Technical Paper*)
- Mengetahui perkembangan ilmu pada bidang yang kita pilih (state-of-the-art) (*Technical atau Survey Paper*)
- Memperjelas masalah penelitian (*Technical Paper*)

Jenis Literatur Ilmiah

1. **Paper dari Journal** *
2. Paper dari Book Chapter
3. Paper dari Conference (Proceedings)
4. Thesis dan Disertasi
5. Report (Laporan) dari Organisasi yang Terpercaya
6. Buku Textbook

** Prioritaskan paper journal terindeks WoS (Clarivate) or SCOPUS*
Cek dengan http://scimagojr.com


www.scimagojr.com/journalsearch.php
SJR
SCImago Journal & Country Rank
EST MODUS IN REBUS
Horatio (Satire 1,1,106)
Home
Journal Rankings
Journal Search
Country Rankings
Country Search
Compare
Map Generator
Help
About Us
Related product
SCIMAGO INSTITUTIONS RANKINGS
Journal Search
Search query
Journal of Systems and Software
in
Journal Title
Search
Exact phrase
Please, select journal:
1. Journal of Systems and Software. United States.

www.scimagojr.com/journalsearch.php

SJR SCImago Journal & Country Rank

EST MODUS IN REBUS
Horatio (Satire 1,1,106)

- Home
- Journal Rankings
- Journal Search
- Country Rankings
- Country Search
- Compare
- Map Generator
- Help
- About Us

Journal Search

Search query

in Journal Title Search

Exact phrase

Journal of Systems and Software

Country: United States

Subject Area: Computer Science

Subject Category:

Category	Quartile (Q1 means highest values and Q4 lowest values)														
	1999	2000	2001	2002	2003	2004	2005	2006	2007	2008	2009	2010	2011	2012	2013
Hardware and Architecture	Q2	Q2	Q1	Q2	Q2	Q2	Q1	Q2	Q2	Q2	Q1	Q2	Q1	Q1	Q1
Information Systems	Q3	Q2	Q2	Q2	Q2	Q2	Q2	Q2	Q2	Q2	Q2	Q2	Q2	Q2	Q2
Software	Q3	Q2	Q2	Q2	Q2	Q2	Q2	Q2	Q2	Q2	Q2	Q2	Q2	Q2	Q2

Publisher: Elsevier Inc.. **Publication type:** Journals. **ISSN:** 01641212

Coverage: 1979-2014

H Index: 60

Scope:

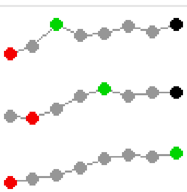
The Journal of Systems and Software publishes papers covering all aspects of programming methodology, software engineering, and related hardware-software-systems issues. [...]

Show full scope

Show this information in your own website

Journal of Systems and Software

Indicator	2006-2013	Value
SJR		0.82
Cites per doc		2.57
Total cites		1645



www.scimagojr.com

Display journal title

ELSEVIER

Type here to search on Elsevier.com

Advanced Product Search

Books & journals | Online tools | Authors, editors & reviewers | About Elsevier | Help | Store

Journal of Systems and Software

The Journal of Systems and Software publishes papers covering all aspects of programming methodology, software engineering, and related hardware-software-systems issues. Topics of interest include, but...

View full aims and scope

Editor-in-Chief, Software Architecture Papers: H. van Vliet
View full editorial board

- Guide for Authors
- Submit Your Paper
- Track Your Paper
- Order Journal
- Access Full Text

Impact Factor:
1.277

5-Year Impact Factor:
1.282

Imprint: ELSEVIER

ISSN: 0164-1212

News

Publish multimedia data in your articles

SciVerse App: Expert Search

VIEW ALL

Guidelines

Charting a Course for a Successful Research Career

Ways to Use Journal Articles Published by Elsevier: A Practical Guide

Call for Papers

Special issue on "Trustworthy Software Systems for the Digital Society"

Special Issue on Variability in Software Architecture

VIEW ALL

Most Downloaded Articles

ScienceDirect

Organisasi Yang Mengindeks Journal

1. Thomson Reuters Web of Science (Clarivate Analytics)
   - Since 1963, formerly produced by ISI, 12032 journals are indexed
   - Pengindeks journal yang memiliki level paling baik
   - *http://wokinfo.com*
2. Scopus
   - Launched by Elsevier in 2004, 20000 journals, conference papers and other are indexed
   - Pengindeks journal level standard, biasa untuk syarat menyelesaikan PhD
   - *http://scopus.com*
3. Google Scholar
   - Launched in 2004, mengindeks semua publikasi ilmiah yang online
   - *http://scholar.google.com*

** Organisasi pengindeks journal selain di atas (EBSCO, DBLP, ProQuest, dsb), boleh dikatakan selevel dengan Google Scholar*

Algoritma Perangkingan Journal

1. Journal Impact Factor (JIF)
   - Data source: ISI Web of Science
2. Eigenfactor Score (ES)
   - Data source: ISI Web of Science
   - *http://www.eigenfactor.org*
3. Scimago Journal Rank (SJR)
   - Data source: Scopus
   - *http://www.scimagojr.com*
4. Source Normalized Impact per Paper (SNIP)
   - Data source: Scopus
   - *http://www.journalindicators.com*
5. h-index
   - Data source: Google Scholar
   - *http://scholar.google.com/intl/en/scholar/metrics.html*

** JIF adalah algoritma yang digunakan oleh ISI, sedangkan SJR adalah algoritma yang digunakan oleh SCOPUS*

Sumber Pencarian Literatur

GRATIS

Journal

- http://scholar.google.com
- http://citeseer.ist.psu.edu

BERBAYAR

Journal

- http://sciencedirect.com
- http://www.ebscohost.com
- http://link.springer.com
- http://ieeexplore.ieee.org
- http://dl.acm.org

scholar.google.com

Google | Software defect prediction

Cendekia — Sekitar 168.000 hasil (**0,08** dtk)

Artikel

Koleksiku

Kapan saja

Sejak 2015

Sejak 2014

Sejak 2011

Rentang khusus...

Urutkan menurut relevansi

Urutkan menurut tanggal

☑ sertakan paten

☑ mencakup kutipan

Buat lansiran

Kiat: Telusuri laman berbahasa **Bahasa Indonesia** saja. Anda dapat menentukan bahasa penelusuran di Setelan Cendekia.

A critique of software defect prediction models — [PDF] dari qmul.ac.uk

NE Fenton, M Neil - **Software** Engineering, IEEE Transactions …, 1999 - ieeexplore.ieee.org

Abstract—Many organizations want to predict the number of defects (faults) in **software** systems, before they are deployed, to gauge the likely delivered quality and maintenance effort. To help in this numerous **software** metrics and statistical models have been **...**

Dirujuk 887 kali Artikel terkait 12 versi Kutip Simpan

Benchmarking classification models for software defect prediction: A proposed framework and novel findings

S Lessmann, B Baesens, C Mues… - **Software** Engineering, …, 2008 - ieeexplore.ieee.org

Abstract—**Software defect prediction** strives to improve **software** quality and testing efficiency by constructing predictive classification models from code attributes to enable a timely identification of fault-prone modules. Several classification models have been evaluated **...**

Dirujuk 352 kali Artikel terkait 12 versi Kutip Simpan

Empirical assessment of machine learning based software defect prediction techniques — [PDF] dari lums.edu.p

VUB Challagulla, FB Bastani, IL Yen… - International Journal on …, 2008 - World Scientific

Automated reliability assessment is essential for systems that entail dynamic adaptation based on runtime mission-specific requirements. One approach along this direction is to monitor and assess the system using machine learning-based **software defect prediction ...**

Dirujuk 122 kali Artikel terkait 10 versi Kutip Simpan

Software defect prediction

BE Hunt Jr, JJ Kirkpatrick, RA Kloss… - US Patent …, 2014 - freepatentsonline.com

Abstract: A method of **software defect prediction** by a computer is provided. The method comprises identifying **software** test organizations scheduled to perform testing on an application or applications, where the scope of the **software** testing varies between **...**

sciencedirect.com

ScienceDirect

Advanced Sea

Search tips ?

ScienceDirect

Journals & Books ?

40 results

Set search alert

Refine by:

Years

- 2020 (1)
- 2019 (5)
- 2018 (2)

Show more

Article type

- Review articles (40)

Publication title

- Information and Sof (29)
- Advances in Comput
- Journal of Systems a
- Computer Science R
- Trends in Biotechnol

Review article

Identifying, categorizing and mitigating threats to validity in software engineering secondary studies

Information and Software Technology, Volume 106, February 2019, Pages 201-230

Apostolos Ampatzoglou, Stamatia Bibi, Paris Avgeriou, Marijn Verbeek, Alexander Chatzigeorgiou

Review article

Reproducibility and credibility in empirical software engineering: A case study based on a systematic literature review of the use of the SZZ algorithm

Information and Software Technology, Volume 99, July 2018, Pages 164-176

Gema Rodríguez-Pérez, Gregorio Robles, Jesús M. González-Barahona

Review article

A systematic mapping study on game-related methods for software engineering education

Information and Software Technology, Volume 95, March 2018, Pages 201-218

Mauricio R. de A. Souza, Lucas Veado, Renata Teles Moreira, Eduardo Figueiredo, Heitor Costa

Review article

Incorporating user preferences in search-based software engineering: A systematic mapping study

Information and Software Technology, Volume 90, October 2017, Pages 55-69

Thiago Nascimento Ferreira, Silvia Regina Vergilio, Jerffeson Teixeira de Souza

Review article • *Open access*

Taxonomies in software engineering: A Systematic mapping study and a revised taxonomy development method

Information and Software Technology, Volume 85, May 2017, Pages 43-59

Muhammad Usman, Ricardo Britto, Jürgen Börstler, Emilia Mendes

Download PDF

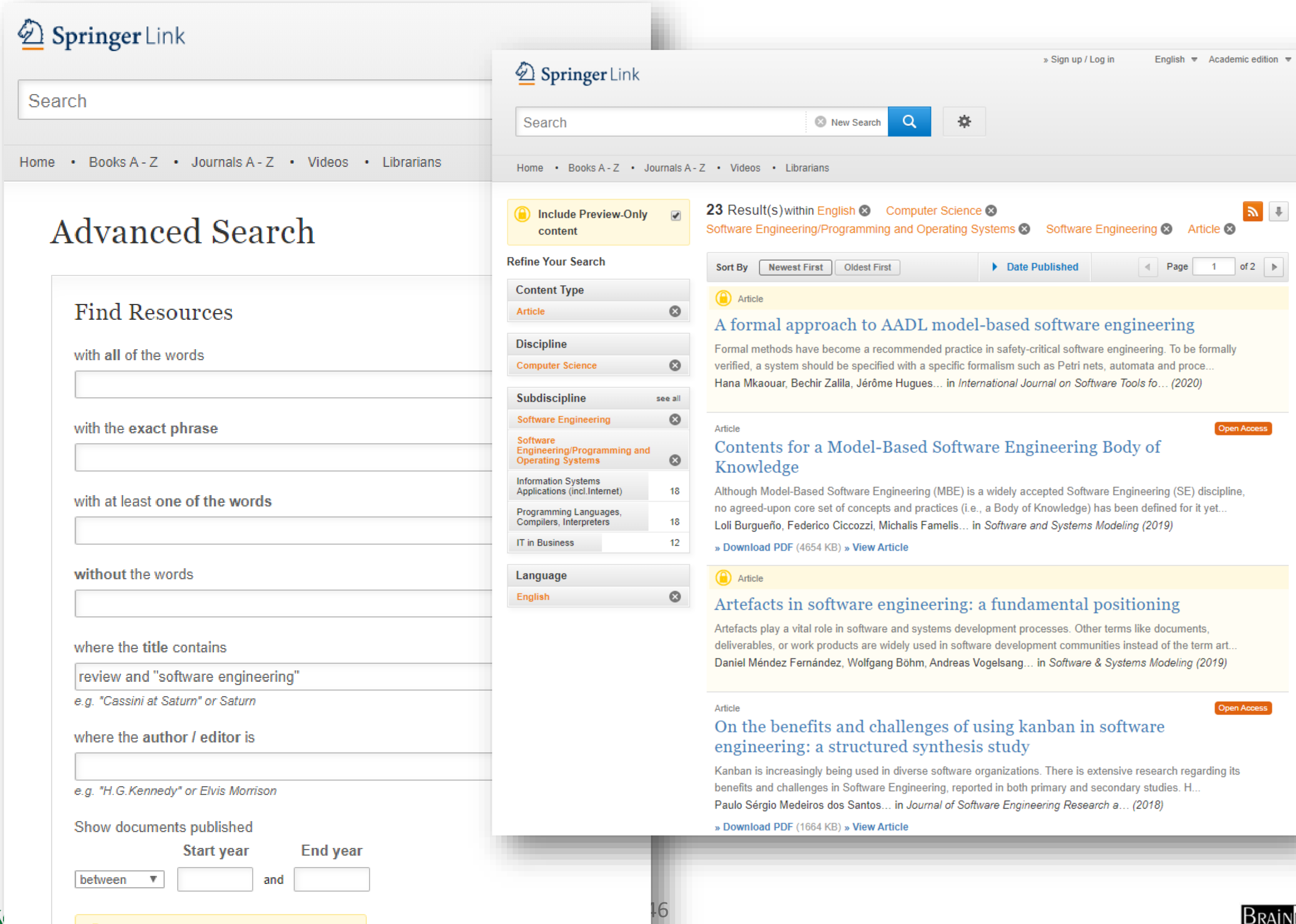
Review article

Challenges and best practices in industry-academia collaborations in software engineering: A systematic literature review

Information and Software Technology, Volume 79, November 2016, Pages 106-127

Vahid Garousi, Kai Petersen, Baris Ozkan

link.springer.com



Science 2.0 and Shadow Library

Subscribe | Latest Issues | SCIENTIFIC AMERICAN | Cart 0 | Sign In | Stay Informed

SHARE

Not logged in Talk Contributions Create account

Article Talk Read Edit View history Search Wikipedia

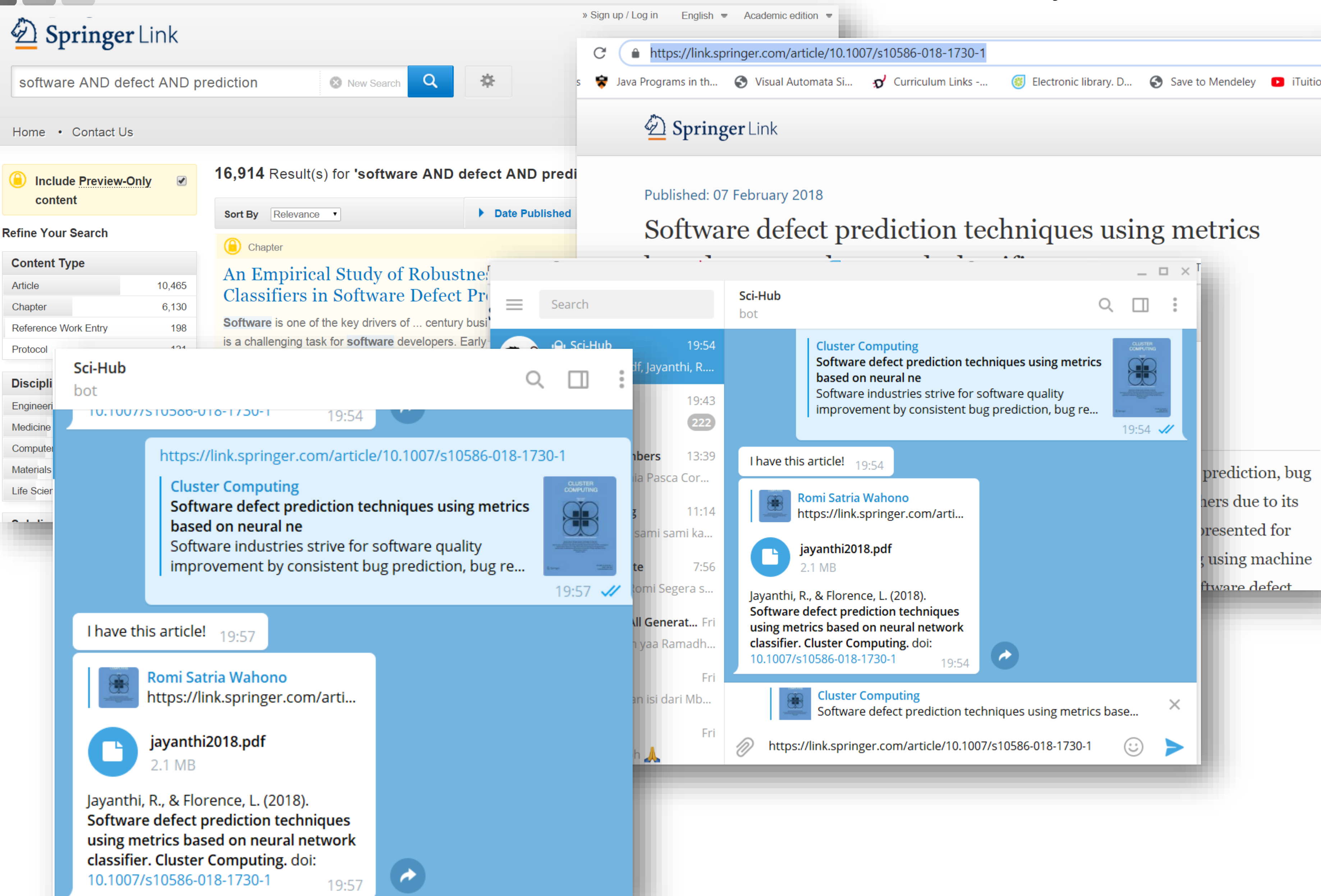
Alexandra Elbakyan

Region	Daily downloads per million inhabitants (2012)	Daily downloads per million inhabitants (2015)	Growth (%)
Europe & Central Asia	28.81	42.87	149%
Middle East & North Africa	8.41	24.31	289%
South Asia	1.22	6.38	523%
Latin America & Caribbean	0.06	11.68	19467%
Sub-Saharan Africa	0.28	3.98	1421%
East Asia & Pacific	2.13	[illegible]	347%
North America	10.47	42.76	408%

Shadow Library Statistics

books. Using a dataset provided to us by one of the administrators of a prominent shadow library, we mapped both the supply of and the demand for academic monographs, textbooks and other learning materials via piratical shadow libraries. Our primary findings suggest that scholarly book piracy is a ubiquitous global phenomenon, with no apparent end in sight. If that is indeed true, what might the consequences be for the status quo in scholarly publishing?

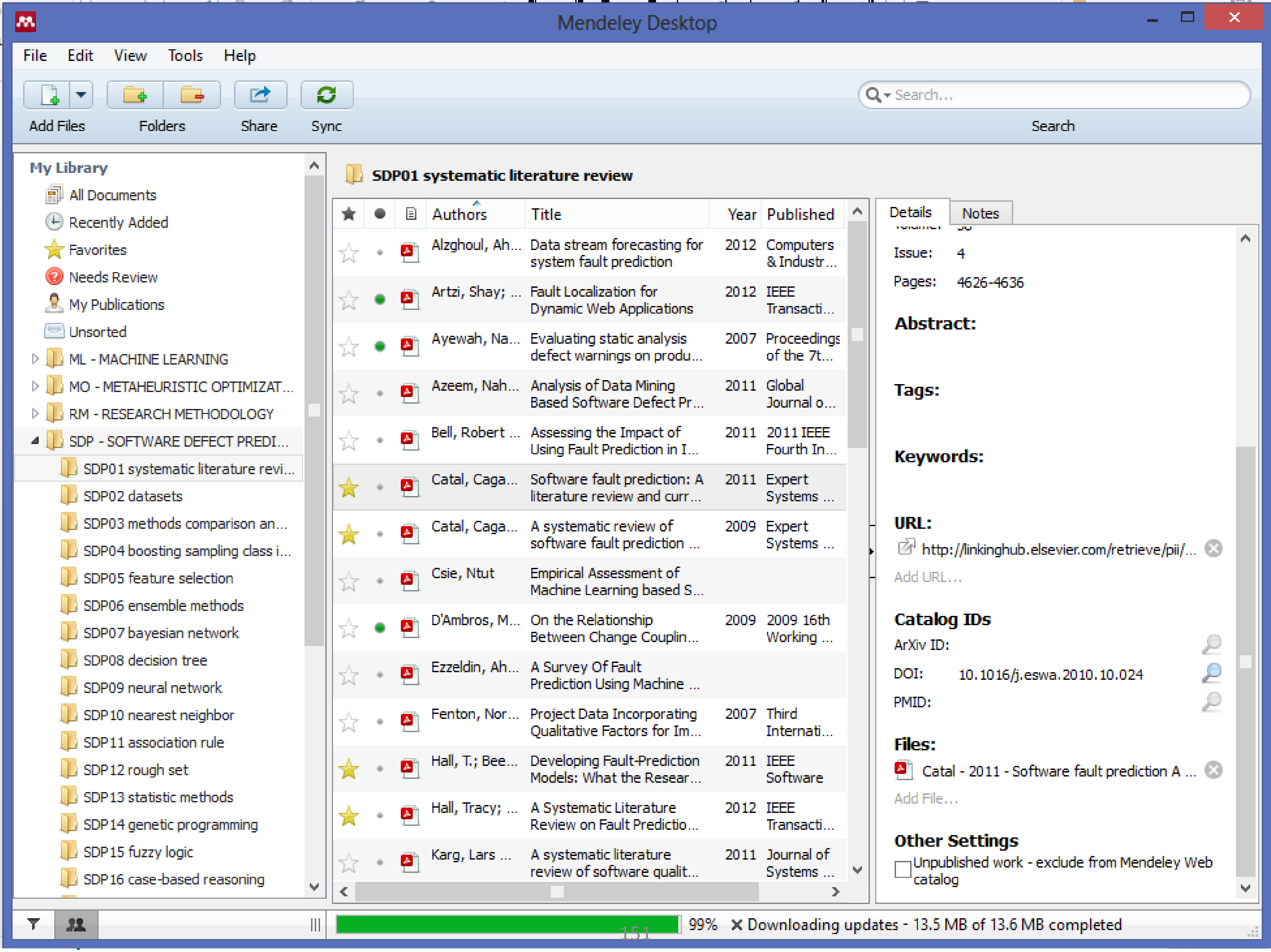
Science 2.0 and Shadow Library



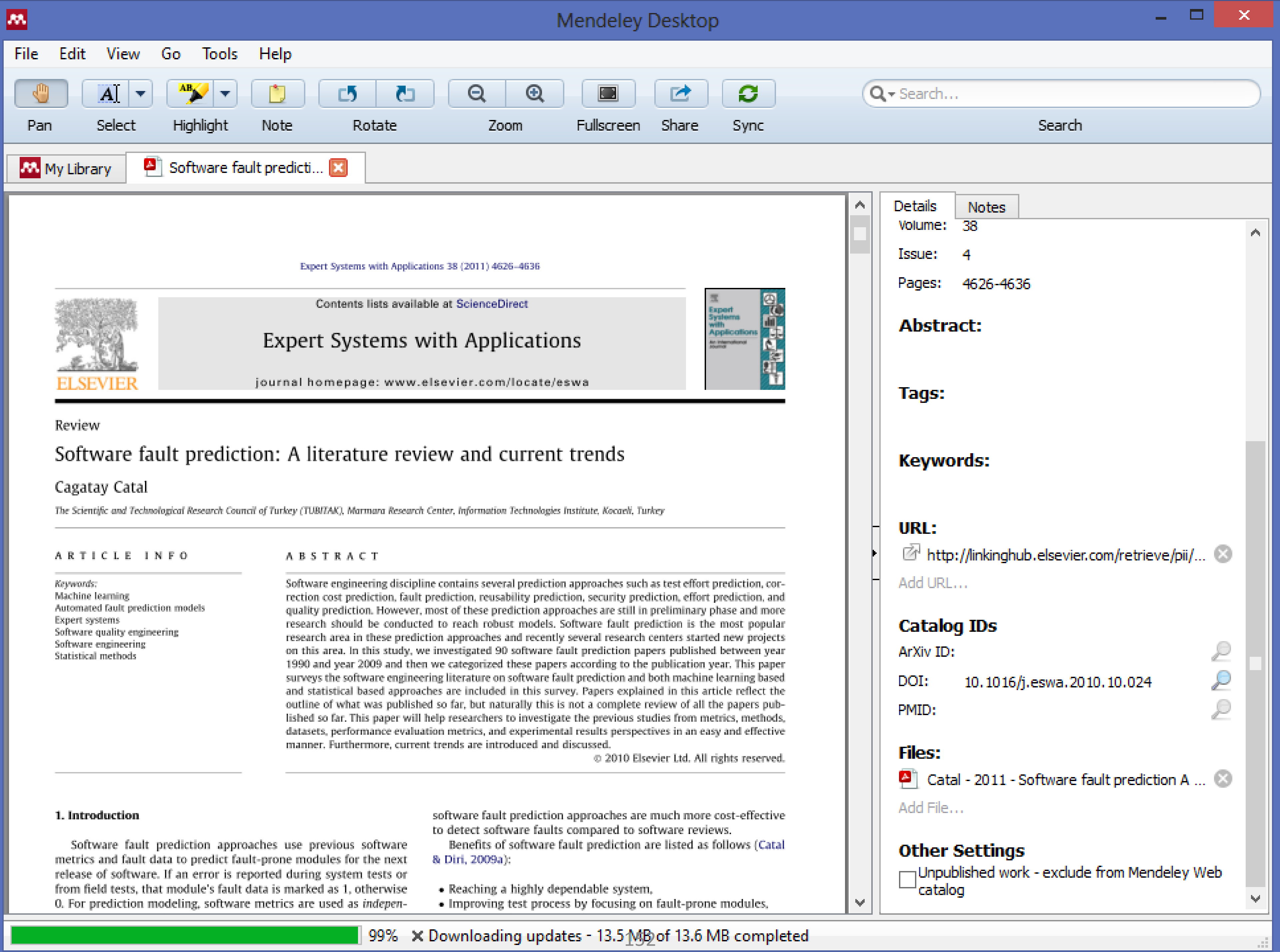
3.2 Teknik Mengelola Paper

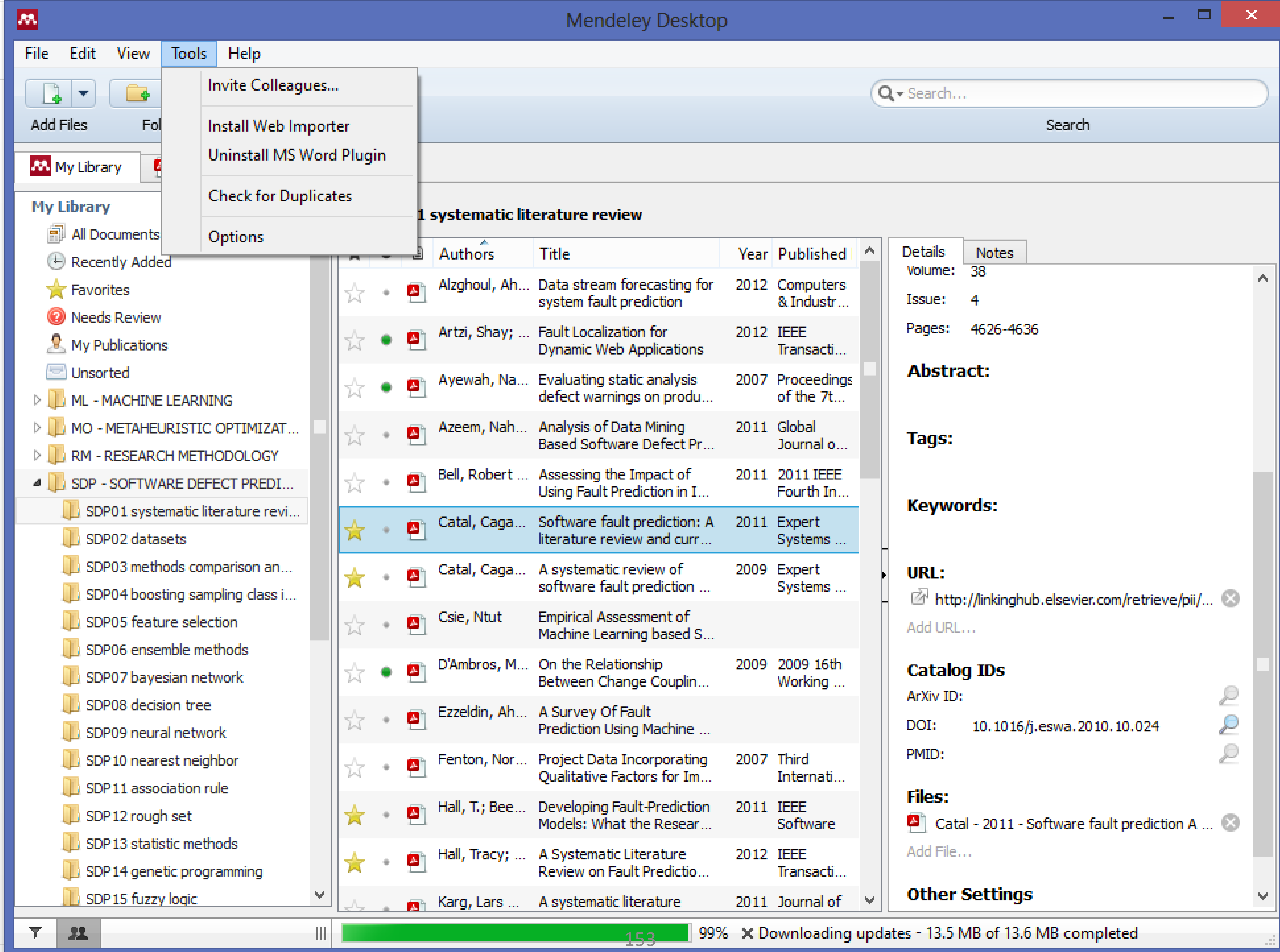
Mengelola Paper Yang Direview

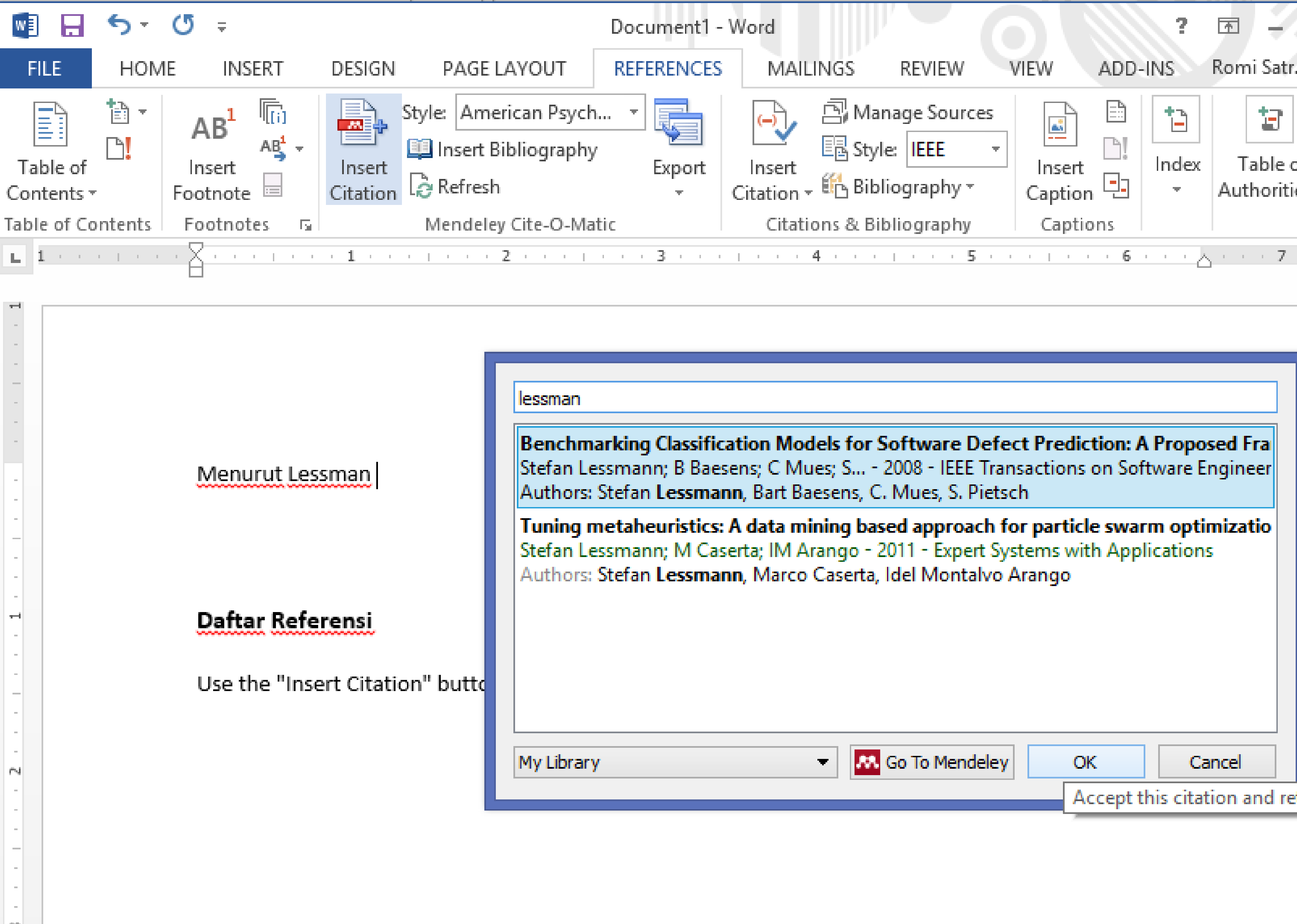
- Buat account di mendeley.com dan download aplikasinya
- Buat folder dan sub folder pada aplikasi mendeley, dan letakkan paper (pdf) yang sudah didownload ke folder yang diinginkan di mendeley
- Bibliografi dari paper akan ditampilkan secara otomatis setelah pdf diletakkan
- Apabila mendeley gagal membaca bibliografi dari paper, cari DOI dari paper, isikan di catalog ID, dan setelah lookup dipilih, bibliografi akan muncul secara otomatis
- Install MS Word Plugin untuk membantu dalam menuliskan sitasi dan daftar referensi

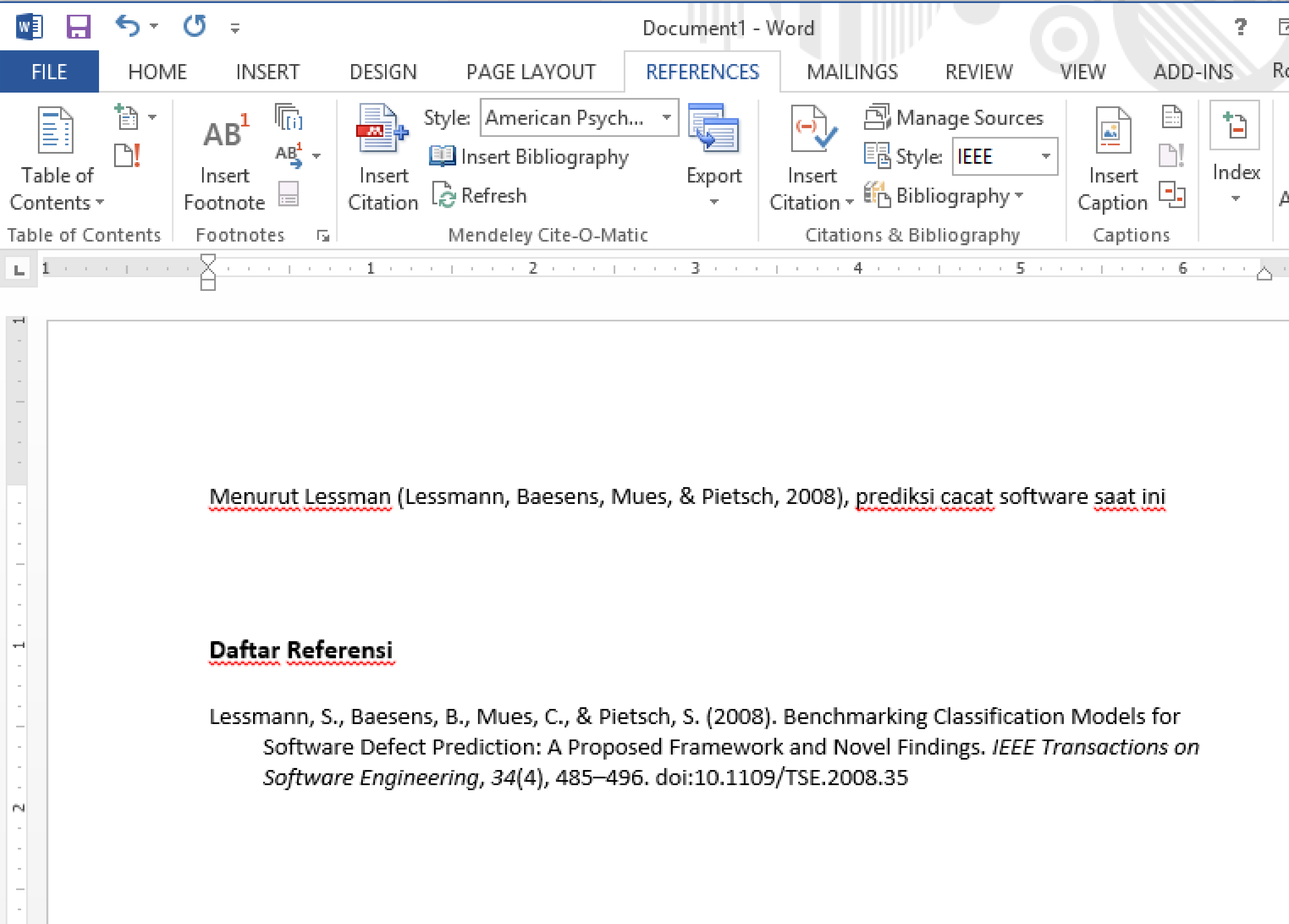


Mendeley Desktop
File Edit View Tools Help
Add Files
Folders
Share
Sync
Search...
Search
My Library
All Documents
Recently Added
Favorites
Needs Review
My Publications
Unsorted
ML - MACHINE LEARNING
MO - METAHEURISTIC OPTIMIZAT...
RM - RESEARCH METHODOLOGY
SDP - SOFTWARE DEFECT PREDI...
SDP01 systematic literature revi...
SDP02 datasets
SDP03 methods comparison an...
SDP04 boosting sampling class i...
SDP05 feature selection
SDP06 ensemble methods
SDP07 bayesian network
SDP08 decision tree
SDP09 neural network
SDP10 nearest neighbor
SDP11 association rule
SDP12 rough set
SDP13 statistic methods
SDP14 genetic programming
SDP15 fuzzy logic
SDP16 case-based reasoning
SDP01 systematic literature review
Authors Title Year Published
Alzghoul, Ah... Data stream forecasting for system fault prediction 2012 Computers & Industr...
Artzi, Shay; ... Fault Localization for Dynamic Web Applications 2012 IEEE Transacti...
Ayewah, Na... Evaluating static analysis defect warnings on produ... 2007 Proceedings of the 7t...
Azeem, Nah... Analysis of Data Mining Based Software Defect Pr... 2011 Global Journal o...
Bell, Robert ... Assessing the Impact of Using Fault Prediction in I... 2011 2011 IEEE Fourth In...
Catal, Caga... Software fault prediction: A literature review and curr... 2011 Expert Systems ...
Catal, Caga... A systematic review of software fault prediction ... 2009 Expert Systems ...
Csie, Ntut Empirical Assessment of Machine Learning based S...
D'Ambros, M... On the Relationship Between Change Couplin... 2009 2009 16th Working ...
Ezzeldin, Ah... A Survey Of Fault Prediction Using Machine ...
Fenton, Nor... Project Data Incorporating Qualitative Factors for Im... 2007 Third Internati...
Hall, T.; Bee... Developing Fault-Prediction Models: What the Resear... 2011 IEEE Software
Hall, Tracy; ... A Systematic Literature Review on Fault Predictio... 2012 IEEE Transacti...
Karg, Lars ... A systematic literature review of software qualit... 2011 Journal of Systems ...
Details Notes
Issue: 4
Pages: 4626-4636
Abstract:
Tags:
Keywords:
URL:
http://linkinghub.elsevier.com/retrieve/pii/...
Add URL...
Catalog IDs
ArXiv ID:
DOI: 10.1016/j.eswa.2010.10.024
PMID:
Files:
Catal - 2011 - Software fault prediction A ...
Add File...
Other Settings
Unpublished work - exclude from Mendeley Web catalog
99% Downloading updates - 13.5 MB of 13.6 MB completed


Mendeley Desktop
File Edit View Go Tools Help
Pan Select Highlight Note Rotate Zoom Fullscreen Share Sync
Search...
Search
My Library
Software fault predicti...
Expert Systems with Applications 38 (2011) 4626–4636
Contents lists available at ScienceDirect
Expert Systems with Applications
journal homepage: www.elsevier.com/locate/eswa
ELSEVIER
Review
Software fault prediction: A literature review and current trends
Cagatay Catal
The Scientific and Technological Research Council of Turkey (TUBITAK), Marmara Research Center, Information Technologies Institute, Kocaeli, Turkey
ARTICLE INFO
Keywords:
Machine learning
Automated fault prediction models
Expert systems
Software quality engineering
Software engineering
Statistical methods
ABSTRACT
Software engineering discipline contains several prediction approaches such as test effort prediction, correction cost prediction, fault prediction, reusability prediction, security prediction, effort prediction, and quality prediction. However, most of these prediction approaches are still in preliminary phase and more research should be conducted to reach robust models. Software fault prediction is the most popular research area in these prediction approaches and recently several research centers started new projects on this area. In this study, we investigated 90 software fault prediction papers published between year 1990 and year 2009 and then we categorized these papers according to the publication year. This paper surveys the software engineering literature on software fault prediction and both machine learning based and statistical based approaches are included in this survey. Papers explained in this article reflect the outline of what was published so far, but naturally this is not a complete review of all the papers published so far. This paper will help researchers to investigate the previous studies from metrics, methods, datasets, performance evaluation metrics, and experimental results perspectives in an easy and effective manner. Furthermore, current trends are introduced and discussed.
© 2010 Elsevier Ltd. All rights reserved.
1. Introduction
Software fault prediction approaches use previous software metrics and fault data to predict fault-prone modules for the next release of software. If an error is reported during system tests or from field tests, that module's fault data is marked as 1, otherwise 0. For prediction modeling, software metrics are used as indepen-
software fault prediction approaches are much more cost-effective to detect software faults compared to software reviews.
Benefits of software fault prediction are listed as follows (Catal & Diri, 2009a):
• Reaching a highly dependable system,
• Improving test process by focusing on fault-prone modules,
Details Notes
Volume: 38
Issue: 4
Pages: 4626-4636
Abstract:
Tags:
Keywords:
URL:
http://linkinghub.elsevier.com/retrieve/pii/...
Add URL...
Catalog IDs
ArXiv ID:
DOI: 10.1016/j.eswa.2010.10.024
PMID:
Files:
Catal - 2011 - Software fault prediction A ...
Add File...
Other Settings
Unpublished work - exclude from Mendeley Web catalog
99% ✕ Downloading updates - 13.5 MB of 13.6 MB completed


Mendeley Desktop
File
Edit
View
Tools
Help
Invite Colleagues...
Install Web Importer
Uninstall MS Word Plugin
Check for Duplicates
Options
Add Files
Search...
Search
My Library
My Library
All Documents
Recently Added
Favorites
Needs Review
My Publications
Unsorted
ML - MACHINE LEARNING
MO - METAHEURISTIC OPTIMIZAT...
RM - RESEARCH METHODOLOGY
SDP - SOFTWARE DEFECT PREDI...
SDP01 systematic literature revi...
SDP02 datasets
SDP03 methods comparison an...
SDP04 boosting sampling class i...
SDP05 feature selection
SDP06 ensemble methods
SDP07 bayesian network
SDP08 decision tree
SDP09 neural network
SDP10 nearest neighbor
SDP11 association rule
SDP12 rough set
SDP13 statistic methods
SDP14 genetic programming
SDP15 fuzzy logic
1 systematic literature review
Authors
Title
Year
Published
Alzghoul, Ah... Data stream forecasting for system fault prediction 2012 Computers & Industr...
Artzi, Shay; ... Fault Localization for Dynamic Web Applications 2012 IEEE Transacti...
Ayewah, Na... Evaluating static analysis defect warnings on produ... 2007 Proceedings of the 7t...
Azeem, Nah... Analysis of Data Mining Based Software Defect Pr... 2011 Global Journal o...
Bell, Robert ... Assessing the Impact of Using Fault Prediction in I... 2011 2011 IEEE Fourth In...
Catal, Caga... Software fault prediction: A literature review and curr... 2011 Expert Systems ...
Catal, Caga... A systematic review of software fault prediction ... 2009 Expert Systems ...
Csie, Ntut Empirical Assessment of Machine Learning based S...
D'Ambros, M... On the Relationship Between Change Couplin... 2009 2009 16th Working ...
Ezzeldin, Ah... A Survey Of Fault Prediction Using Machine ...
Fenton, Nor... Project Data Incorporating Qualitative Factors for Im... 2007 Third Internati...
Hall, T.; Bee... Developing Fault-Prediction Models: What the Resear... 2011 IEEE Software
Hall, Tracy; ... A Systematic Literature Review on Fault Predictio... 2012 IEEE Transacti...
Karg, Lars ... A systematic literature 2011 Journal of
Details
Notes
Volume: 38
Issue: 4
Pages: 4626-4636
Abstract:
Tags:
Keywords:
URL:
http://linkinghub.elsevier.com/retrieve/pii/...
Add URL...
Catalog IDs
ArXiv ID:
DOI: 10.1016/j.eswa.2010.10.024
PMID:
Files:
Catal - 2011 - Software fault prediction A ...
Add File...
Other Settings
99%
Downloading updates - 13.5 MB of 13.6 MB completed


Document1 - Word
FILE
HOME
INSERT
DESIGN
PAGE LAYOUT
REFERENCES
MAILINGS
REVIEW
VIEW
ADD-INS
Romi Satr
Table of Contents
Insert Footnote
Insert Citation
Style: American Psych...
Insert Bibliography
Refresh
Export
Insert Citation
Manage Sources
Style: IEEE
Bibliography
Insert Caption
Index
Table o Authoriti
Table of Contents
Footnotes
Mendeley Cite-O-Matic
Citations & Bibliography
Captions
Menurut Lessman
Daftar Referensi
Use the "Insert Citation" butto
lessman
Benchmarking Classification Models for Software Defect Prediction: A Proposed Fra
Stefan Lessmann; B Baesens; C Mues; S... - 2008 - IEEE Transactions on Software Engineer
Authors: Stefan Lessmann, Bart Baesens, C. Mues, S. Pietsch
Tuning metaheuristics: A data mining based approach for particle swarm optimizatio
Stefan Lessmann; M Caserta; IM Arango - 2011 - Expert Systems with Applications
Authors: Stefan Lessmann, Marco Caserta, Idel Montalvo Arango
My Library
Go To Mendeley
OK
Cancel
Accept this citation and re


Document1 - Word
FILE
HOME
INSERT
DESIGN
PAGE LAYOUT
REFERENCES
MAILINGS
REVIEW
VIEW
ADD-INS
Table of Contents
Insert Footnote
Insert Citation
Style: American Psych...
Insert Bibliography
Refresh
Export
Insert Citation
Manage Sources
Style: IEEE
Bibliography
Insert Caption
Index
Table of Contents
Footnotes
Mendeley Cite-O-Matic
Citations & Bibliography
Captions
Menurut Lessman (Lessmann, Baesens, Mues, & Pietsch, 2008), prediksi cacat software saat ini
Daftar Referensi
Lessmann, S., Baesens, B., Mues, C., & Pietsch, S. (2008). Benchmarking Classification Models for Software Defect Prediction: A Proposed Framework and Novel Findings. IEEE Transactions on Software Engineering, 34(4), 485–496. doi:10.1109/TSE.2008.35

3.3 Teknik Mereview Paper

Jenis Paper Ilmiah

1. Technical Paper
   1. Paper yang isinya adalah hasil penelitian dan eksperimen yang dilakukan seorang peneliti
   2. Penilaian kualitas technical paper dari kontribusi ke pengetahuan

2. Survey Paper
   1. Paper yang isinya adalah review dan survey tentang topik/tema suatu penelitian, biasanya jumlah penelitian yang direview mencapai ratusan atau ribuan
   2. Rujukan dan panduan penting bagi peneliti yang baru memulai penelitian untuk memahami suatu topic/tema penelitian secara komprehensif

3.3.1 Technical Paper

Kiat Mereview Technical Paper

1. Pahami Masalah Penelitian
   - Apakah penelitian hanya menyelesaikan masalah yang dibuat-buat?
   - Apakah masalah penelitian dilandasi dan divalidasi?
2. Pahami Kontribusi
   - Apakah peneliti hanya mengulang hal yang sudah ada?
   - Apakah peneliti menyadari literatur lain yang berhubungan dengan penelitiannya?
   - Apa yang baru dan orisinil di paper itu (metodologi, algoritma, evaluasi, validasi, tool, dsb.)?
3. Pahami Validitas Kontribusi
   - Apakah teori atau model yang diusulkan sudah terbukti benar? Tidak adakah kesalahan pada pembuktian?
   - Adakah faktor-faktor aneh pada proses eksperimen penelitian?
   - Apakah benchmark yang dilakukan realistis atau hanya buatan? Ataukah membandingkan apel dan jeruk?
   - Apakah generalisasi cukup valid?

Masalah Penelitian

- Masalah penelitian adalah alasan utama mengapa penelitian harus dilakukan
- Reviewer jurnal internasional menjadikan “masalah penelitian“sebagai parameter utama proses review
- Masalah penelitian harus objective (tidak subjective), dan harus dibuktikan secara logis dan valid bahwa masalah itu benar-benar masalah
- Supaya logis dan valid, perlu dilakukan objektifikasi masalah, dengan cara melandasi masalah penelitian dengan literature terbaru
- Dimana munculnya di paper:
  - Abstract
  - Introduction

Alur Terbentuknya Masalah Penelitian

- Penelitian dilakukan karena ada masalah penelitian
- Dimana masalah penelitian sendiri muncul karena ada latar belakang masalah penelitian
- Latar belakang masalah penelitian itu berangkatnya bisa dari masalah kehidupan (obyek penelitian)

Contoh Alur Latar Belakang Masalah Penelitian:

Penerapan XYZ untuk Masalah E pada SVM untuk Prediksi Nilai Tukar Uang

- Nilai tukar uang adalah faktor penting pada perekonomian suatu negara. Nilai tukar uang perlu diprediksi supaya kebijakan perekonomian bisa diambil dengan lebih akurat dan efisien…
- Metode untuk prediksi nilai tukar yang saat ini digunakan adalah regresi linier, neural network dan support vector machine…
- Regresi linier memiliki kelebihan A dan kelemahan B…
- Neural network memiliki kelebihan C dan kelemahan D…
- Support vector machine memiliki kelebihan bisa mengatasi masalah B (pada regresi linier) dan D (pada neural network)… tapi memiliki kelemahan E

- Masalah penelitian pada penelitian di atas?
  - Kebijakan perekonomian negara?
  - Prediksi nilai tukar uang?
  - Metode apa yang sebaiknya dipakai untuk prediksi nilai tukar?
- Masalah: Support vector machine memiliki kelebihan memecahkan masalah B dan D (argumentasi dipilih), tapi memiliki kelemahan E
- Tujuan: Menerapkan metode XYZ untuk memecahkan masalah E pada support vector machine

Contoh Alur Latar Belakang Masalah Penelitian:

Penerapan XYZ untuk E pada Fuzzy Logic untuk Pengaturan Lampu Lalu Lintas Dinamis

- Kemacetan lalu lintas di kota besar semakin meningkat
- Penyebab kemacetan adalah traffic light persimpangan jalan
- Traffic light yang ada adalah statis (tetap waktunya) sehingga tidak dapat menyelesaikan kondisi kepadatan kendaraan yang di berbagai waktu
- Traffic light harus didesain dinamis sesuai perubahan berbagai parameter
- Metode untuk menentukan waktu yang tepat secara dinamis dapat menggunakan AHP, ANP, Fuzzy Logic,
- AHP memiliki kelebihan A dan kelemahan B…
- ANP memiliki kelebihan C dan kelemahan D…
- Fuzzy logic memiliki kelebihan bisa mengatasi masalah B (pada AHP) dan D (pada ANP)… tapi memiliki kelemahan E

- Masalah penelitian pada penelitian di atas?
  - Bagaimana mengatasi kemacetan lalu lintas?
  - Bagaimana mendesain traffic light?
  - Metode apa yang sebaiknya dipakai untuk penentuan traffic light secara dinamis?
- Masalah: Fuzzy logic memiliki kelebihan memecahkan masalah B dan D (argumentasi dipilih), tapi memiliki kelemahan E
- Tujuan: Menerapkan metode XYZ untuk memecahkan masalah E pada fuzzy logic

Contoh Masalah Penelitian

- ***Ungu***: *Obyek Data (Opsional, Bisa Dataset Publik)*
- ***Oranye***: *Topik (Obyek Metode yang Diperbaiki)*
- ***Merah***: *Masalah Penelitian*
- ***Hijau***: *Metode Perbaikan yang Diusulkan*
- ***Biru***: *Pengukuran Penelitian (Tidak Harus Masuk Judul)*

Penerapan Particle Swarm Optimization untuk Pemilihan Parameter Secara Otomatis pada Support Vector Machine untuk Prediksi Produksi Padi

Research Problem (RP)	Research Question (RQ)	Research Objective (RO)
SVM dapat memecahkan masalah 'over-fitting', lambatnya konvergensi, dan sedikitnya data training, akan tetapi memiliki kelemahan pada sulitnya pemilihan parameter SVM yang sesuai yang mengakibatkan akurasi tidak stabil	Seberapa meningkat akurasi metode SVM apabila PSO diterapkan pada proses pemilihan parameter?	Menerapkan PSO untuk pemilihan parameter yang sesuai pada SVM (C, lambda dan epsilon) , sehingga hasil prediksinya lebih akurat

Contoh Masalah Penelitian

- ***Ungu***: *Obyek Data (Opsional, Bisa Dataset Publik)*
- ***Oranye***: *Topik (Obyek Metode yang Diperbaiki)*
- ***Merah***: *Masalah Penelitian*
- ***Hijau***: *Metode Perbaikan yang Diusulkan*
- ***Biru***: *Pengukuran Penelitian (Tidak Harus Masuk Judul)*

- Masalah Penelitian (*Research Problem*):
  - Neural network terbukti memiliki performa bagus untuk menangani data besar seperti pada data prediksi harga saham, akan tetapi memiliki kelemahan pada pemilihan arsitektur jaringannya yang harus dilakukan secara trial error, sehingga tidak efisien dan mengakibatkan hasil prediksi kurang akurat
- Rumusan Masalah (*Research Question*):
  - Bagaimana peningkatan akurasi dan efisiensi neural network apabila pada pemilihan arsitektur jaringan diotomatisasi menggunakan algoritma genetika?
- Tujuan Penelitian (*Research Objective*):
  - Menerapkan algoritma genetika untuk mengotomatisasi pemilihan arsitektur jaringan pada neural nework sehingga lebih efisien dan hasil prediksi lebih akurat

Contoh Masalah Penelitian

- ***Ungu***: *Obyek Data (Opsional, Bisa Dataset Publik)*
- ***Oranye***: *Topik (Obyek Metode yang Diperbaiki)*
- ***Merah***: *Masalah Penelitian*
- ***Hijau***: *Metode Perbaikan yang Diusulkan*
- ***Biru***: *Pengukuran Penelitian (Tidak Harus Masuk Judul)*

- Research Problem (RP):
  - Algoritma K-Means merupakan algoritma clustering yang populer karena efisien dalam komputasi, akan tetapi memiliki kelemahan pada sulitnya penentuan K yang optimal dan komputasi yang tidak efisien bila menangani data besar (Zhao, 2010)
- Research Question (RQ):
  - Seberapa efisien algoritma Bee Colony bila digunakan untuk menentukan nilai K yang optimal pada K-Means?
- Research Objective (RO):
  - Menerapkan algoritma bee colony untuk menentukan nikai K yang optimal pada K-Means sehingga komputasi lebih efisien

Masalah Penelitian dan Landasannya

Masalah Penelitian	Landasan Literatur
Data set pada prediksi cacat software berdimensi tinggi, memiliki atribut yang bersifat noisy, dan classnya bersifat tidak seimbang, menyebabkan penurunan akurasi pada prediksi cacat software	There are noisy data points in the software defect data sets that can not be confidently assumed to be erroneous using such simple method *(Gray, Bowes, Davey, & Christianson, 2011)*
	The performances of software defect prediction improved when irrelevant and redundant attributes are removed *(Wang, Khoshgoftaar, & Napolitano, 2010)*
	The software defect prediction performance decreases significantly because the dataset contains noisy attributes *(Kim, Zhang, Wu, & Gong, 2011)*
	Software defect datasets have an imbalanced nature with very few defective modules compared to defect-free ones *(Tosun, Bener, Turhan, & Menzies, 2010)*
	Imbalance can lead to a model that is not practical in software defect prediction, because most instances will be predicted as non-defect prone *(Khoshgoftaar, Van Hulse, & Napolitano, 2011)*
	Software fault prediction data sets are often highly imbalanced *(Zhang & Zhang, 2007)*

Formulasi RP-RQ-RO

Research Problems (RP)		Research Questions (RQ)		Research Objectives (RO)	
RP	Data set pada prediksi cacat software berdimensi tinggi, dan memiliki atribut yang bersifat noisy, serta classnya bersifat tidak balance	RQ1	Algoritma pemilihan fitur apa yang performanya terbaik untuk meyelesaikan masalah atribut yang noisy pada prediksi cacat software?	RO1	Untuk mengidentifikasi algoritma pemilihan fitur apa yang memiliki performa terbaik apabila digunakan untuk menyelesaikan masalah atribut yang noisy pada prediksi cacat software
		RQ2	Algoritma meta learning apa yang performanya terbaik untuk menyelesaikan masalah class imbalance pada prediksi cacat software?	RO2	Untuk mengidentifikasi algoritma meta learning apa yang memiliki performa terbaik apabila digunakan untuk menyelesaikan masalah class imbalance pada prediksi cacat software
		RQ3	Bagaimana pengaruh penggabungan algoritma pemilihan fitur dan metode meta learning apabila digunakan untuk prediksi cacat software?	RO3	Untuk mengembangkan algoritma baru yang menggabungkan algoritma pemilihan fitur dan meta learning untuk prediksi cacat software

Latihan Mereview Paper

- Technical Paper:
  - Judul: Chinese Grain Production Forecasting Method Based on Particle Swarm Optimization-based Support Vector Machine
  - Author: Sheng-Wei Fei, Yu-Bin Miao and Cheng-Liang Liu
  - Publications: Recent Patents on Engineering 2009, 3, 8-12
  - Download: **http://romisatriawahono.net/lecture/rm/paper/**

- Tugas:

  Pahami dan rangkumkan paper di atas dalam 3 slide:

  1. Masalah penelitian
  2. Metode-metode yang ada (State-of-the-art Method)
  3. Metode yang diusulkan
  4. Metode pengukuran penelitian

Prediksi Produksi Padi dengan SVM berbasis PSO

- Obyek Data: Padi
- Latar Belakang: Prediksi Produksi Padi
- Metode Yang Ada (State-of-the-Art Methods):
  - Konvensional: Remote Sensing, Statistik
    - Masalah: tingkat error tinggi, periode pendek
  - Time Series: NN, GM, SVM
    - SVM itu bisa mengatasi masalah yang ada di NN dan GM
- Masalah:
  - SVM itu bisa mengatasi masalah yang ada di NN dan GM, akan tetapi memiliki kelemahan pada pemilihan parameter (C, e, gamma)

Latihan Mereview Paper

- Technical Paper:
  - Judul: Resampling Logistic Regression untuk Penanganan Ketidakseimbangan Class pada Prediksi Cacat Software
  - Author: Harsih Rianto dan Romi Satria Wahono
  - Publications: Journal of Software Engineering, Vol. 1, No. 1, April 2015
  - Download: **http://romisatriawahono.net/lecture/rm/paper/**

- Tugas:

  Pahami dan rangkumkan paper di atas dalam 3 slide:

  1. Masalah penelitian
  2. Metode-metode yang ada (State-of-the-art Method)
  3. Metode yang diusulkan
  4. Metode pengukuran penelitian

Latihan Mereview Paper

- Technical Paper:
  - Judul: Genetic Algorithms With Guided and Local Search Strategies for University Course Timetabling
  - Author: Shengxiang Yang and Sadaf Naseem Jat
  - Publications: IEEE Transactions on Systems, Man and Cybernetics Vol. 41, No. 1, 2011
  - Download: **http://romisatriawahono.net/lecture/rm/paper/**

- Tugas:

  Pahami dan rangkumkan paper di atas dalam 3 slide:

  1. Masalah penelitian
  2. Metode-metode yang ada (State-of-the-art Method)
  3. Metode yang diusulkan
  4. Metode pengukuran penelitian

Latihan Mereview Paper

- Technical Paper:
  - Judul: Integrasi Kromosom Buatan Dinamis untuk Memecahkan Masalah Konvergensi Prematur pada Algoritma Genetika untuk Traveling Salesman Problem
  - Author: Muhammad Rikzam Kamal dan Romi Satria Wahono
  - Publications: Journal of Intelligent Systems, Vol. 1, No. 2, December 2015
  - Download: **http://romisatriawahono.net/lecture/rm/paper/**

- Tugas:

  Pahami dan rangkumkan paper di atas dalam 3 slide:

  1. Masalah penelitian
  2. Metode-metode yang ada (State-of-the-art Method)
  3. Metode yang diusulkan
  4. Metode pengukuran penelitian

Latihan Mereview Paper

- Technical Paper:
  - Judul: Credal-C4.5: Decision tree based on imprecise probabilities to classify noisy data
  - Author: Carlos J. Mantas, Joaquín Abellán
  - Publications: Expert Systems with Applications 41 (2013) 4625–4627
  - Download: **http://romisatriawahono.net/lecture/rm/paper/**

- Tugas:
  Pahami dan rangkumkan paper di atas dalam 3 slide:
  1. Masalah penelitian
  2. Metode-metode yang ada (State-of-the-art Method)
  3. Metode yang diusulkan
  4. Metode pengukuran penelitian

Latihan Mereview Paper

- Technical Paper:
  - Judul: Penerapan Metode Average Gain, Threshold Pruning dan Cost Complexity Pruning untuk Split Atribut pada Algoritma C4.5
  - Author: Erna Rahayu dan Romi Satria Wahono
  - Publications: Journal of Intelligent Systems, Vol. 1, No. 2, December 2015
  - Download: **http://romisatriawahono.net/lecture/rm/paper/**

- Tugas:

  Pahami dan rangkumkan paper di atas dalam 3 slide:

  1. Masalah penelitian
  2. Metode-metode yang ada (State-of-the-art Method)
  3. Metode yang diusulkan
  4. Metode pengukuran penelitian

Latihan Mereview Paper

- Technical Paper:
  - Judul: Genetic Feature Selection for Software Defect Prediction
  - Author: Romi Satria Wahono and Nanna Suryana Herman
  - Publications: Advanced Science Letters, Vol 20 No 1, 2014
  - Download: **http://romisatriawahono.net/lecture/rm/paper/**

- Tugas:

  Pahami dan rangkumkan paper di atas dalam 3 slide:

  1. Masalah penelitian
  2. Metode-metode yang ada (State-of-the-art Method)
  3. Metode yang diusulkan
  4. Metode pengukuran penelitian

Latihan Mereview Paper

- Technical Paper:
  - Judul: Particle swarm optimization for parameter determination and feature selection of support vector machines
  - Author: Shih-Wei Lin, Kuo-Ching Ying, Shih-Chieh Chen, and Zne-Jung Lee
  - Publications: Expert Systems with Applications 35 (2008) 1817–1824
  - Download: **http://romisatriawahono.net/lecture/rm/paper/**

- Tugas:
  Pahami dan rangkumkan paper di atas dalam 3 slide:
  1. Masalah penelitian
  2. Metode-metode yang ada (State-of-the-art Method)
  3. Metode yang diusulkan
  4. Metode pengukuran penelitian

Tugas Mereview Paper

REVIEW PAPER

Penulis, Judul, Jurnal, Tahun

JIF/SJR:

LATAR BELAKANG MASALAH

- Obyek Penelitian (What- Why)
- Metode-Metode yg Ada
- Analisis Gap Metode yg Ada

- Rangkum **Masalah Penelitian**

MASALAH PENELITIAN (MP) DAN LANDASAN

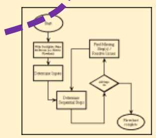
MP	LANDASAN
	Landasan (blablabla, 2014)
	Blablabla (john, 2013)
	Blablabla (Yang, 2014)

PENELITIAN YANG BERHUBUNGAN (PYB)

1. Metode A (2014)
2. Metode B (2015)
3. Metode C (2016)

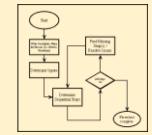
METODE A (2014)

- deskripsi
- deskripsi
- deskripsi
- deskripsi
- deskripsi



METODE B (2015)

- deskripsi
- deskripsi
- deskripsi
- deskripsi
- deskripsi



METODE C (2016)

- deskripsi
- deskripsi
- deskripsi
- deskripsi
- deskripsi

METODE YANG DIUSULKAN (MYD)

- deskripsi
- deskripsi
- deskripsi
- deskripsi
- deskripsi

ANALISIS PERBEDAAN PYB DAN MYD

M	P 1	P 2	P 3
A			
B			
C			
MYD			

TAHAPAN PENELITIAN

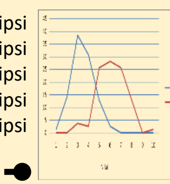
1. Tahap 1
2. Tahap 2
3. Tahap 3
4. Tahap 4

DATASET

- Penjelasan Dataset
- Penjelasan Dataset
- Penjelasan Dataset

HASIL PENELITIAN

- deskripsi
- deskripsi
- deskripsi
- deskripsi
- deskripsi



KESIMPULAN

- Kesimpulan 1
- Kesimpulan 2
- Kesimpulan 3

TUGAS:

1. Pilih Satu Technical Paper dari Topik Penelitian yang Dipilih, yang Terbit di Jurnal Terindeks ISI/SCOPUS 3 Tahun Terakhir, Rangkum dengan Format di Atas
2. Rangkumkan SLR dengan Topik yang Sesuai Paper yang Dipilih (No 1)
3. Rangkumkan Bagian Buku Teks yang Sesuai dengan Paper

3.3.2 Survey Paper

Literature Review

- This literature review aims to identify and analyze the state-of-the-art research and methods in the field of interest
- Type of Literature Review:
    1. Traditional Review
    2. Systematic Literature Review or Systematic Review
    3. Systematic Mapping Study (Scoping Study)
    4. Tertiary Study
- SLR is now well established review method in the field of software engineering

(Kitchenham & Charters, Guidelines in performing Systematic Literature Reviews in Software Engineering, EBSE Technical Report version 2.3, 2007)

1. Traditional Review

- Provides an overview of the research findings on particular topics
- Advantages: produce insightful, valid syntheses of the research literature if conducted by the expert
- Disadvantages: vulnerable to unintentional and intentional bias in the selection, interpretation and organization of content
- Examples:
    - Liao et al., Intrusion Detection System: A Comprehensive Review, Journal of Network and Computer Applications, 36(2013)
    - Galar et al., A Review on Ensembles for the Class Imbalance Problem: Bagging-, Boosting-, and Hybrid-Based Approaches, IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews), Vol. 42, No. 4, July 2012
    - Cagatay Catal, Software fault prediction: A literature review and current trends, Expert Systems with Applications 38 (2011)

2. Systematic Mapping Study

- Suitable for a very broad topic
- Identify clusters of evidence (making classification)
- Direct the focus of future SLRs
- To identify areas for future primary studies
- Examples:
  - Neto et al., A systematic mapping study of software product lines testing, Information and Software Technology Vol. 53, Issue 5, May 2011
  - Elberzhager et al., Reducing test effort: A systematic mapping study on existing approaches, Information and Software Technology 54 (2012)

3. Systematic Literature Review (SLR)

- The purpose of a systematic literature reviews is to provide as complete a list as possible of all the published studies relating to a particular subject area
- A process of identifying, assessing, and interpreting all available research evidence, to provide answers for a particular research question
- A form of secondary study that uses a well-defined methodology
- SLRs are well established in other disciplines, particularly medicine. They integrate an individual clinical expertise and facilitate access to the outcomes of the research

(Kitchenham & Charters, Guidelines in performing Systematic Literature Reviews in Software Engineering, EBSE Technical Report version 2.3, 2007)

3. Systematic Literature Review (SLR)

Examples of SLR:

- Hall et al., A Systematic Literature Review on Fault Prediction Performance in Software Engineering, IEEE Transaction on Software Engineering, Vol. 38, No. 6, 2012
- Romi Satria Wahono, A Systematic Literature Review of Software Defect Prediction: Research Trends, Datasets, Methods and Frameworks, Journal of Software Engineering, Vol. 1, No. 1, April 2015
- Matthias Galster, Danny Weyns, Dan Tofan, Bartosz Michalik, and Paris Avgeriou, Variability in Software Systems: A Systematic Literature Review, IEEE Transactions on Software Engineering, Vol 40, No 3, 2014

4. Tertiary study

- Is a SLR of SLRs
- To answer a more wider question
- Uses the same method as in SLR
- Potentially less resource intensive
- Examples:
  - Kitchenham et al., Systematic literature reviews in software engineering – A tertiary study, Information and Software Technology 52 (2010)
  - Cruzes et al., Research synthesis in software engineering: A tertiary study, Information and Software Technology 53 (2011)

Kiat Mereview Paper Survey

- Pahami Research Question (RQ) yang biasanya tertulis secara eksplisit di paper
- Jawaban RQ ada di bagian "result and analysis" di halaman belakang
- Perhatikan pelan-pelan apabila RQ ada tentang "best method/algorithm" karena di situ akan dibahas tentang state-of-the-art method
- Perhatikan juga RQ tentang "research challenge/problems", karena di situ kita bisa menemukan masalah penelitian terkini (state-of-the-art problem)

Contoh Survey Paper

Contents lists available at SciVerse ScienceDirect

Information and Software Technology

journal homepage: www.elsevier.com/locate/infsof

Systematic literature review of machine learning based software development effort estimation models

Jianfeng Wen [a,*], Shixian Li [a], Zhiyong Lin [b], Yong Hu [c], Changqin Huang [d]

[a] *Department of Computer Science, Sun Yat-sen University, Guangzhou, China*
[b] *Department of Computer Science, Guangdong Polytechnic Normal University, Guangzhou, China*
[c] *Institute of Business Intelligence and Knowledge Discovery, Department of E-commerce, Guangdong University of Foreign Studies, Sun Yat-sen University, Guangzhou, China*
[d] *Engineering Research Center of Computer Network and Information Systems, South China Normal University, Guangzhou, China*

ARTICLE INFO

Article history:
Received 28 October 2010
Received in revised form 8 August 2011
Accepted 8 September 2011
Available online 16 September 2011

Keywords:
Software effort estimation
Machine learning
Systematic literature review

ABSTRACT

Context: Software development effort estimation (SDEE) is the process of predicting the effort required to develop a software system. In order to improve estimation accuracy, many researchers have proposed machine learning (ML) based SDEE models (ML models) since 1990s. However, there has been no attempt to analyze the empirical evidence on ML models in a systematic way.
Objective: This research aims to systematically analyze ML models from four aspects: type of ML technique, estimation accuracy, model comparison, and estimation context.
Method: We performed a systematic literature review of empirical studies on ML model published in the last two decades (1991–2010).
Results: We have identified 84 primary studies relevant to the objective of this research. After investigating these studies, we found that eight types of ML techniques have been employed in SDEE models. Overall speaking, the estimation accuracy of these ML models is close to the acceptable level and is better than that of non-ML models. Furthermore, different ML models have different strengths and weaknesses and thus favor different estimation contexts.
Conclusion: ML models are promising in the field of SDEE. However, the application of ML models in industry is still limited, so that more effort and incentives are needed to facilitate the application of ML models. To this end, based on the findings of this review, we provide recommendations for researchers as well as guidelines for practitioners.

Contoh Survey Paper

IEEE TRANSACTIONS ON SOFTWARE ENGINEERING, VOL. 38, NO. 6, NOVEMBER/DECEMBER 2

A Systematic Literature Review on Fault Prediction Performance in Software Engineering

Tracy Hall, Sarah Beecham, David Bowes, David Gray, and Steve Counsell

Abstract—*Background*: The accurate prediction of where faults are likely to occur in code can help direct test effort, reduce costs, and improve the quality of software. *Objective*: We investigate how the context of models, the independent variables used, and the modeling techniques applied influence the performance of fault prediction models. *Method:* We used a systematic literature review to identify 208 fault prediction studies published from January 2000 to December 2010. We synthesize the quantitative and qualitative results of 36 studies which report sufficient contextual and methodological information according to the criteria we develop and apply. *Results:* The models that perform well tend to be based on simple modeling techniques such as Naive Bayes or Logistic Regression. Combinations of independent variables have been used by models that perform well. Feature selection has been applied to these combinations when models are performing particularly well. *Conclusion:* The methodology used to build models seems to be influential to predictive performance. Although there are a set of fault prediction studies in which confidence is possible, more studies are needed that use a reliable methodology and which report their context, methodology, and performance comprehensively.

Latihan Mereview Paper Survey

- Survey Paper:
  - Judul: Intrusion Detection System: A Comprehensive Review
  - Author: Hung-Jen Liao, Chun-Hung Richard Lin, Ying-ChihLin, Kuang-YuanTung
  - Publications: Journal of Network and Computer Applications, 36(2013)
  - Download: **http://romisatriawahono.net/lecture/rm/survey/**

- Tugas:
  - Pahami dan rangkumkan paper di atas dalam bentuk slide dengan format:
    1. Identifikasi Research Question (RQ)
    2. Analisis jawaban dari Research Question (RQ)

Latihan Mereview Paper Survey

- Survey Paper:
  - Judul: A Systematic Literature Review of Software Defect Prediction: Research Trends, Datasets, Methods and Frameworks
  - Author: Romi Satria Wahono
  - Publications: Journal of Software Engineering, Vol. 1, No. 1, April 2015
  - Download: **http://romisatriawahono.net/lecture/rm/survey/**

- Tugas:

  Pahami dan rangkumkan paper di atas dalam bentuk slide dengan format:

  1. Identifikasi Research Question (RQ)
  2. Analisis jawaban dari Research Question (RQ)

Latihan Mereview Paper Survey

- Survey Paper:
  - Judul: Systematic literature review of machine learning based software development effort estimation models
  - Author: Jianfeng Wen, Shixian Li, Zhiyong Lin, Yong Hu, Changqin Huang
  - Publications: Information and Software Technology 54 (2012) 41–59
  - Download: **http://romisatriawahono.net/lecture/rm/survey/**

- Tugas:

  Pahami dan rangkumkan paper di atas dalam bentuk slide dengan format:

  1. Identifikasi Research Question (RQ)
  2. Analisis jawaban dari Research Question (RQ)

Latihan Mereview Paper Survey

- Survey Paper:
  - Judul: Variability in Software Systems: A Systematic Literature Review
  - Author: Matthias Galster, Danny Weyns, Dan Tofan, Bartosz Michalik, and Paris Avgeriou
  - Publications: IEEE Transactions on Software Engineering, Vol 40, No 3, 2014
  - Download: **http://romisatriawahono.net/lecture/rm/survey/**

- Tugas:
  - Pahami dan rangkumkan paper di atas dalam bentuk slide dengan format:
    1. Identifikasi Research Question (RQ)
    2. Analisis jawaban dari Research Question (RQ)

Latihan Mereview Paper Survey

- Survey Paper:
  - Judul: A Systematic Literature Review on Fault Prediction Performance in Software Engineering
  - Author: Tracy Hall, Sarah Beecham, David Bowes, David Gray, and Steve Counsell
  - Publications: IEEE Transaction on Software Engineering, Vol. 38, No. 6, 2012
  - Download: **http://romisatriawahono.net/lecture/rm/survey/**
- Tugas:

  Pahami dan rangkumkan paper di atas dalam bentuk slide dengan format:

  1. Identifikasi Research Question (RQ)
  2. Analisis jawaban dari Research Question (RQ)

4. Penulisan Ilmiah

4.1 Mengapa Penulian dan Publikasi Ilmiah?

4.2 Sitasi dan Penulisan Referensi

4.3 Struktur dan Kiat Penulisan Tesis

4.4 Kiat dan Prosedur Publikasi Ilmiah untuk Jurnal Internasional

4.5 Penulisan Systematic Literature Review (SLR)

4.1 Mengapa Penulisan dan Publikasi Ilmiah?

Mengapa Melakukan Publikasi Ilmiah?

"A paper is an organized description of hypotheses, data and conclusions, intended to instruct the reader. If your research does not generate papers, it might just as well not have been done"

(Whitesides, 2004)

"If it wasn't published, it wasn't done"

(Miller 1993)

Mengapa Melakukan Publikasi Ilmiah?

Surat Edaran Dirjen Dikti No. 152/E/T/2012 tentang Publikasi Karya Ilmiah:

"Terhitung kelulusan setelah Agustus 2012, untuk lulusan program sarjana harus menghasilkan makalah yang terbit pada jurnal ilmiah"

Tujuan Menerbitkan Jurnal Berkala

- Meregistrasi kegiatan kecendekiaan
- Menyertifikasi hasil kegiatan yang memenuhi persyaratan ilmiah
- Mendiseminasikannya secara meluas kepada khalayak ramai
- Mengarsipkan semua temuan hasil kegiatan kecendekiaan ilmuwan dan pandit yang dimuatnya

(Permendiknas no 22/2011 tentang terbitan berkala)

Peraturan Diknas

- Surat Edaran Dirjen Dikti No. 152/E/T/2012 tentang Publikasi Karya Ilmiah
- Surat Edaran Dirjen DIKTI No. 2050/E/T/2011 tentang Kebijakan Unggah Karya Ilmiah dan Jurnal
- Permendiknas No. 17 Tahun 2010 tentang Pencegahan dan Penanggulangan Plagiat di Perguruan Tinggi
- Permendiknas No. 22 Tahun 2011 tentang Terbitan Berkala Ilmiah
- Peraturan Dirjen Dikti No. 29/DIKTI/Kep/2011 tentang Pedoman Akreditasi Berkala Ilmiah

Kualitas Peneliti Ditentukan oleh Apa?

Jumlah paper yang diterbitkan di journal berimpact factor tinggi

Jumlah citation ke paper seorang peneliti dari paper peneliti lain

Romi Satria Wahono

IKUTI

BUAT PROFIL SAYA

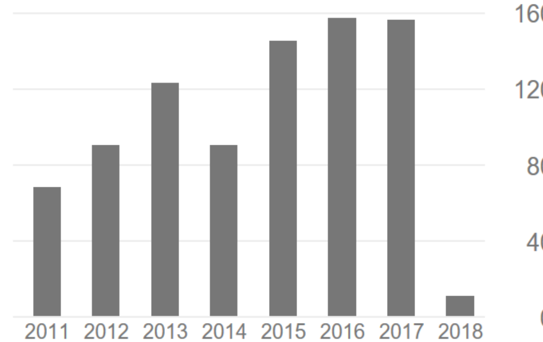
Brainmatics Cipta Informatika, Universitas Dian Nuswantoro, Indonesia
Email yang diverifikasi di brainmatics.com - Beranda

Software Engineering   Machine Learning

JUDUL	DIKUTIP OLEH	TAHUN
Penerapan Jaringan Syaraf Tiruan Backpropagation Dalam Prediksi Produksi Bahan Pangan Pokok di Indonesia NO Syamsiah, RS Wahono Jurnal Teknik Informatika 3 (1), 1-7		2017
PENGUKURAN MATURITAS PENGEMBANGAN PERANGKAT LUNAK MELALUI PENDEKATAN INTEGRASI CAPABILITY MATURITY MODEL INTEGRATION DAN... B Waseso, RS Wahono Jurnal Informatika dan Komputasi 11 (1), 33-40		2017
TACKLING IMBALANCED CLASS IN SOFTWARE DEFECT PREDICTION USING TWO-STEP CLUSTER BASED RANDOM UNDERSAMPLING AND STACKING TE... A Wijaya, RS Wahono JURNAL TEKNOLOGI 79 (7-2), 45-50		2017
Artikel: Aspek dan Kriteria Penilaian Media Pembelajaran RS Wahono	2	2016
Langkah Mudah Membuat Multimedia Pembelajaran RS Wahono	8	2016
Penanganan Fitur Kontinyu dengan Feature Discretization Berbasis Expectation Maximization Clustering untuk Klasifikasi Spam Email Menggunakan Algoritma ID3 RS Wahono, C Supriyanto Journal of Intelligent Systems 1 (2), 148-155		2015
Integrasi Metode Information Gain Untuk Seleksi Fitur dan Adaboost Untuk Mengurangi Bias Pada Analisis Sentimen Review Restoran Menggunakan Algoritma Naï...	3	2015

Dikutip oleh   LIHAT SEMUA

	Semua	Sejak 2013
Kutipan	1022	688
indeks-h	15	11
indeks-i10	19	12

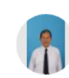


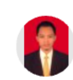
Pengarang bersama   LIHAT SEMUA

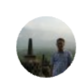

Behrouz H. Far
Professor of Software Engineerin...


Sabrina Ahmad
Faculty of Information and Com...


Abdul Syukur
Dian Nuswantoro University


Vincent Suhartono
UDINUS


Aries Saifudin
Universitas Pamulang


Catur Supriyanto
Universitas Dian Nuswantoro

About Us

- About Us
- Contact Us

Ranking of scientists in Indonesian Institutions according to their Google Scholar Citations public profiles

RANK	NAME	INSTITUTION	H-INDEX	CITATIONS
1	**Suharyo Sumowidagdo**	Indonesian Institute of Sciences	91	37075
2	**Johannes V D Wirjawan**	Widya Mandala Surabaya Catholic University	61	12228
4	**L P Ligthart**	Universitas Indonesia; ITS	30	3734
5	**Suryadi Ismadji**	Widya Mandala Surabaya Catholic University	28	3236
6	**Hairiah Kurniatun**	Universitas Brawijaya; ICRAF World Agroforestry Centre CGIAR	28	2440
7	**Ferry Iskandar**	Institut Teknologi Bandung	27	3018
8	**Christopher Martius**	Center for International Forestry Research; Institut Pertanian Bogor	26	2500
9	**Manabu D Yamanaka**	BPPT Indonesia; Kobe University	25	2149
10	**Azyumardi Azra**	Universitas Islam Negeri Syarif Hidayatullah Jakarta	24	2241
11	**L Broto Sugeng Kardono**	Indonesian Institute of Sciences	24	1801
12	**Danny Hilman Natawidjaja**	Indonesian Institute of Sciences	22	2556
13	**Agus Sudaryanto 1**	Agency for the Assessment and Application of Technology Indonesia	21	1791
14	**Djoko Iskandar**	Institut Teknologi Bandung	21	1759

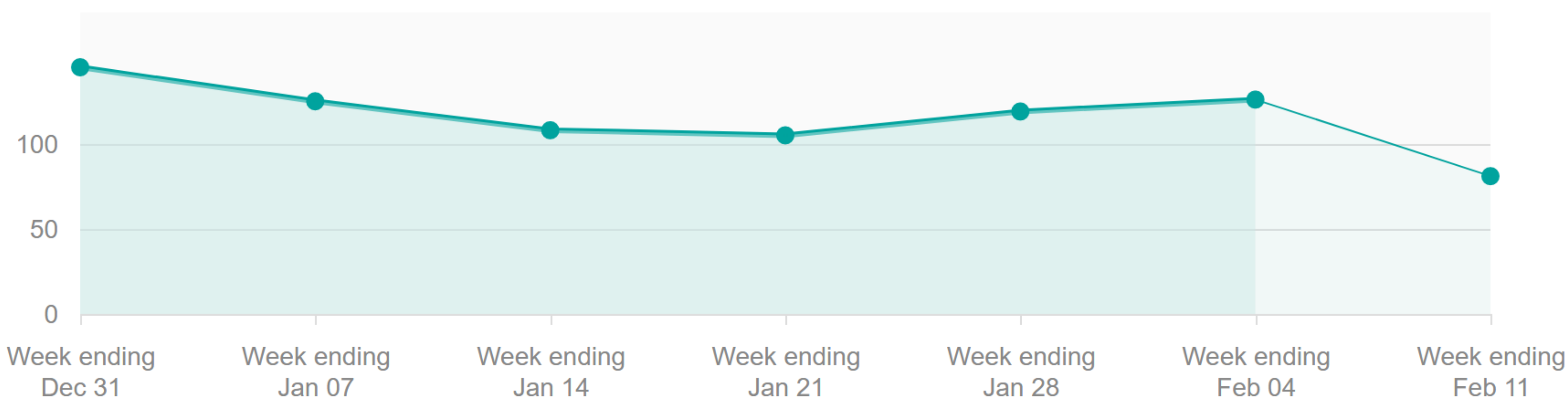
Romi Satria Wahono

9.98 · M.Eng, PhD

Overview | Contributions | Info | Stats | Reputation | Research Interests

Reads	Citations	Recommendations
12,941	121	16
Last week: 126	Last month: 2	Last week: 0

Reads



Author Profile

ROMI SATRIA WAHONO
Universitas Dian Nuswantoro
Computer Science

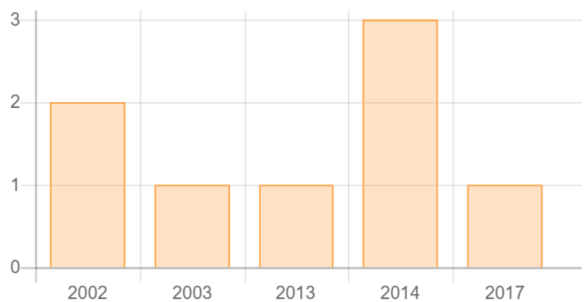
953 sinta
Rank in National

1 sinta
Rank in Affiliation

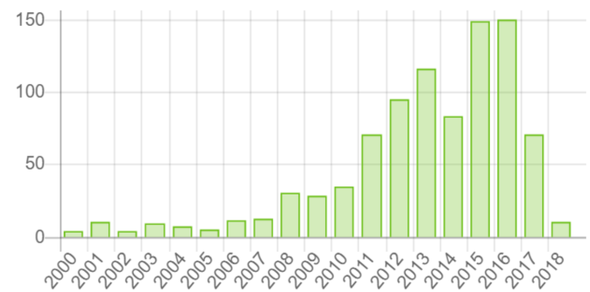
13.9 sinta Science and Technology Index
Score

Overview | Network

Documents per Year Scopus



Citations per Year Google Scholar



Research Output Scopus

6
Journal Articles
0
Book Chapters
2
Conference Papers

Score

	Articles	Citations	H-Index	i10-Index
Scopus	8	37	4	1
Google Scholar	159	1018	15	19

Top 5 Papers by Citations

Top 5 Papers by Citations	Citation
Sistem elearning berbasis model motivasi komunitas Jurnal Teknodik No 21 | vol: | issue : | 2007	28
A systematic literature review of software defect prediction: research trends, datasets, methods and frameworks Journal of Software Engineering 1 (1), 1-16 | vol: | issue : | 2015	26
Analyzing requirements engineering problems IECI Japan workshop, Japan, pages55-58 | vol: | issue : | 2003	25
Reasoning about rational agents | vol: | issue : | 2002	19
Combining particle swarm optimization based feature selection and bagging technique for software defect prediction International Journal of Software Engineering and Its Applications 7 (5�… | vol: | issue : | 2013	19

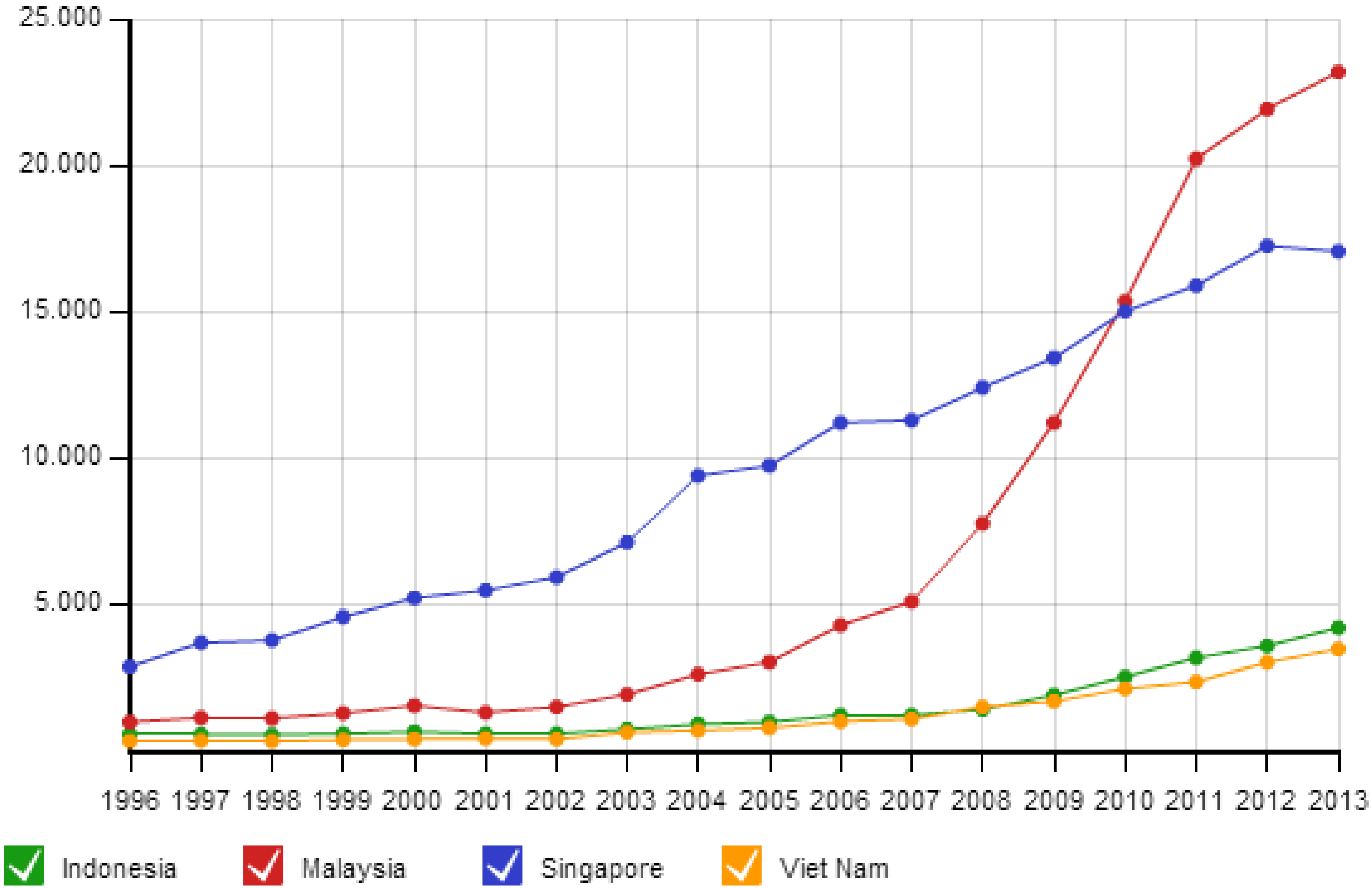
Rangking Publikasi Ilmiah (ScimagoJR.Com)

	Country	Documents	Citable documents	Citations	Self-Citations	Citations per Document	H index
1	United States	7.063.329	6.672.307	129.540.193	62.480.425	20,45	1.380
2	China	2.680.395	2.655.272	11.253.119	6.127.507	6,17	385
3	United Kingdom	1.918.650	1.763.766	31.393.290	7.513.112	18,29	851
4	Germany	1.782.920	1.704.566	25.848.738	6.852.785	16,16	740
5	Japan	1.776.473	1.734.289	20.347.377	6.073.934	12,11	635
6	France	1.283.370	1.229.376	17.870.597	4.151.730	15,60	681
7	Canada	993.461	946.493	15.696.168	3.050.504	18,50	658
8	Italy	959.688	909.701	12.719.572	2.976.533	15,26	588
9	Spain	759.811	715.452	8.688.942	2.212.008	13,89	476
10	India	750.777	716.232	4.528.302	1.585.248	7,99	301
11	Australia	683.585	643.028	9.338.061	2.016.394	16,73	514
12	Russian Federation	586.646	579.814	3.132.050	938.471	5,52	325
13	South Korea	578.625	566.953	4.640.390	1.067.252	10,55	333
14	Netherlands	547.634	519.258	10.050.413	1.701.502	21,25	576
15	Brazil	461.118	446.892	3.362.480	1.151.280	10,09	305
16	Taiwan	398.720	389.411	3.259.864	790.103	10,41	267
17	Switzerland	395.703	377.016	7.714.443	1.077.442	22,69	569
18	Sweden	375.891	361.569	6.810.427	1.104.677	20,11	511
19	Poland	346.611	339.712	2.441.439	652.956	8,25	302
20	Turkey	306.926	291.814	1.935.431	519.675	8,24	210
21	Belgium	299.077	285.735	4.696.153	701.283	18,16	454
22	Israel	224.674	215.590	3.663.004	530.340	17,78	414

Rangking Publikasi Ilmiah (ScimagoJR.Com)

49	Slovenia	50.565	49.471	403.209	83.402	9,53	153
50	Bulgaria	45.348	44.609	319.449	56.183	7,80	138
51	Nigeria	40.952	40.124	174.002	42.457	6,23	89
52	Tunisia	38.334	36.859	169.981	39.062	6,77	85
53	Colombia	35.890	34.768	228.686	36.843	10,61	133
54	Serbia	28.882	28.312	81.010	23.288	8,75	68
55	Morocco	27.253	26.175	157.219	29.432	7,11	99
56	Venezuela	27.138	26.445	204.691	29.729	8,42	130
57	Algeria	25.714	25.387	105.945	20.698	6,49	78
58	Belarus	24.801	24.466	122.850	24.438	5,08	106
59	Lithuania	24.755	24.434	151.748	37.377	8,61	109
60	Cuba	24.606	23.847	123.183	28.193	5,81	93
61	Indonesia	20.166	19.740	146.670	16.149	10,94	112
62	Jordan	19.847	19.507	107.550	15.257	7,24	82
63	Bangladesh	19.481	19.037	115.329	22.662	8,37	97
64	Estonia	19.141	18.774	204.306	38.547	13,58	130
65	United Arab Emirates	19.051	18.331	100.247	11.207	7,56	87
66	Kenya	16.727	16.044	206.886	34.874	15,09	131
67	Viet Nam	16.474	16.116	125.927	18.500	11,79	107
68	Kuwait	13.775	13.425	93.290	12.879	7,67	83
69	Lebanon	13.677	12.847	97.316	10.182	9,70	97
70	Philippines	13.163	12.796	141.070	15.727	13,38	116
71	Puerto Rico	11.209	10.955	150.252	11.819	15,34	129

Statistik Jumlah Publikasi (ScimagoJR.Com)



Kondisi Jurnal Ilmiah Indonesia

- Dipublikasikan dalam jumlah terbatas
- Tidak dilanggan oleh perpustakaan (Indonesia maupun internasional)
- Ditulis dalam Bahasa Indonesia
- Tidak digunakan pengajar di Indonesia sebagai materi pengajaran di kampus
- Tidak begitu dipedulikan oleh dunia akademik
- Sangat sedikit yang terindeks oleh lembaga pengindeks jurnal (11 jurnal terindeks scopus)

Jurnal Indonesia Terindeks Scopus

No	Journals	Publisher	SJR
1	Nutrition Bulletin	PAGI	0.365
2	Bulletin of Chemical Reaction Engineering and Catalysis	UNDIP	0.303
3	Acta medica Indonesiana	ISIM	0.250
4	Telkomnika	IAES	0.236
5	Indonesian Journal of Chemistry	UGM	0.171
6	International Journal on Electrical Engineering and Informatics	ITB	0.168
7	Critical Care and Shock	ISCCM	0.141
8	Journal of Engineering and Technological Sciences	ITB	0.139
9	International Journal of Power Electronics and Drive Systems	IAES	0.134
10	International Journal of Technology	UI	0.123
11	Biotropia	BIOTROP	0.112
12	Journal of ICT Research and Applications	ITB	0.107
13	Gadjah Mada International Journal of Business	UGM	0.100
14	Agrivita	UB	-
15	ITB Journal of Engineering Science	ITB	-
16	Journal of Mathematical and Fundamental Sciences	ITB	-

Jurnal Indonesia Terakreditasi Dikti

Bidang Ilmu	NO	Nama Terbitan Berkala Ilmiah	ISSN	Penerbit	Rekomendasi	Peringkat
PERTANIAN	1	Jurnal Agronomi Indonesia	2085-2916	Perhimpunan Agronomi Indonesia (PERAGI) dan Departemen Agronomi dan Hortikultura, Fak. Pertanian, Institut Pertanian Bogor	Terakreditasi	A
	2.	Agritech (Jurnal Teknologi Pertanian)	0216-0455	Fak. Teknologi Pertanian UGM Perhimpunan Ahli Teknologi Pangan Cabang Yogyakarta & Perhimpunan Teknik Pertanian Cabang Yogyakarta	Terakreditasi	B
	3.	Jurnal Entomologi Indonesia	1827-7722	Perhimpunan Entomologi Indonesia Bogor	Terakreditasi	B
REKAYASA	1.	Makara Seri Teknologi	1693-6698	Universitas Indonesia	Terakreditasi	B
	2.	Jurnal Teknik Sipil	0853-2982	Fakultas Teknik Sipil dan Lingkungan Institut Teknologi Bandung (ITB)	Terakreditasi	B
	3.	ITB Journal of Information and Communication Technology	1978-3086	LPPM ITB berkerja sama dengan Persatuan Insinyur Indonesia (PII) Bandung	Terakreditasi	B
	4.	Jurnal Teknologi Industri Pertanian	0216-3160	Asosiasi Agroindustri Indonesia bekerja sama dengan Departemen Teknologi Industri Pertanian, FATETA IPB	Terakreditasi	B
	5.	Jurnal Teknik Industri	1411-2485	Jur. Teknik Industri, Fak. Teknologi Industri Univ. Kristen Petra Surabaya	Terakreditasi	B

Mengapa Indonesia Sedikit Publikasi?

- Budaya Indonesia adalah lisan dan bukan tulisan
- Budaya akademik di Indonesia baru mulai untuk mengajar, dan bukan untuk meneliti
- Rendahnya minat penelitian dan mempublikasikan hasil penelitian
- Kurangnya penghargaan dan insentif dari universitas
- Kurang mengerti bagaimana cara menulis paper untuk jurnal dan prosedur pengirimannya
- Tidak memahami metodologi penelitian dengan baik

4.2 Sitasi dan Penulisan Referensi

Pengambilan Sitasi (*Citation*)

- Citation atau sitasi adalah penggunaan referensi di teks atau naskah tulisan ilmiah
- Penulisan sitasi tergantung dari standard (style) penulisan referensi yang digunakan
- Usahakan sitasi dan referensi penelitian hanya diambil dari journal ilmiah yang terindeks oleh SCOPUS atau ISI
- Pengambilan sitasi dari literature yang tidak berkualitas akan mempengaruhi kualitas penelitian kita

Tahapan Pengambilan Citation

1. Cari dan baca referensi penelitian yang berhubungan dengan masalah penelitian
2. Ambil catatan dari apa yang kita baca. Ikuti aturan umum pengambilan catatan (*citation*)
3. Atur susunan tinjauan pustaka (referensi) dari catatan yang kita ambil dengan baik. Ikuti aturan umum penulisan referensi

Jenis *Citation*

1. Kutipan (Quotation): Kata-kata yang diambil persis sama dengan apa yang dituliskan (tanpa perubahan). Ditulis dalam tanda kutip
2. Paraphrase: Menyusun kembali pemikiran penulis dan mengungkapkannya dengan kata-kata sendiri
3. Ringkasan: Sari dari suatu tulisan
4. Evaluasi: Interpretasi dalam bentuk komentar, baik setuju atau tidak dengan menyebutkan alasannya

(Beast & Kohn, 1998)

Aturan *Citation*

- Kutipan yang diambil dari buku dan jurnal diperbolehkan, selama tidak melebihi 250 kata untuk buku teks dan 5% panjang tulisan untuk artikel jurnal
- Menyebutkan sumber dari mana kutipan dan paraphrase diperoleh
- Menyalin dari artikel berupa grafik dan bagan memerlukan izin dari pembuatnya

**American Psychological Association (APA)*

Konsep Dasar Penulisan

- Kutipan itu tidak berarti bahwa satu paragraf kita copy-paste. Praktek seperti ini tetap disebut plagiarism meskipun referensi disebutkan
- Kutipan hanya untuk hal penting (hasil penelitian, teori, data, model, definisi) dalam paper
- Segala kalimat yang tidak merujuk atau menunjuk ke kutipan, berarti adalah tulisan karya sendiri
- Daftar referensi bukan daftar bacaan, tapi daftar rujukan atau kutipan (dibaca langsung, bukan dari penulis ketiga)

Mensitasi Sitasi Orang Lain

- Mensitasi (mengutip) hasil rangkuman dan kutipan yang dilakukan orang lain di buku atau papernya
- Definisi logika fuzzy menurut Lotfie Zadeh dalam Suyanto (Suyanto, 2009) adalah:

  blablabla
- Jangan terlalu banyak dilakukan kecuali dalam keadaan:
  - Kita tidak bisa mengakses publikasi asli
  - Bahasa asli publikasi bukan bahasa inggris (sulit dipahami)
- Terlalu banyak melakukan akan membuat orang lain menyebut kita "peneliti malas"

Standard Penulisan Referensi

1. APA Style
2. Harvard Style
3. Vancouver Style
4. IEEE Style
5. ISO Style

Menggunakan fitur references pada word processor akan mempermudah pengaturan dan pengelolaan referensi pada dokumen

Penulisan Citation (APA)

- Teks (Nama Keluarga Penulis, Tahun Terbit)
  - Model motivasi komunitas efektif diterapkan pada implementasi eLearning publik (Wahono, 2007) (satu penulis)
  - Model komunikasi multiagent system mengacu pada konsep game theory (Wahono & Far, 2003) (dua penulis)
  - Model komunikasi multiagent system mengacu pada konsep game theory (Wahono et al., 2003) (lebih dari 6 penulis)
- Teks (Tahun Terbit)
  - Penelitian yang dilakukan Wahono menunjukkan bahwa model motivasi komunitas efektif diterapkan pada implementasi eLearning publik (2007)
  - Penelitian yang dilakukan Wahono dan Far menunjukkan bahwa model komunikasi multiagent system mengacu pada konsep game theory (2003)

Penulisan Referensi (APA) -1-

JURNAL DAN KARYA ILMIAH

- Wahono, R.S. (2007, Agustus). Sistem eLearning Berbasis Model Motivasi Komunitas, Jurnal Teknodik , No. 21 Vol. XI, pp. 60-80. (satu penulis)
- Wahono, R.S. & Far, B.H (2003, August). Cognitive-Decision-Making Issues for Software Agents, Kluwer journal of Brain and Mind , Vol. 4 No. 2, pp.239-252. (dua penulis)
- Wahono, R.S. et al. (2002, March). A Framework for Object Identification and Refinement Process, IEEE Transaction on Software Engineering, Vol. 12 No 4, pp. 125-143. (lebih dari enam penulis)

Penulisan Referensi (APA) -2-

BUKU

- Wahono, R.S. (2004). Cepat Mahir Bahasa C, Jakarta: Elex Media Komputindo. (Satu penulis)
- Wahono, R.S. & Amri, M.C (2006). Migrasi Windows-Linux, Jakarta:IlmuKomputer.Com. (dua penulis)
- Wahono, R.S. et al. (2007). Panduan Pengembangan Multimedia Pembelajaran, Jakarta: Direktorat Pembinaan SMA, Ditjen Manajemen Pendidikan Dasar dan Menengah, Depdiknas. (lebih dari enam penulis)

Penulisan Referensi (APA) -3-

TESIS DAN DISERTASI

- Wahono, R.S. (1999). Distributed Knowledge Based System for Automatic Object-Oriented Software Design Development. B.Eng Dissertation, Saitama University, Saitama- Japan.

ARTIKEL DI INTERNET

- Wahono, R.S. (2008). Pengembangan Konten di Era Web 2.0. Diambil 5 Mei 2008, dari http://romisatriawahono.net/2008/04/21/pengembangan-konten-di-era-web-20/

4.3 Struktur dan Kiat Penulisan Tesis

Struktur Umum Penulisan Ilmiah

1. Introduction
2. Related Works
3. Research Method
4. Result and Analysis
5. Conclusion

1. Introduction
2. Literature Review
3. Experiment Settings
4. Experiment Results
5. Conclusion

1. Motivation
2. Related Works
3. Method
4. Results
5. Conclusion

4.3.1 Judul Penelitian

Judul Penelitian

- Judul penelitian sebaiknya singkat padat dan mewakili seluruh isi penelitian kita
- Maksimal hanya terdiri dari 8-12 kata
- Tidak ada singkatan
- Tidak menggunakan kata-kata redundant (*study on*, *research on*, dsb)
- Judul penelitian wajib memuat:
    1. Metode yang Diusulkan
    2. Tujuan Penelitan
    3. Obyek Penelitian

Judul Penelitian

- Hindari kata “berbasis”, harus lebih jelas sebenarnya tujuan apa, masalahnya apa, dan solusinya yg ditawarkan juga apa
  - Prediksi Produksi Padi dengan Menggunakan SVM berbasis PSO (X)
  - Pemilihan Parameter pada SVM dengan menggunakan PSO untuk Prediksi Produksi Padi (O)

Contoh Judul Penelitian

Metode **Tujuan** **Obyek**

- Penerapan Algoritma Semut untuk Pemilihan Arsitektur Jaringan pada Neural Network untuk Pengujian Software Metode Blackbox
- Penerapan Algoritma A* yang Diperbaiki untuk Pencarian Tempat Parkir Kosong di Mal dan Supermaket
- Penggabungan Forward Selection dan Backward Elimination untuk Pemilihan Fitur pada Prediksi Mahasiswa DO dengan menggunakan Algoritma C4.5

Penulis

- Tanpa disertai gelar akademik
- Hanya yang memberikan kontribusi signifikan yang berhak dicantumkan sebagai penulis
- Mencantumkan nama lembaga asal penulis
- Disertai alamat untuk korespondensi (alamat surat atau email penulis)

4.3.2 Abstrak Penelitian

Abstrak Penelitian

- Harus menggambarkan keseluruhan isi dari tulisan atau penelitian yang kita lakukan
- Abstrak diuraikan dengan bahasa lugas, langsung ke sasaran, dan harus memuat:
    1. Masalah penelitian
    2. Metode (plus pengembangan/perbaikan) yang kita gunakan untuk memecahkan masalah penelitian
    3. Hasil penelitian
- Abstrak dibuat dalam bentuk satu paragraf saja
- Kata kunci memuat kata-kata konseptual, dan jumlah sekitar 3-5 kata
- Pola pembuatan abstrak (*pro forma abstract*) ditulis oleh (Newman, 1994)

Pro Forma Abstract (Newman, 1994)

Existing <model-type> models are deficient in dealing with <properties> of <solution strategy>. An enhanced <model-type> is described, capable of providing more accurate analyses/predictions of <properties> in <solution strategy> designs. The model has been tested by comparing analyses/predictions with empirically measured values of <properties>

Pro Forma Abstract (Sample)

Existing GOMS models are deficient in dealing with the speed of use of workstation applications involving dynamic visual information and multi-party conversation. An enhanced GOMS model is described, capable of providing more accurate predictions of speed of use in such workstation application designs. The model has been tested by comparing predictions with empirically measured values of speed of use.

(John, 1990)

Pro Forma Abstract (Sample)

Pada proses XYZ ada beberapa masalah yang muncul yaitu masalah ABC, yang membuat proses XYZ tidak efisien. Masalah ABC akan diselesaikan dengan metode DEF, yang terbukti efisien digunakan untuk memecahkan masalah seperti pada proses XYZ. Hasil penelitian menunjukkan bahwa metode DEF berhasil meningkatkan efisiensi sebesar X% apabila diterapkan untuk memecahkan masalah ABC pada proses XYZ

4.3.3 Bab 1: Pendahuluan

Struktur Tesis – Bab I

Bab I Pendahuluan

1.1 Latar Belakang Masalah
1.2 Identifikasi Masalah (Research Problems)
1.3 Rumusan Masalah (Research Questions)
1.4 Tujuan Penelitian (Research Objectives)
1.5 Manfaat Penelitian
1.6 Korelasi RP – RQ - RO
1.7 Kontribusi Penelitian
1.8 Sistematika Penulisan

Latar belakang masalah

- Ikuti pola latar belakang masalah OMKKMasaSolTu, seperti yang ada di http://romisatriawahono.net/2012/06/18/kiat-menyusun-alur-latar-belakang-masalah-penelitian/
- Jangan meletakkan sitasi untuk referensi pada akhir paragraf *(Berndtsson, 2009)*
- Satu pernyataan yang mensitasi ke banyak reference harus dilakukan dengan hati-hati
- Kalimat dalam satu paragraf harus berisi satu pokok pikiran, dan bisa diletakkan di awal, tengah atau akhir paragraf
- Antar paragraf harus dibuat mengalir (ada kohesi), bersambungan, dengan alur cerita yang runut

Pokok Pikiran dalam Paragraf?

Decision Tree merupakan algoritma pengklasifikasian yang sering digunakan dan mempunyai struktur yang sederhana dan mudah untuk diinterpretasikan (Mantas & Abellán, 2014). Pohon yang terbentuk menyerupai pohon terbalik, dimana akar (root) berada di bagian paling atas dan daun (leaf) berada di bagian paling bawah. Decision Tree merupakan model klasifikasi yang berbentuk seperti pohon, dimana Decision Tree mudah untuk dimengerti meskipun oleh pengguna yang belum ahli sekalipun dan lebih efisien dalam menginduksi data (C. Sammut, 2011). Induksi di Decision Tree adalah salah satu teknik tertua dan yang paling tertua untuk model *learning discriminatory*, yang mana model tersebut telah dikembangkan secara mandiri di statistik dan di komunitas *machine learning*. Proses pembentukan *Decision Tree* dibagi menjadi 3 (T Warren Liao, 2007) yaitu, (1) pembentukan pohon (*tree*), (2) *pruning*, (3) mengekstrak aturan (*rule*) dari pohon keputusan yang terbentuk. *Decision Tree* baik digunakan untuk klasifikasi atau prediksi.

Pokok Pikiran dalam Paragraf?

Decision Tree telah diaplikasikan di bidang pengobatan (Setsirichok et al., 2012). Salah satu contohnya adalah penerapan C4.5 yang digunakan untuk mengklasifikasikan karakteristik darah sehingga dapat mengklasifikasikan 80 class kelainan thalassemia yang menyebar di Thailand. Ture, Tokatli, & Kurt juga menggunakan Decision Tree untuk memprediksi pasien kanker payudara (Ture, Tokatli, & Kurt, 2009).

Pokok Pikiran dalam Paragraf?

Pengujian dan perbaikan bug software merupakan fase pengembangan software yang paling mahal dan banyak memakan waktu. Lebih dari 50% usaha dan biaya pengembangan software digunakan untuk pengujian dan perbaikan bug (Fakhrahmad & Sami, 2009). Perbaikan bug akan semakin meningkat biayanya, apabila dilakukan di fase akhir pengembangan software. Strangio menyatakan bahwa biaya untuk memperbaiki bug akibat salah analisis kebutuhan setelah fase penerapan dapat mencapai 100 kali lipat. Biaya untuk memperbaiki bug pada tahap desain setelah pengiriman produk mencapai 60 kali lipat. Sedangkan biaya untuk memperbaiki bug pada tahap desain yang ditemukan oleh pelanggan adalah 20 kali lipat (Strangio, 2009).

Latar belakang masalah

- Masalah penelitian yang kita angkat harus dilandasi dengan publikasi paper yang kuat (usahakan dari paper journal ber-impact factor tinggi, tidak dari conference procedings)
- Harus dipahami bahwa tujuan latar belakang masalah adalah memberi argumentasi bahwa masalah penelitian yg diangkat adalah valid
- Tidak menggunakan kata "peneliti atau penulis", tapi membuat kalimat jadi pasif, contoh:
  - Peneliti akan mencoba memecahkan masalah tersebut dengan metode A (X)
  - Pada penelitian ini, metode A akan diterapkan untuk memecahkan masalah tersebut (O)

Contoh Sitasi Yang Salah

Recent work has reported that the importance of computers in industry cannot be overestimated. Several useful services (such as booking train tickets) rely on computers. However, the importance of using computers in our everyday life has been questioned. It has been argued that having too many computers in our everyday life causes security problems, since people cannot protect their computers from hackers and Internet viruses. The researchers are still debating these hot topics (Jones, 1993).

Contoh Sitasi Yang Sudah Diperbaiki

Recent work has reported that the importance of computers in industry cannot be overestimated. Several useful services (such as booking train tickets) rely on computers (Bridge, 2010). However, the importance of using computers in our everyday life has been questioned. It has been argued that having too many computers in our everyday life causes security problems (Jones, 1993), since people cannot protect their computers from hackers and Internet viruses. The researchers are still debating these hot topics. So, we need new finding and research results on these topics.

Contoh Sitasi ke Banyak Reference

Recent work has reported that the importance of computers in industry cannot be overestimated. Several useful services (such as booking train tickets) rely on computers. However, the importance of using computers in our everyday life has been questioned. It has been argued that having too many computers in our everyday life causes security problems, since people cannot protect their computers from hackers and Internet viruses. The researchers are still debating these hot topics (Jones, 1993) (Lessmann, 2007) (Zhang, 2009). So, we need new finding and research results on these topics.

Contoh Sitasi ke Banyak Reference

Various types of classification algorithms have been applied for software defect prediction, including logistic regression (Denaro 2000), decision trees (Khoshgoftaar & Seliya, 2002) (Khoshgoftaar & Gao, 2009), neural networks (Park et al. 2011) (Wang & Yu 2004) (Zheng 2010), naive bayes (Menzies et al. 2007). This research is focused and concerned with the third approach.

Alur Penulisan Latar Belakang Masalah*

- Technical Paper:
  - Judul: Chinese Grain Production Forecasting Method Based on Particle Swarm Optimization-based Support Vector Machine
  - Author: Sheng-Wei Fei, Yu-Bin Miao and Cheng-Liang Liu
  - Publications: Recent Patents on Engineering 2009, 3, 8-12
  - Download: **http://romisatriawahono.net/lecture/rm/paper/**
- Tugas Literature Review:
  1. Baca dan pahami paper di atas
  2. Tentukan latar belakang masalah, pernyataan masalah, pertanyaan penelitian, tujuan penelitian, existing methods, kontribusi penelitian dan hasil penelitian
  3. Rangkumkan dalam 7 slide

** http://romisatriawahono.net/2012/06/18/kiat-menyusun-alur-latar-belakang-masalah-penelitian/*

Kiat Menyusun Latar Belakang Masalah*

1. Latar belakang masalah penelitian harus menjawab semua pertanyaan MENGAPA (*WHY*) dari judul penelitian kita. Bila judul penelitian: Penerapan Particle Swarm Optimization untuk Pemilihan Parameter pada Support Vector Machine untuk Prediksi Produksi Padi, maka latar belakang masalah harus bisa menjawab pertanyaan:
    1. mengapa padi?
    2. mengapa prediksi produksi padi?
    3. mengapa support vector machine?
    4. mengapa particle swarm optimization?

**http://romisatriawahono.net/2012/06/18/kiat-menyusun-alur-latar-belakang-masalah-penelitian/*

Kiat Menyusun Latar Belakang Masalah

2. Pola alur paragraf mengikuti OMKKMasaSolTu
   1. obyek penelitian (O)
   2. metode-metode yang ada (M)
   3. kelebihan dan kelemahan metode yang ada (KK)
   4. masalah pada metode yang dipilih (Masa)
   5. solusi perbaikan metode (Sol)
   6. rangkuman tujuan penelitian (Tu)

Research Background

1. Padi adalah komoditas yang penting di china, karena tingkat produksinya tinggi (FAO Report, 2009) (1. mengapa padi?). Produksi padi perlu diprediksi dengan akurat, karena hasil prediksi yang akurat sangat penting untuk membuat kebijakan nasional (Traill, 2008) (2. mengapa prediksi produksi padi?).
   [1. obyek penelitian (O)]
2. Metode prediksi rentet waktu seperti Support Vector Machine (SVM) (Yongsheng, 2008), Neural Network (NN) (Tseng, 2007) dan Grey Model (GM) (Wu, 2007) diusulkan oleh banyak peneliti (Huifei, 2009) untuk prediksi produksi padi.
   [2. metode-metode yang ada]

Research Background

3. NN memiliki kelebihan pada prediksi nonlinear, kuat di parallel processing dan kemampuan untuk mentoleransi kesalahan, tapi memiliki kelemahan pada perlunya data training yang besar, over-fitting, lambatnya konvergensi, dan sifatnya yang local optimum (Rosario, 2007). GM punya kelebihan di tingginya akurasi prediksi meskipun menggunakan data yang sedikit, akan tetapi GM memiliki kelemahan pada prediksi data yang sifatnya naik turun secara fluktuatif seperti pada data produksi padi (Wu, 2007). [3. kelebihan dan kelemahan metode yang ada]

Research Background

4. SVM dapat memecahkan masalah NN dan GM, yaitu over-fitting, lambatnya konvergensi, dan sedikitnya data training (Vapnik, 2005), yang mana ini tepat untuk karakteristik data produksi padi pada penelitian ini (3. mengapa support vector machine?). Tetapi SVM memiliki kelemahan pada sulitnya pemilihan parameter SVM yang optimal (Coussement, 2008).
[4. masalah pada metode yang dipilih]
5. Particle Swarm Optimization (PSO) adalah metode optimisasi yang terbukti efektif digunakan untuk memecahkan masalah optimisasi multidimensi dan multiparameter pada pembelajaran pada machine learning seperti di NN, SVM, dan classifier lain (Brits, 2009) (4. mengapa particle swarm optimization?).
[5. solusi perbaikan metode]
6. Pada penelitian ini PSO akan diterapkan untuk pemilihan parameter SVM yang sesuai dan optimal, sehingga hasil prediksi lebih akurat. [6. rangkuman tujuan penelitian]

Masalah Penelitian (*Research Problem*)

- Harus merangkumkan suatu masalah penelitian dari uraian pada latar belakang masalah
- Harus bahasa masalah
- Menemukan masalah bisa dari *future work* peneliti lain yg ada di paper technical, biasanya diletakkan di dalam conclusion
- Masalah juga kadang bisa ditemukan dari paper review, khususnya yang membahas tentang problems atau challenge pada topic penelitian itu

Masalah Penelitian (*Research Problem*)

SVM dapat memecahkan masalah NN dan GM, yaitu ‘over-fitting’, lambatnya konvergensi, dan sedikitnya data training, akan tetapi SVM memiliki kelemahan pada sulitnya pemilihan parameter SVM yang optimal sehingga menyebabkan tingkat akurasi prediksi menjadi rendah

Rumusan Masalah (*Research Question*)

- Pertanyaan penelitian: how, how does, what .. But not "how to"
- Pertanyaan penelitian menggantikan hipotesis
- Gunakan kalimat tanya seperti bagaimana, seberapa efisien/akurat/cepat, dsb
- Pertanyaan pada rumusan masalah itu, akan dijawab oleh eksperimen penelitian kita, dan dirangkumkan secara lugas, jelas di bagian kesimpulan
- Jumlah eksperimen dan hasil yang dilakukan (Bab Hasil dan Pembahasan), ditentukan oleh jumlah research question (RQ) pada penelitian kita
- Uraikan dalam bentuk point-point apabila rumusan masalah lebih dari satu sehingga mudah dipahami

Rumusan Masalah (*Research Question*)

Seberapa tinggi akurasi metode SVM apabila PSO diterapkan pada proses pemilihan parameter yang optimal?

Tujuan Penelitian (*Research Objective*)

- Tujuan pada hakekatnya adalah judul, yang diuraikan dengan lebih detil atau spesifik
- Harus memuat metode dan tujuan beserta pengukurannya (sinkron dengan masalah)
- Uraikan dalam bentuk point-point apabila tujuan lebih dari satu sehingga mudah dipahami

Tujuan Penelitian (*Research Objective*)

Menerapkan PSO untuk pemilihan parameter yang sesuai (C, gamma dan epsilon) pada Support Vector Machine (SVM), sehingga hasil prediksinya lebih akurat

Manfaat Penelitian

- Hal baik yg datang setelah tujuan penelitian tercapai, baik dari sisi teoritis maupun organisasi
- Manfaat bukan mengulang-ulang tujuan
- Uraikan dalam bentuk point-point sehingga mudah dipahami

Korelasi RP-RQ-RO

RP	RQ	RO
SVM dapat memecahkan masalah 'over-fitting', lambatnya konvergensi, dan sedikitnya data training, akan tetapi memiliki kelemahan pada sulitnya pemilihan parameter SVM yang sesuai	Seberapa meningkat akurasi metode SVM apabila PSO diterapkan pada proses pemilihan parameter?	Menerapkan PSO untuk pemilihan parameter yang sesuai pada SVM (C, lambda dan epsilon) , sehingga hasil prediksinya lebih akurat

Pola Alur Alternatif (Pokok Pikiran Paragraf) untuk Penelitian Perbaikan Algoritma

1. Definisi Algoritma
2. Kelebihan Algoritma
3. Kelemahan Algoritma
4. Metode-Metode yang ada untuk Memecahkan Kelemahan Algoritma
5. Masalah pada Metode-Metode yang ada (Research Gap → Masalah Penelitian)
6. Solusi pemecahan masalah pada metode-metode yang ada

4.3.4 Bab 2: Literature Review

Struktur Tesis – Bab II

Bab II Landasan Teori

Tradisional Literature Review

2.1 Tinjauan Studi (Related Research)

(uraikan minimal 3 penelitian lain yang berhubungan (masalah-metode-hasil), serta tunjukkan bedanya dengan penelitian kita)

2.2 Tinjauan Pustaka (Landasan Teori)

2.2.1 Obyek Penelitian

2.2.2-2.2.* Landasan Teori Tentang Metode, Tahapan Algoritma dan Contoh Penerapannya

2.3 Kerangka Pemikiran

(gambar kerangka pemikiran beserta penjelasannya)

Struktur Tesis – Bab II

Bab II Landasan Teori

Systematic Literature Review

2.1 Introduction
2.2 Review Method
    2.2.1 Research Questions
    2.2.2 Search Strategy
    2.2.3 Study Selection
    2.2.4 Data Extraction
    2.2.5 Study Quality Assessment
    2.2.6 Data Synthesis
    2.2.7 Threats to Validity
2.3 Results and Analysis
    2.3.1 (RQ1 Results)
    2.3.n (RQn Results)
2.4 Summary

(Direkomendasikan untuk Menggunakan Systematic Literature Review (SLR), Lihat Slide Struktur Penulisan Tesis untuk Memahami Teknik Pembuatan SLR)

Tinjauan Studi

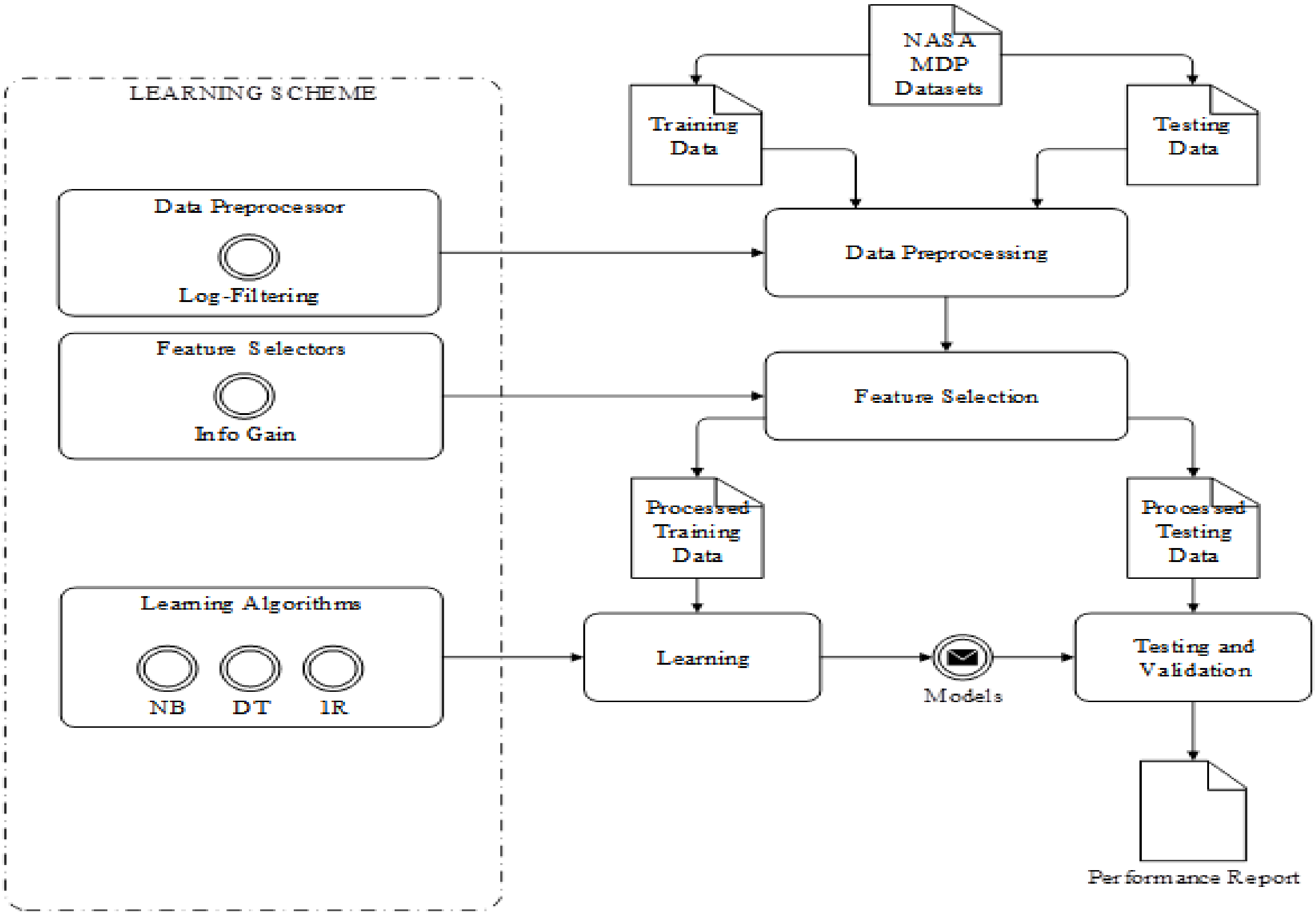
- Memuat penelitian yang benar-benar terkait, dalam aspek, metode di paper tersebut kita kembangkan
- Uraikan dengan format masalah-metode-hasil, tidak perlu ke sana sini
- Objek penelitiannya dekat dengan penelitian kita lebih baik
- Dipilih dari sisi kebaruan, kedekatan, dan memang kita memperbaiki metode yang dikembangkan oleh peneliti tersebut
- Setelah tinjauan studi ditulis, buat rangkuman dalam bentuk tabel state-of-the-art yang berisi: nama peneliti, tahun, masalah, metode dan hasil
- Akhiri subbab tinjauan studi dengan menjelaskan perbedaan dan kelebihan penelitian kita dengan penelitian di tinjauan studi tsb

State-of-the-Art Frameworks

Three frameworks have been highly cited and influential in software defect prediction field

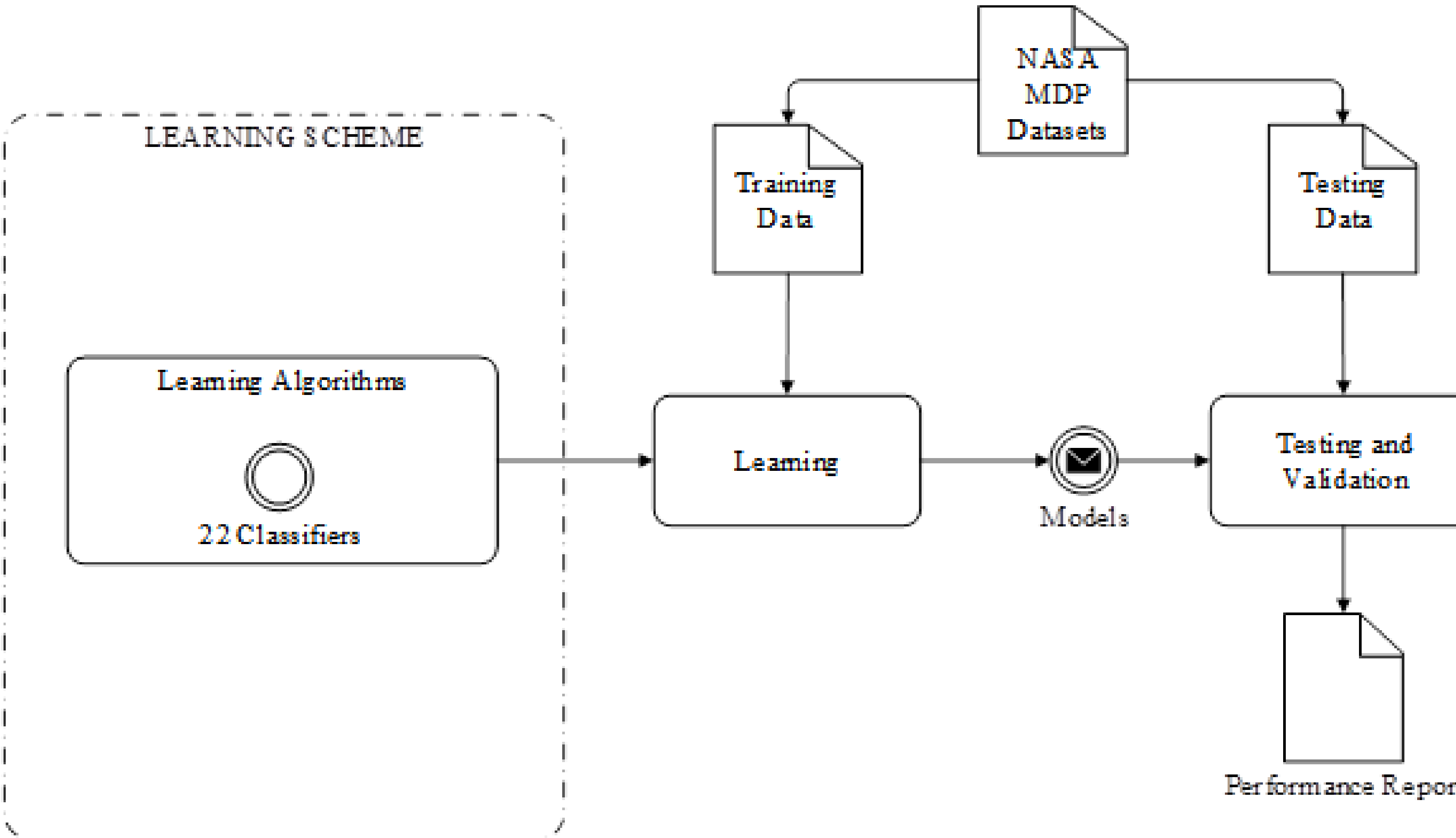
Menzies Framework	**Lessmann** Framework	**Song** Framework
(Menzies et al. 2007)	(Lessmann et al. 2008)	(Song et al. 2011)

Menzies Framework (Menzies et al. 2007)



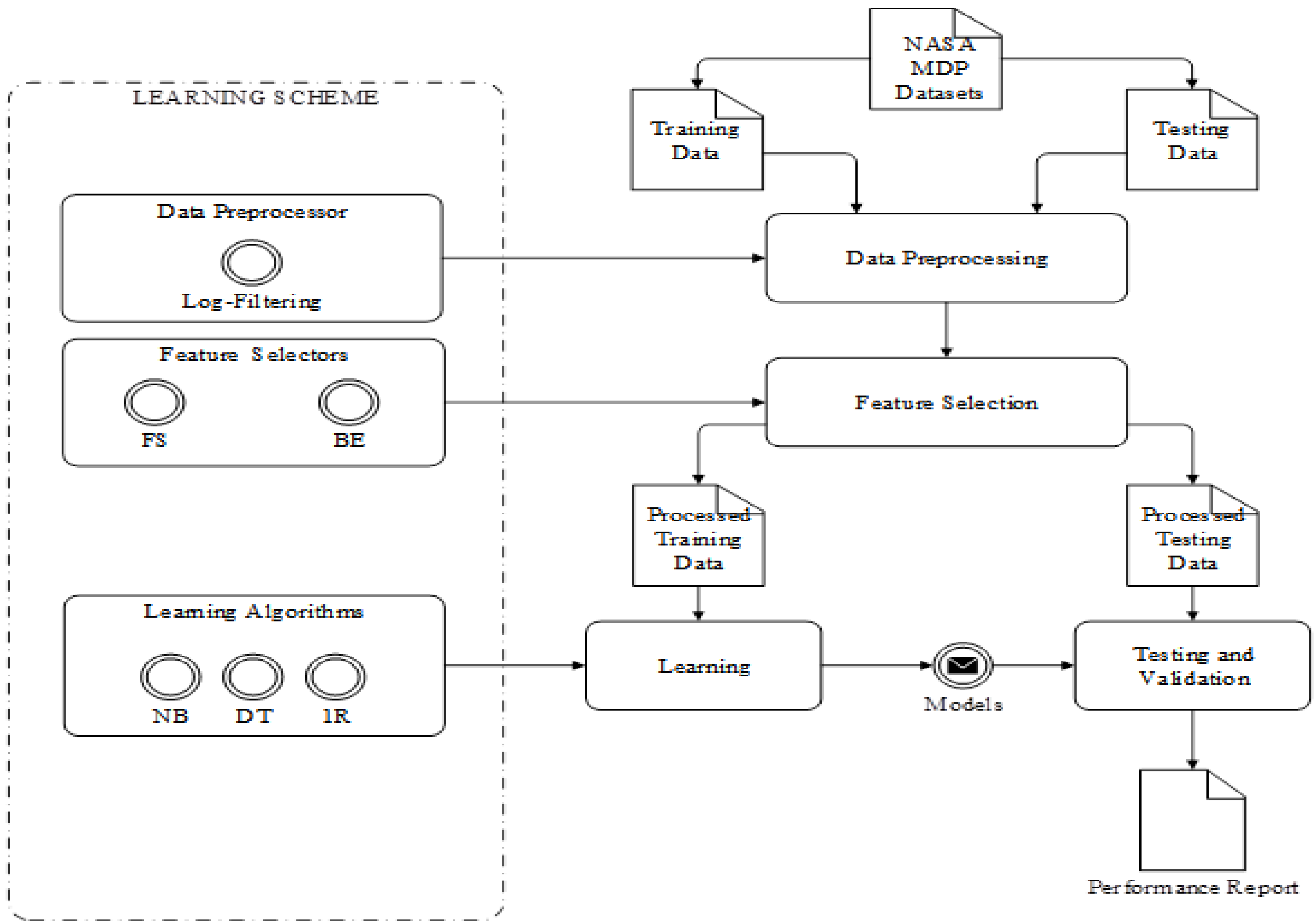
Framework	Dataset	Data Preprocessor	Feature Selectors	Meta-learning	Classifiers	Parameter Selectors	Validation Methods	Evaluation Methods
(Menzies et al. 2007)	NASA MDP	**Log Filtering**	**Info Gain**	**-**	**3 algorithms (DT, 1R, NB)**	**-**	10-Fold X Validation	ROC Curve (AUC)

Lessmann Framework (Lessmann et al. 2008)



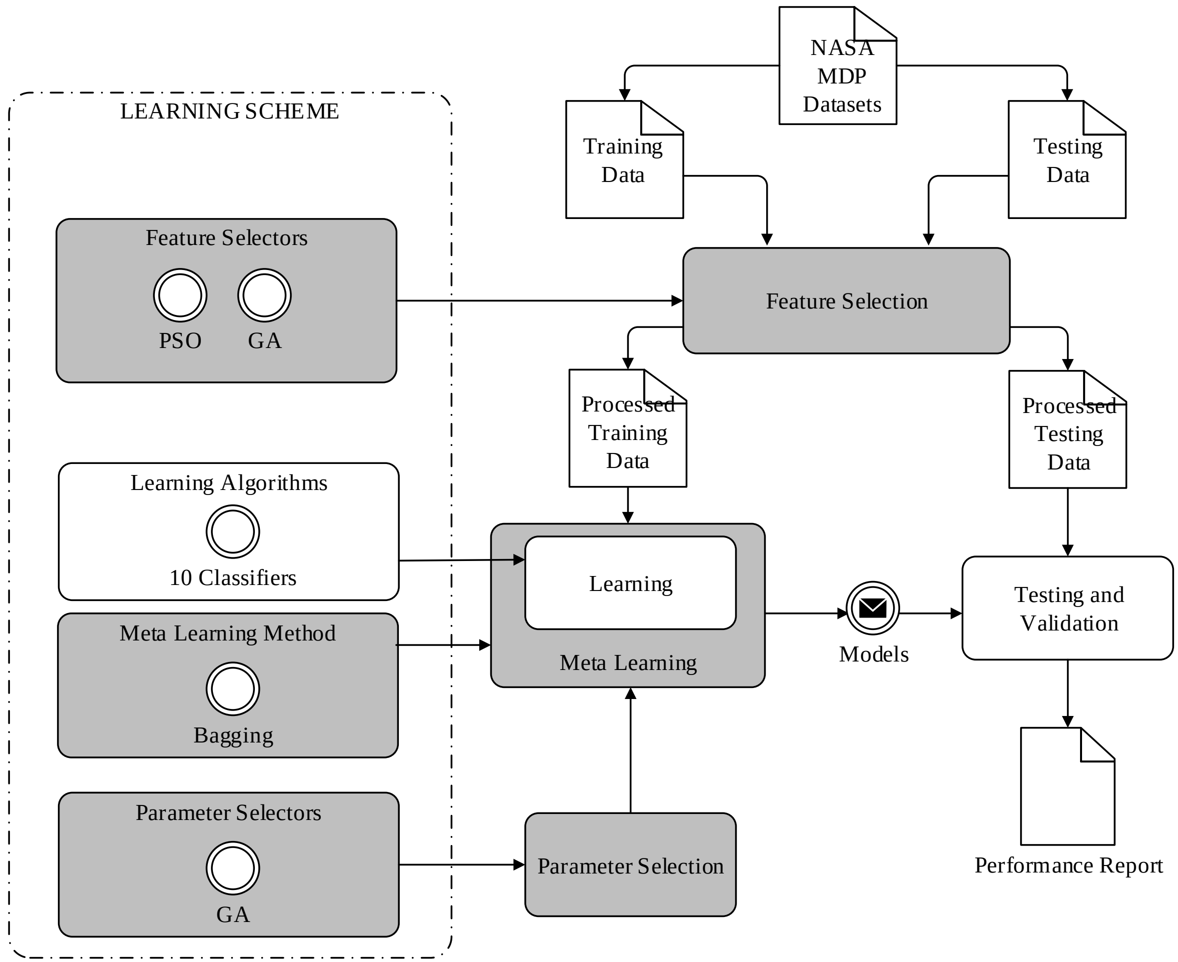
Framework	Dataset	Data Preprocessor	Feature Selectors	Meta-learning	Classifiers	Parameter Selectors	Validation Methods	Evaluation Methods
(Lessman et al. 2008)	NASA MDP	-	-	-	**22 algorithms**	-	10-Fold X Validation	ROC Curve (AUC)

Song Framework (Song et al. 2011)



Framework	Dataset	Data Preprocessor	Feature Selectors	Meta-learning	Classifiers	Parameter Selectors	Validation Methods	Evaluation Methods
(Song et al. 2011)	NASA MDP	**Log Filtering**	**FS, BE**	-	**3 algorithms (DT, 1R, NB)**	-	10-Fold X Validation	ROC Curve (AUC)

Proposed Framework



Framework	Dataset	Data Preprocessor	Feature Selectors	Meta-Learning	Classifiers	Parameter Selectors	Validation Methods	Evaluation Methods
(Menzies et al. 2007)	NASA MDP	Log Filtering	Info Gain		3 algorithm (DT, 1R, NB)	-	10-Fold X Validation	ROC Curve (AUC)
(Lessman et al. 2008)	NASA MDP	-	-		22 algorithm	-	10-Fold X Validation	ROC Curve (AUC)
(Song et al. 2011)	NASA MDP	Log Filtering	FS, BE		3 algorithm (DT, 1R, NB)	-	10-Fold X Validation	ROC Curve (AUC)
Proposed Framework	NASA MDP	-	**PSO, GA**	**Bagging**	**10 algorithms**	**GA**	10-Fold X Validation	ROC Curve (AUC)

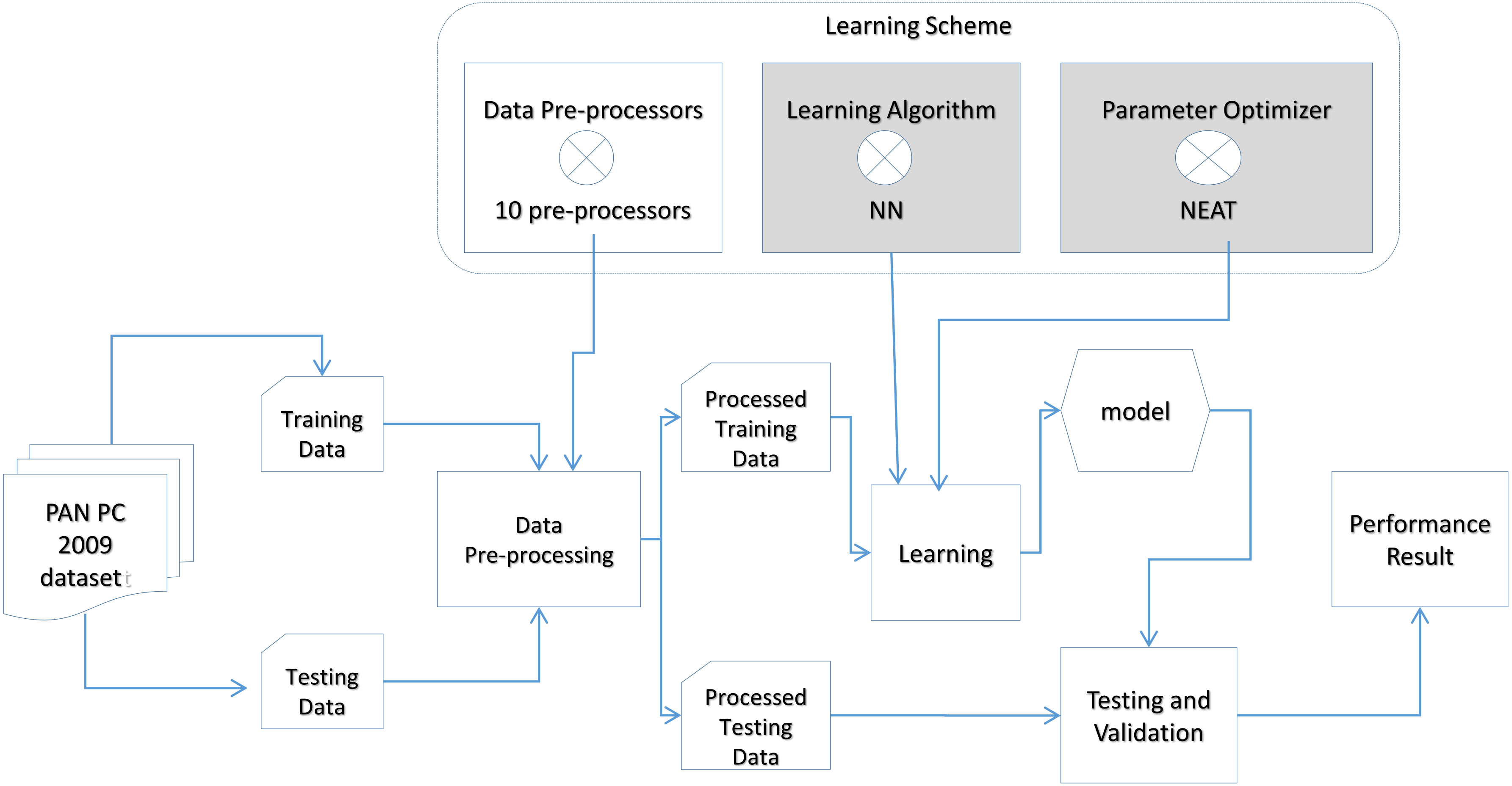
Contoh State-of-the-Art Methods

Model	Masalah Penelitian	Dataset	Pengukuran	Hasil
Jiang et al. (2007)	SPM memiliki tingginya penerimaan informasi warna kulit yang benar, akan tetapi tidak dapat menerima pada informasi fitur tekstur	Deteksi Kulit: Simulasi BrainWeb dan IBTD dataset	Akurasi segmentasi yang optimal (FPR and TPR)	false positive rate (FPR) = 6,2% true positive rate (TPR) = 92,97%
Wighton et al. (2011)	LDA tidak cukup akurat untuk mengidentifikasi dalam ukuran kernel yang berbeda	Deteksi Kulit: Simulasi BrainWeb dan IBTD dataset	Sensitivitas dan Spesifisitas (TN dan TP)	True Negative = 91% True Positive= 93%
Kawulok et al. (2013)	Performa dari SPM secara signifikan dapat mengalami "kebocoran" karena transisi halus antara kulit dan non-kulit	Deteksi Kulit: IBTD dataset	Analisa spasial yang baik (δ_{fp} + δ_{fn})	Detection Rate (DR) =94% false positive rate (FPR) = 34% false negative rate (FNR) = 6.13%
Muryan (2014)	SPM sulit untuk mendeteksi informasi fitur tekstur pada warna kulit dan kondisi pencahayaan Algoritma LDA sulit mengekstrak pada kernel yang berbeda	Deteksi Kulit: IBTD dataset	Analisa spasial yang baik (δ_{fp} + δ_{fn})	?

Contoh State-of-the-Art Methods

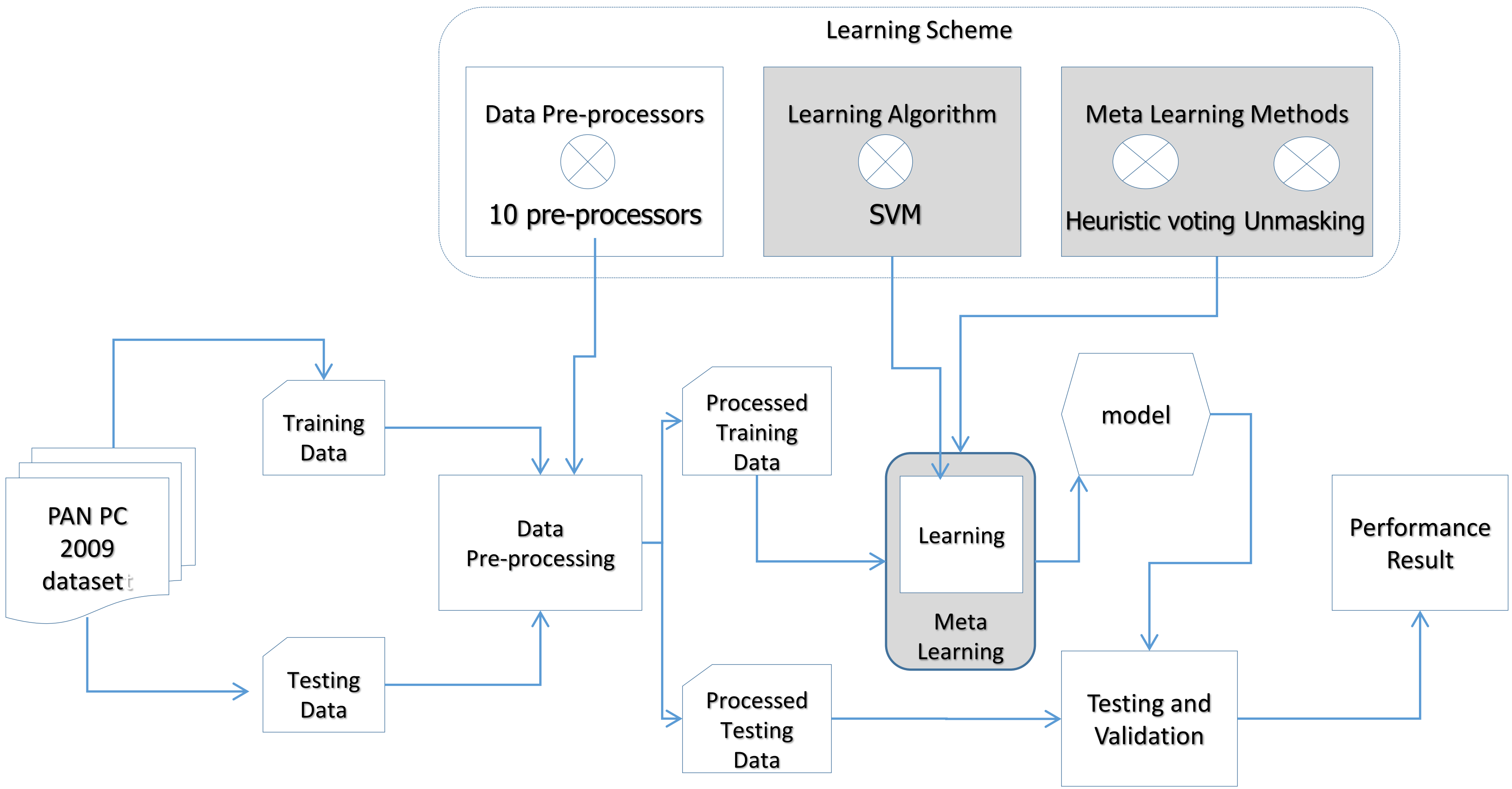
Models	Dataset	Data Preprocessor	Classifiers	Parameter Optimization	Result
(Curran, 2010)	PAN PC 2009	10 preprocessors	NN	NEAT Method	Accuracy: 60%
(Stein et al., 2011)	PAN PC 2009	10 preprocessors	SVM	-	P: 0.82 R: 0.73 F: 0.77
(Seaward & Matwin, 2009)	PAN PC 2009	Kolmogorov complexity measure	SVM	-	P: 0.521 R: 0.671 F: 0.585
			NN	-	P: 0.548 R: 0.670 F: 0.603

Curran's Model (Curran, 2010)



Model	Dataset	Data Pre-processor	Classifiers	Parameter Optimizer	Feature Creator	Meta Learning	Validation method	Evaluation Method
(Curran, 2010)	PAN PC 2009	10 Pre-processors	NN	NEAT	-	-	Fitness function	Accuracy

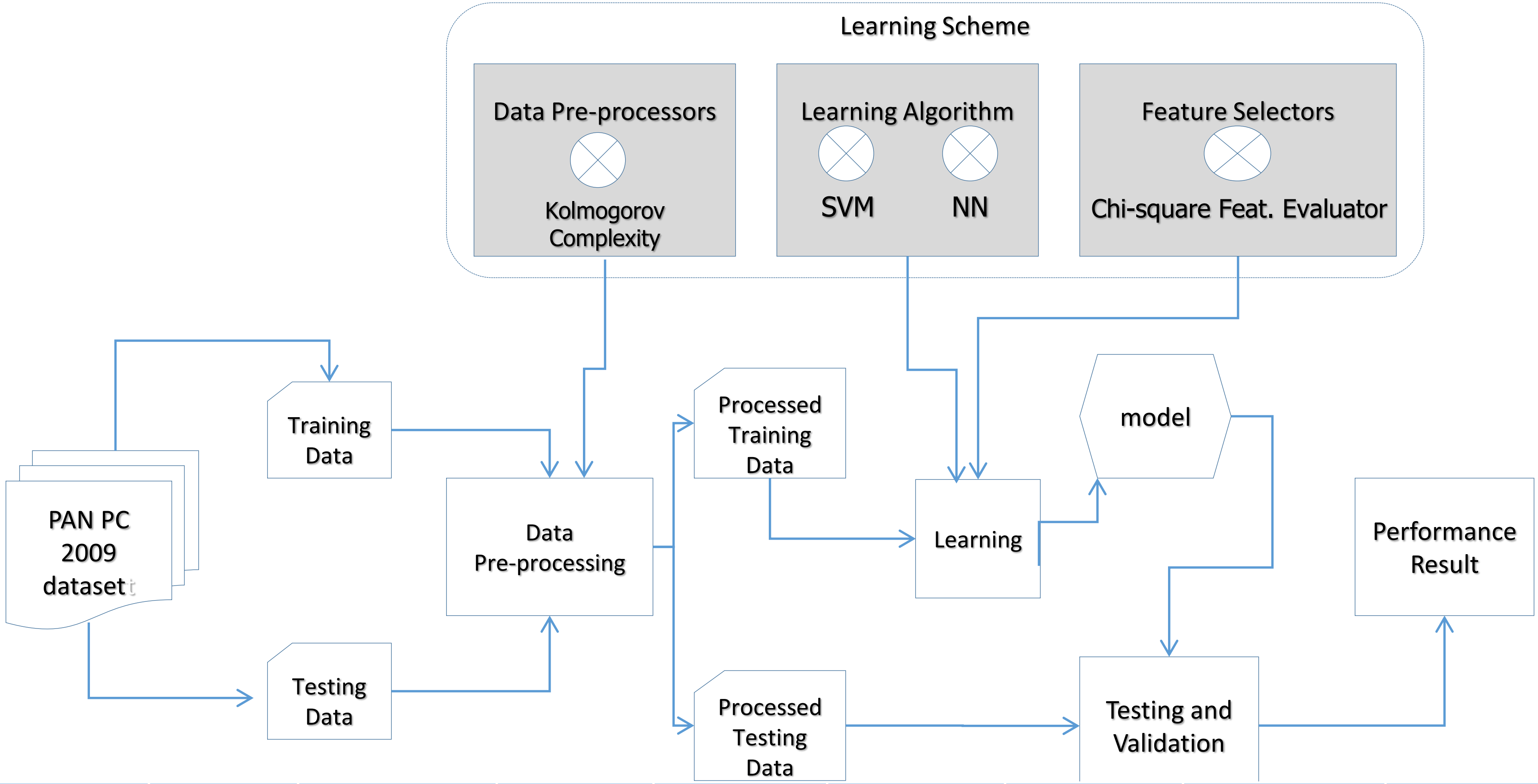
Stein *et al.*'s Model (Stein et al., 2011)



Model	Dataset	Data Pre-processor	Classifiers	Parameter Optimizer	Feature Creator	Meta Learning	Validation method	Evaluation Method
(Stein et al., 2011)	PAN PC 2009	10 Pre-processors	SVM	-	-	Heuristic Voting , Unmasking	5-fold X validation	Recall Precision F-measure

Seaward & Matwin's Model

(Seaward & Matwin, 2009)



Model	Dataset	Data Pre-processor	Classifiers	Parameter Optimizer	Feature Creator	Meta Learning	Validation method	Evaluation Method
(Seaward & Matwin, 2009)	PAN PC 2009	Kolmogorov Complexity	SVM, NN	-	FS: Chi-square Feat. Evaluator	-	-	Recall Precision F-measure

Masalah – SotA Methods – Proposed Method

Research Problems	State-of-the-Art Methods	Proposed Method (Contribution)	Evaluation Methods
Masalah 1	Metode 1.1	Metode PM1 (Average Gain)	Accuracy, Precision Recall, F Measure, G Mean (Friedman Test)
	Metode 1.2		
	Metode 1.3		
	Metode 1.4		
Masalah 2	Metode 2.1	Metode PM2 (Threshold + Cost Complexity Prunning)	
	Metode 2.2		
	Metode 2.3		
	Metode 2.4		
Masalah 1 + Masalah 2		Integrasi PM1 dan PM2	

Tinjauan Pustaka

- Memuat apa yang ada di judul
  - Contoh: prediksi produksi padi dengan SVM berbasis PSO
  - Isi tinjauan pustaka: SVM, PSO, Produksi Padi
- Penjelasan harus lengkap, tuntas, dan merangkumkan dari banyak sumber, bukan memindahkan isi satu buku atau publikasi lain ke tesis kita
- Algoritma harus berisi tahapan, formula dan contoh penerapannya

Gambar dan Tabel

- Caption untuk Gambar di bawah, sedangkan untuk Tabel di atas
- Tidak ada dalam kalimat yang menyatakan “gambar sebagai berikut…”, tapi yang benar adalah “Gambar 2.1 menjelaskan tentang …”
- Semua gambar dan tabel harus dinarasikan, harus dideskripsikan dan dijelaskan maksudnya apa
- Penjelasan kalimat, misalnya Gambar 2.1 atau Tabel 3.4, G dan T nya harus kapital, mengikuti caption dari gambar dan tabel

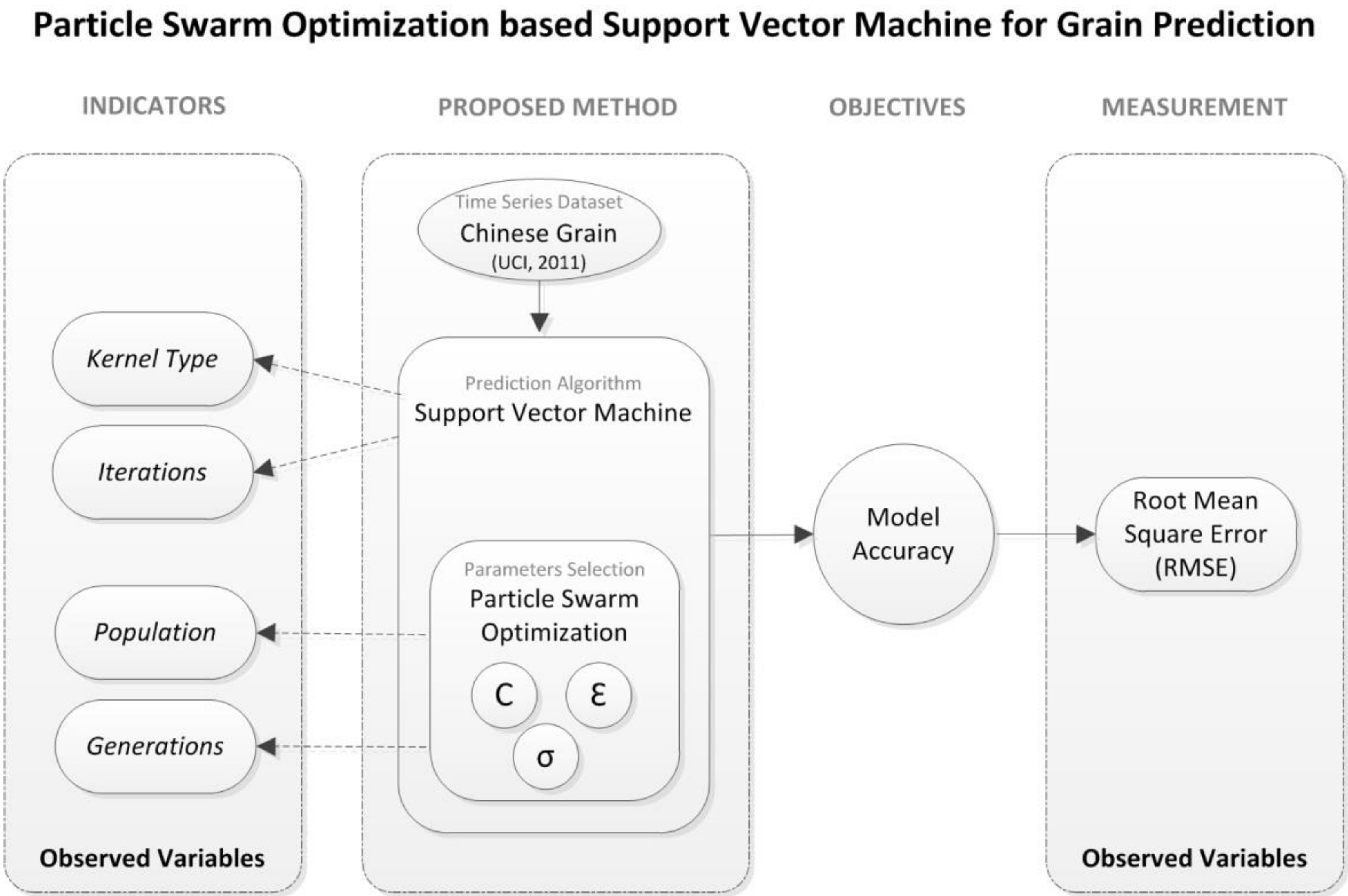
Kerangka Pemikiran

- Kerangka pemikiran adalah suatu bagan alur yang menghubungkan masalah dan pendekatan penelitian yang dihasilkan dari teori/konsep/model yang ada di landasan teori
- Kerangka pemikiran menjelaskan bagaimana pola pikir dan konsep kita dalam melakukan penelitian
- Kerangka pemikiran akan menjadi acuan kita dalam menyusun metodologi penelitian
- Kerangka pemikiran bisa digunakan untuk menguji logika penelitian

Kerangka Pemikiran

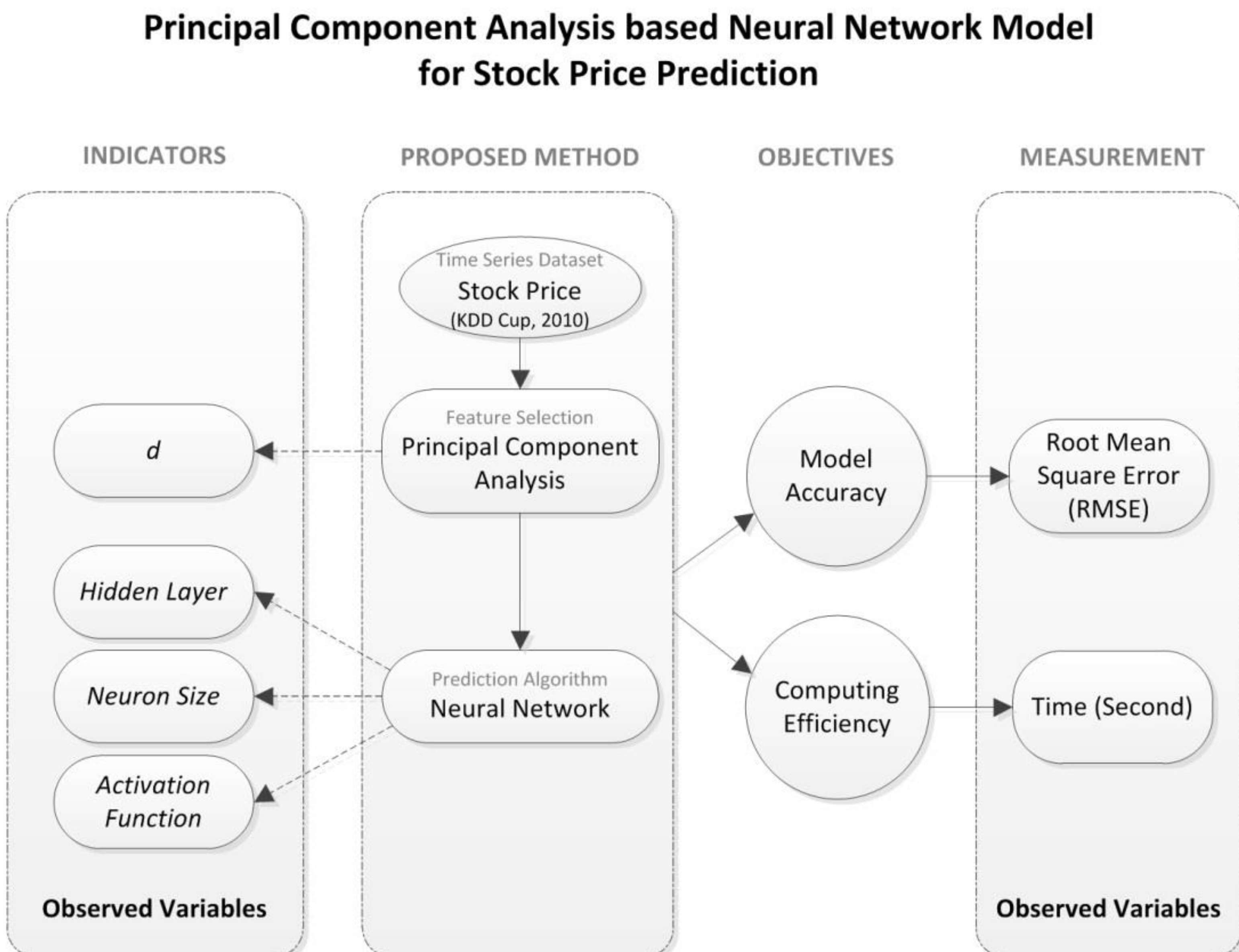
- Gunakan format Indicators, Proposed Method, Objective, Measurement, seperti yang ada di romisatriawahono.net
  - Format visio ada di folder *romi-rm/metode penelitian*
- Harus diuraikan dalam bentuk kalimat yg jelas, detil dan komprehensif yang menjelaskan semua gambar kerangka pemikiran

Contoh Kerangka Pemikiran*



*http://romisatriawahono.net/2012/08/07/kiat-menyusun-kerangka-pemikiran-penelitian/

Contoh Kerangka Pemikiran



4.3.5 Bab 3: Metode Penelitian

Struktur Tesis – Bab III

Bab III Metode Penelitian

3.1 Desain Penelitian

Metode penelitian yang dilakukan adalah metode penelitian eksperimen, dengan tahapan penelitian seperti berikut:

1. Pengumpulan Data (**Data Gathering**)
   (jelaskan langkah yang dilakukan di tahapan ini ...)
2. Pengolahan Awal Data (**Data Pre-processing**)
3. Model/Metode Yang Diusulkan/Dikembangkan (**Proposed Model/Method**)
4. Eksperimen dan Pengujian Model/Metode (**Model/Method Test and Experiment**)
5. Evaluasi dan Validasi Hasil (**Result Evaluation and Validation**)

Struktur Tesis – Bab III

3.2 Pengumpulan Data

(jelaskan tentang sumber data dan metode pengumpulan data)

3.3 Pengolahan Awal Data

(jelaskan teknik pengolahan awal (pre-processing) data yang akan dilakukan)

3.4 Metode Yang Diusulkan

(jelaskan perbaikan, revisi, usulan atau pengembangan metode/model yang telah kita lakukan dalam bentuk diagram skema dan formula)

3.5 Eksperimen dan Pengujian Metode

(jelaskan dengan detail dan algoritmik bagaimana teknik eksperimen/pengujian/penerapan metode/model yang akan dilakukan)

3.6 Evaluasi dan Validasi Hasil

(jelaskan dengan detail dan algoritmik bagaimana teknik evaluasi dan validasi metode/model yang akan dilakukan)

Metode Penelitian

- Intinya berisi desain dan tahapan penelitian
- Metode penelitian berisi rencana dan tahapan penelitian saja, bukan hasilnya apalagi pembahasannya

Contoh Tahapan Penelitian

1. **Pengumpulan Data**
2. **Pengolahan Awal Data**
3. **Model/Metode Yang Diusulkan/Dikembangkan**
4. **Eksperimen dan Pengujian Model/Metode**
5. **Evaluasi dan Validasi Hasil**

Desain Penelitian Eksperimen

1. Pre-Experimental Design
   1. One-Shot Case Study
   2. One Group Pretest-Posttest Design
   3. Intact-Group Comparison
2. True-Experimental Design
   1. Posttest Only Control Design
   2. Pretest-Control Group Design
3. Factorial Experimental Design
4. Quasi Experimental Design
   1. Time-Series Design
   2. Nonequivalent Control Group Design

Pre-Experimental Design

- Tidak ada variable kontrol dan data tidak dipilih secara random
- Belum disebut eksperimen yang sebenarnya, karena kemungkinan ada variabel eksternal yang mempengaruhi terbentuknya variable dependen
- Bentuk pre-experimental design:
  1. One-Shot Case Study
  2. One Group Pretest-Posttest Design
  3. Intact-Group Comparison

One-Shot Case Study

X O

X = perlakuan yang diberikan (variabel independen)

O = hasil (variabel dependen)

One Group Pretest-Posttest Design

$$O_1 \; X \; O_2$$

O_1 = Pretest
X = perlakuan yang diberikan
O_2 = Posttest

Intact-Group Comparison

$$\begin{array}{ll} X & O_1 \\ & O_2 \end{array}$$

X = perlakuan yang diberikan
O_1 = hasil pengukuran setengah kelompok yang diberi perlakuan
O_2 = hasil pengukuran setengah kelompok yang tidak diberi perlakuan

True-Experimental Design

- Kelompok kontrol dan sample diambil secara random dari populasi
- Peneliti dapat mengontrol semua variabel eksternal, sehingga validitas internal tinggi
- Bentuk true-experimental design:
  1. Posttest Only Control Design
  2. Pretest-Control Group Design

Posttest Only Control Design

$$\begin{array}{lll} \mathbf{R} & \mathbf{X} & \mathbf{O_1} \\ \mathbf{R} & & \mathbf{O_2} \end{array}$$

R = Random (Acak)
X = perlakuan yang diberikan
O_1 = hasil setelah perlakuan
O_2 = hasil tanpa perlakuan

Pretest-Control Group Design

$$
\begin{array}{llll}
\mathbf{R} & \mathbf{O_1} & \mathbf{X} & \mathbf{O_2} \\
\mathbf{R} & \mathbf{O_3} & & \mathbf{O_4}
\end{array}
$$

R = random (acak)
O_1 = pretest
X = perlakuan yang diberikan
O_2 = posttest setelah perlakuan
O_3 = pretest
O_2 = posttest tanpa perlakuan

Factorial Experimental Design

- Perbaikan dari true-experimental design dengan memperhatikan kemungkinan adanya variabel moderator yang mempengaruhi perlakuan
- Seluruh kelompok dipilih secara random dan masing-masing dilakukan pretest
- Kelompok penelitian dinyatakan baik apabila setiap kelompok memiliki nilai pretest yang sama

Factorial Experimental Design

$$\begin{array}{llll} R & O_1 & X & Y_1 \; O_2 \\ R & O_3 & & Y_1 \; O_4 \\ R & O_5 & X & Y_2 \; O_6 \\ R & O_7 & & Y_2 \; O_8 \end{array}$$

Y = variabel moderator

Quasi Experimental Design

- Digunakan ketika kita sulit mendapatkan kelompok kontrol seperti pada true-experimental design
- Lebih baik daripada pre-experimental, meskipun tidak sebaik true-experimental
- Bentuk quasi experimental design:
  1. Time-Series Design
  2. Nonequivalent Control Group Design

Time-Series Design

$$O_1\ O_2\ O_3\ O_4\ \ X\ \ O_5\ O_6\ O_7\ O_8$$

O_1 = Pretest
X = perlakuan yang diberikan
O_2 = Posttest

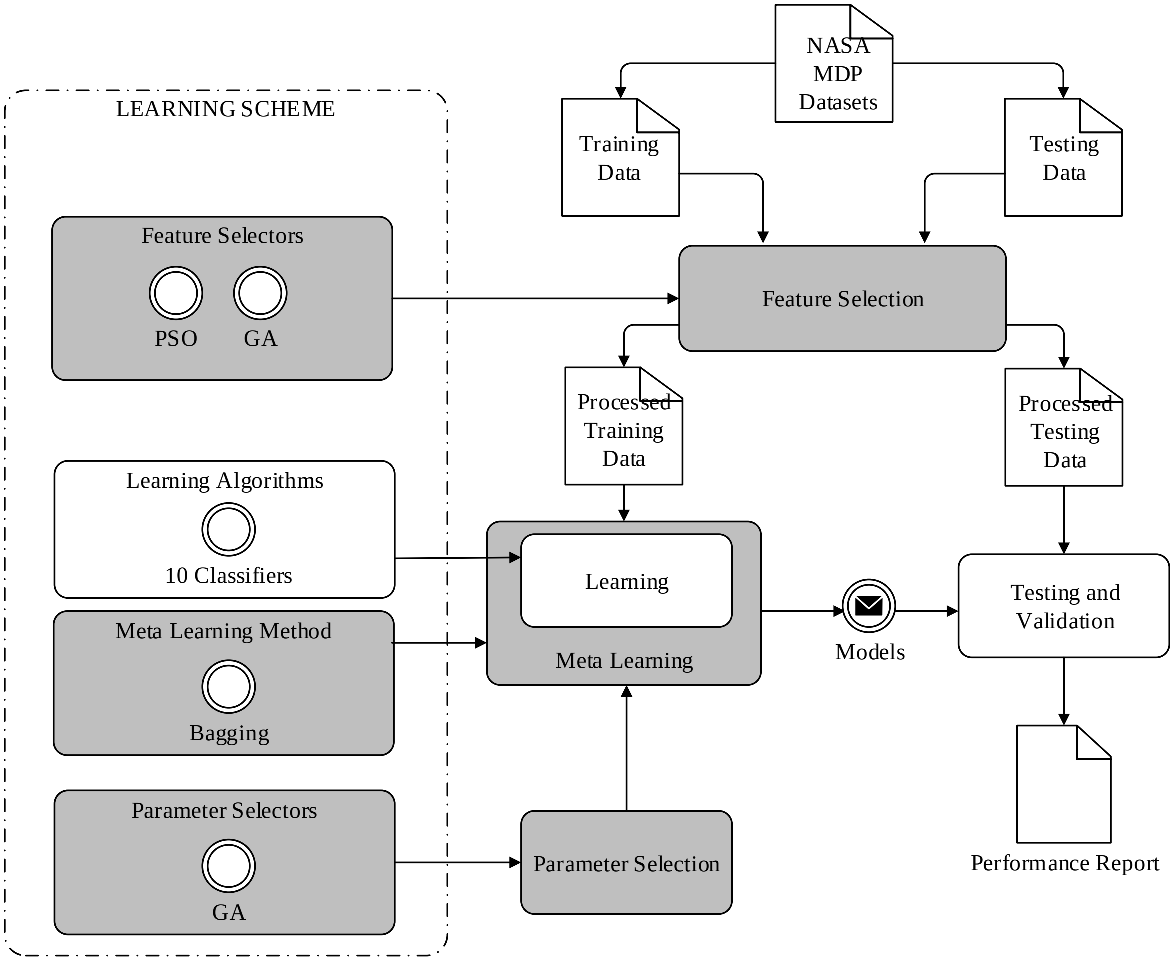
Nonequivalent Control Group Design

$$\begin{array}{ccc} \mathbf{O_1} & \mathbf{X} & \mathbf{O_2} \\ \mathbf{O_3} & & \mathbf{O_4} \end{array}$$

O_1 = pretest
X = perlakuan yang diberikan
O_2 = posttest setelah perlakuan
O_3 = pretest
O_2 = posttest tanpa perlakuan

- Hampir sama dengan pretest-posttest control group design, hanya pada desain ini kelompok tidak dipilih secara random

Proposed Framework



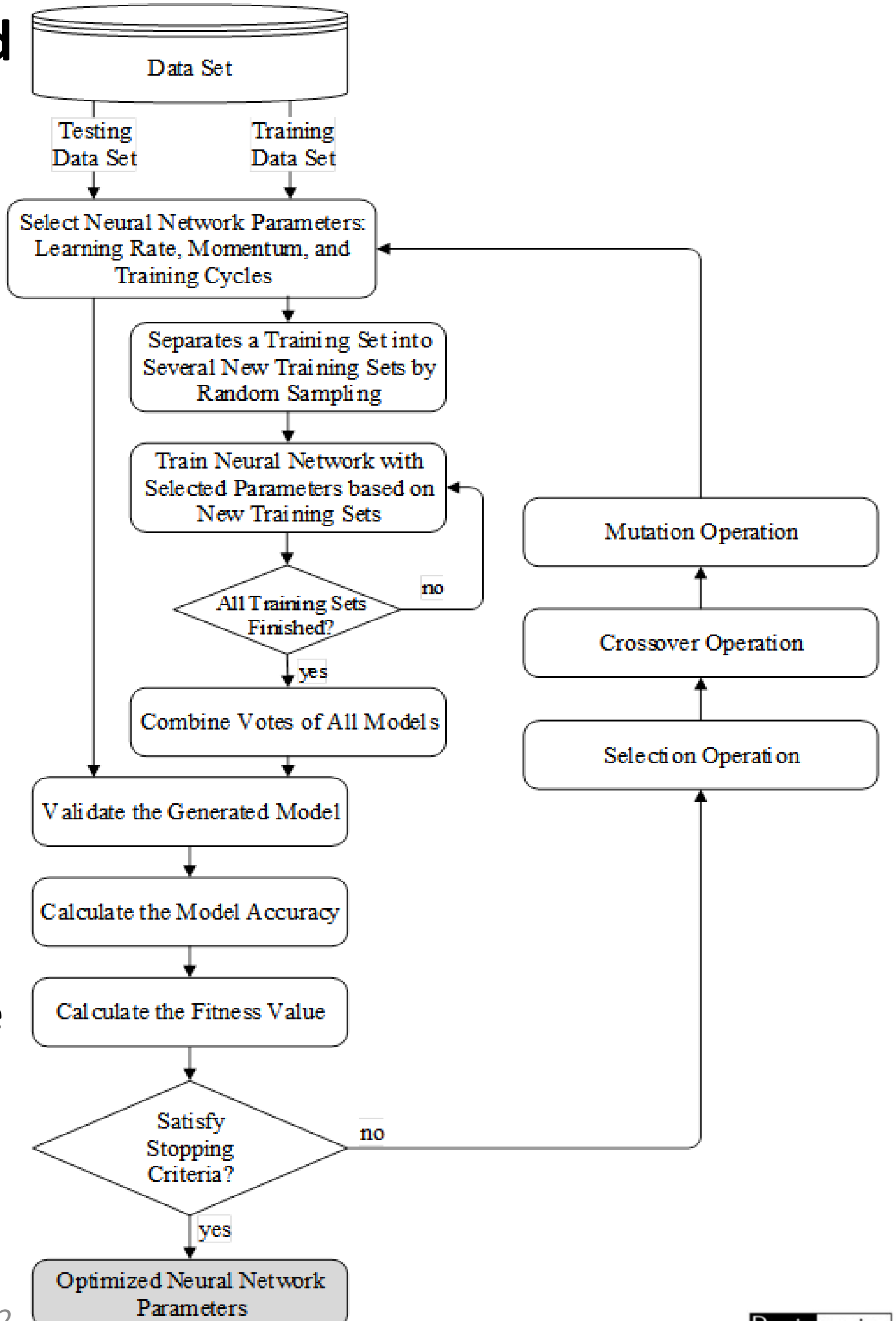
Framework	Dataset	Data Preprocessor	Feature Selectors	Meta-Learning	Classifiers	Parameter Selectors	Validation Methods	Evaluation Methods
(Menzies et al. 2007)	NASA MDP	Log Filtering	Info Gain		3 algorithm (DT, 1R, NB)	-	10-Fold X Validation	ROC Curve (AUC)
(Lessman et al. 2008)	NASA MDP	-	-		22 algorithm	-	10-Fold X Validation	ROC Curve (AUC)
(Song et al. 2011)	NASA MDP	Log Filtering	FS, BE		3 algorithm (DT, 1R, NB)	-	10-Fold X Validation	ROC Curve (AUC)
Proposed Framework	NASA MDP	-	**PSO, GA**	**Bagging**	**10 algorithms**	**GA**	10-Fold X Validation	ROC Curve (AUC)

A Hybrid Genetic Algorithm based Neural Network Parameter Optimization and Bagging Technique for Software Defect Prediction (NN GAPO+B)

- Every chromosome is evaluated by the fitness function Equation

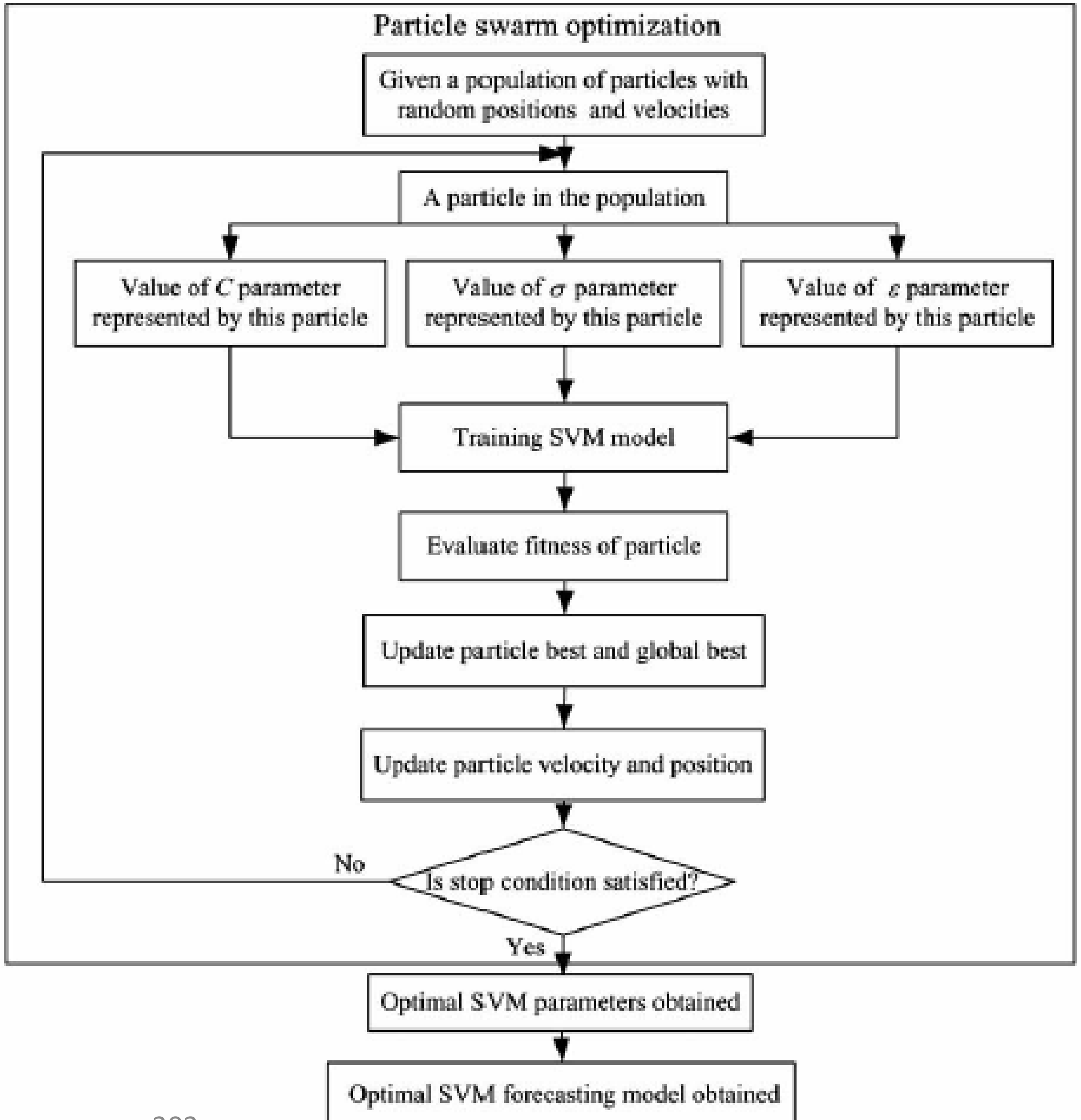
$$fitness = W_A \times A + W_P \times \left(S + \left(\sum_{i=1}^{n} C_i \times P_i \right) \right)^{-1}$$

- Where
  - A: classification accuracy
  - P_i: parameter value
  - W_A: weight of classification accuracy
  - W_p: parameter weight
  - C_i: feature cost
  - S: setting constant
- When ending condition is satisfied, the operation ends and the optimized NN parameters are produced. Otherwise, the process will continue with the next generation operation



Contoh Proposed Method

Metode yang diusulkan adalah metode SVM dengan pemilihan parameter C, Gamma dan Epsilon diotomatisasi menggunakan PSO



4.3.6 Bab 4: Hasil dan Pembahasan

Struktur Tesis – Bab IV

Bab IV Hasil dan Pembahasan

4.1 Hasil

(sajikan hasil eksperimen dan pengujian metode/model pada data eksperimen)

4.2 Pembahasan

(lakukan analisis dan pembahasan secara lengkap dan menyeluruh hasil eksperimen, evaluasi dan validasi hasil pengujian yang telah kita lakukan)

Contoh Kompilasi Hasil Eksperimen

Method	DS 1	DS 2	DS 3	DS 4
NN	2.4	3.1	1.5	6.7
NN + PCA	1.2	0.3	0.06	1.6

Year	Actual Value/10^4 t	GM		BP		PSO-SVM	
		Forecasting Value/10^4 t	Error/%	Forecasting Value/10^4 t	Error/%	Forecasting Value/10^4 t	Error/%
2001	45264	51983	14.852	47092	4.0436	45695	0.9573
2002	45706	52936	15.814	46971	2.7644	44848	-1.8797
2003	43070	53904	25.160	45836	6.4240	43407	0.7842
2004	46947	54893	16.924	51117	8.8830	44860	-4.447
2005	48401	55897	15.486	51176	5.7310	45773	-5.431
MAPE/%		17.647		5.5692		2.6998	

Analisis Statistik

1. Statistik Deskriptif
   - Nilai mean (rata-rata), standar deviasi, varians, data maksimal, data minimal, dsb

2. Statistik Inferensi
   - Perkiraan dan estimasi
   - Pengujian Hipotesis

Statistik Inferensi

Penggunaan	Parametrik	Non Parametrik
Dua sampel saling berhubungan (*Two Dependent samples*)	T Test Z Test	Sign test Wilcoxon Signed-Rank Mc Nemar Change test
Dua sampel tidak berhubungan (*Two Independent samples*)	T Test Z Test	Mann-Whitney U test Moses Extreme reactions Chi-Square test Kolmogorov-Smirnov test Walt-Wolfowitz runs
Beberapa sampel berhubungan (*Several Dependent Samples*)		Friedman test Kendall W test Cochran's Q
Beberapa sampel tidak Berhubungan (*Several Independent Samples*)	Anova test (F test)	Kruskal-Wallis test Chi-Square test Median test

Metode Parametrik

- Metode parametrik dapat dilakukan jika beberapa persyaratan dipenuhi, yaitu:
  - Sampel yang dianalisis haruslah berasal dari populasi yang berdistribusi normal
  - Jumlah data cukup banyak
  - Jenis data yang dianalisis adalah biasanya interval atau rasio

Metode Non Parametrik

- Metode ini dapat dipergunakan secara lebih luas, karena tidak mengharuskan datanya berdistribusi normal
  - Dapat dipakai untuk data nominal dan ordinal sehingga sangat berguna bagi para peneliti sosial untuk meneliti perilaku konsumen, sikap manusia, dsb
  - Cenderung lebih sederhana dibandingkan dengan metode parametrik
- Selain keuntungannya, berikut kelemahan metode non parametrik:
  - Tidak adanya sistematika yang jelas seperti metode parametrik
  - Terlalu sederhana sehingga sering meragukan
  - Memakai tabel-tabel yang lebih bervariasi dibandingkan dengan tabel-tabel standar pada metode parametrik

4.3.7 Kesimpulan dan Saran

Struktur Tesis – Bab V

Bab V Kesimpulan dan Saran

5.1 Kesimpulan

(menjawab rumusan masalah, sinkron dengan tujuan)

5.2 Saran

(future works yang akan dilakukan sebagai tahapan berikutnya dari penelitian kita, boleh dari temuan-temuan hasil eksperimen)

Kesimpulan

- Pernyataan umum (general) hasil penelitian
- Ringkasan dari temuan-temuan yang didapat dari analisa hasil penelitian
- Contoh:
  - Dari hasil eksperimen dan evaluasi penelitian, disimpulkan bahwa bahwa algoritma klasifikasi C4.5 akurat digunakan untuk penentuan kelayakan kredit perbankan
  - Dari hasil eksperimen dan evaluasi penelitian dapat disimpulkan bahwa akurasi metode fuzzy c-means pada pemetaan pemilihan peminatan mahasiswa mencapai 83%

Saran

- Langkah berikutnya setelah temuan diperoleh (*Future Works*)
- Saran bisa berupa teori, implementasi (praktis), atau untuk penelitian berikutnya
- Apabila kesimpulan menolak hipotesis, maka perlu disarankan penelitian lebih lanjut untuk menguji teori-teori yang ada
- Apabila kesimpulan menerima hipotesis, maka saran diarahkan ke langkah praktis bagaimana supaya hasil penelitian bisa diimplementasikan

Daftar Referensi

- Tidak boleh dibuat dengan ngawur dan asal-asalan
- Harus lengkap, penulis (bedakan mana family name dan mana first name), judul, publikasi, tahun, Vol., No., dsb
- Untuk yang pakai mendeley, cek dengan baik atribut paper, usahakan update dengan nomor DOI yang kita cari lewat internet or mendeley.com untuk menjamin kebenaran referensi
- Lakukan editing pada data referensi yang tidak rapi, misalnya judul kapital semua, tidak lengkap datanya, dsb

Tugas

- Susun Latar Belakang Masalah dari penelitian yang akan dilakukan dengan menggunakan pola OMKKMasaSolTu
- Tentukan pokok pikiran dahulu, baru uraikan dalam bentuk paragraf
- Tentukan Identifikasi Masalah (RP), Rumusan Masalah (RQ), dan Tujuan Penelitian (RO)
- Semua literature yg disitasi harus dimasukkan ke Mendeley, dan disertakan di Daftar Referensi
- Rangkumkan semua dalam format tesis menggunakan styling dan heading yg sudah diajarkan
- Deadline 2 pekan, upload docx di Trello.com

4.4 Publikasi Ilmiah untuk Jurnal Internasional

Kiat Penulisan Ilmiah

1. Tampilkan kontribusi pengetahuan dari penelitian yang dilakukan
2. Susunan penulisan menggunakan konsep IMRaD (Introduction, Methods, Result and Discussion)
3. Judul harus singkat, padat dan jelas menunjukkan kontribusi
4. Abstrak harus memuat masalah, metode dan hasil
5. Masalah penelitian harus tajam, eksplisit dan dilandasi
6. Metode yang diusulkan harus divalidasi dan diukur efektifitasnya
7. Penarikan kesimpulan harus sesuai dengan hasil penelitian dan selaras dan masalah dan tujuan penelitian
8. Pengambilan referensi penelitian harus dari jurnal yang terindeks oleh SCOPUS dan ISI

Proses Pengiriman Paper

1. Tentukan jurnal yang ingin kita kirim, cek nilai JIF dan SJR dari jurnal tersebut untuk mengukur kemampuan kita
2. Siapkan paper dengan mengikuti format penulisan yang stylenya sudah disediakan di situs journal
3. Kirimkan paper melalui sistem submission yang disediakan di situs journal
4. Paper yang sudah dikirim akan direview oleh reviewer. Proses review memakan waktu 3 – 12 bulan, tergantung kualitas jurnal yang kita kirim
5. Setelah menerima hasil review, perbaiki paper sesuai hasil review tersebut. Tahap ini bisa berulang ulang dilakukan
6. Setelah paper kita dinyatakan diterima tanpa perlu revisi lagi, kita tinggal menunggu nomor dan volume dari jurnal yang memuat paper kita

4.5 Penulisan Systematic Literature Review (SLR)

4.5.1 Tahapan Planning

4.5.2 Tahapan Conducting

4.5.3 Tahapan Reporting

Literature Review Methods

- Types and Methods of Literature Review:
  1. Traditional Review
  2. Systematic Literature Review or Systematic Review
  3. Systematic Mapping Study (Scoping Study)
  4. Tertiary Study

- SLR is now well established review method in the field of software engineering

(Kitchenham & Charters, Guidelines in performing Systematic Literature Reviews in Software Engineering, EBSE Technical Report version 2.3, 2007)

1. Traditional Review

- Provides an overview of the research findings on particular topics
- Advantages: produce insightful, valid syntheses of the research literature if conducted by the expert
- Disadvantages: vulnerable to unintentional and intentional bias in the selection, interpretation and organization of content
- Examples:
  - Liao et al., Intrusion Detection System: A Comprehensive Review, Journal of Network and Computer Applications, 36(2013)
  - Galar et al., A Review on Ensembles for the Class Imbalance Problem: Bagging-, Boosting-, and Hybrid-Based Approaches, IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews), Vol. 42, No. 4, July 2012
  - Cagatay Catal, Software fault prediction: A literature review and current trends, Expert Systems with Applications 38 (2011)

2. Systematic Mapping Study

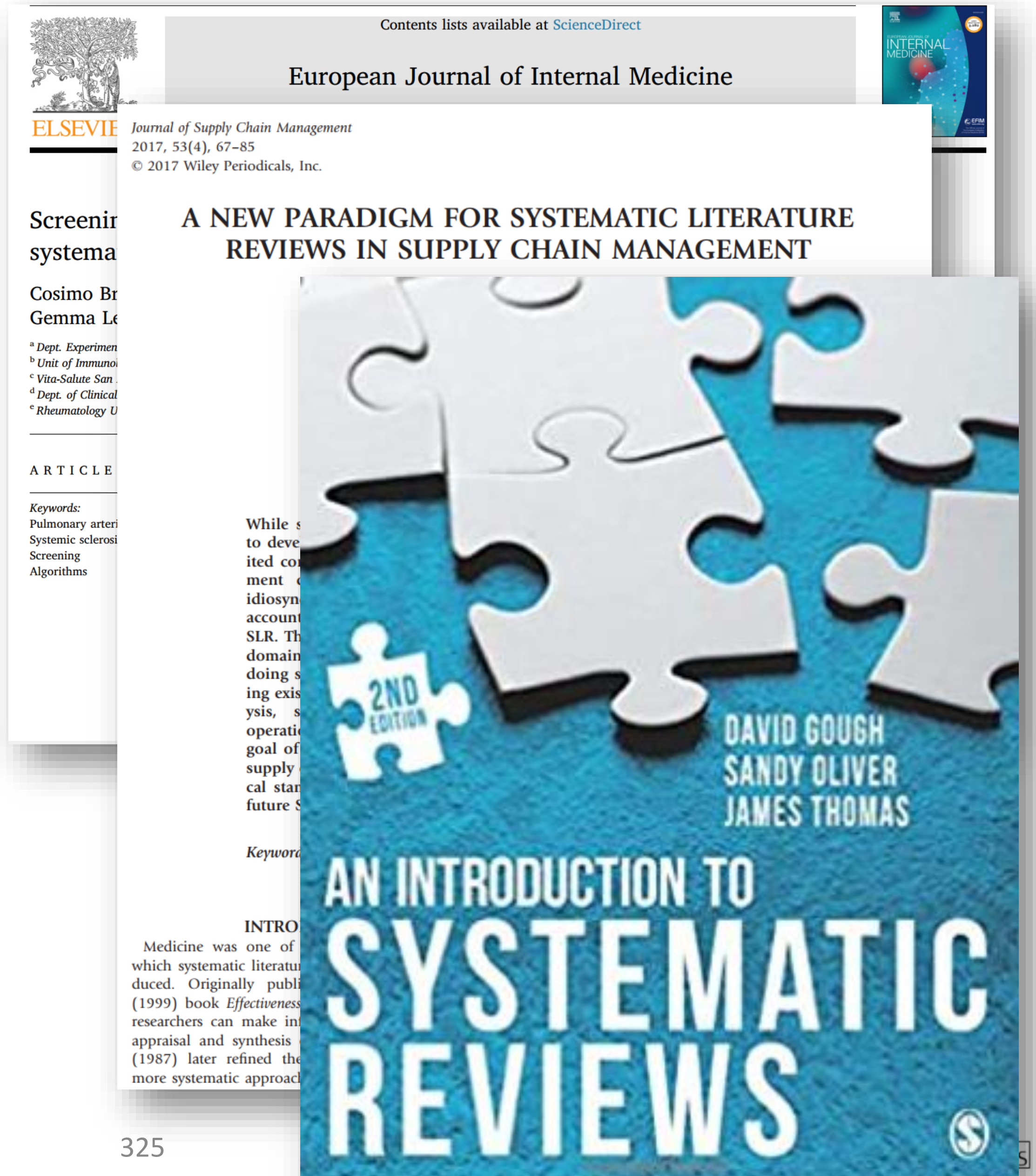
- Suitable for a very broad topic
- Identify clusters of evidence (making classification)
- Direct the focus of future SLRs
- To identify areas for future primary studies
- Examples:
  - Neto et al., A systematic mapping study of software product lines testing, Information and Software Technology Vol. 53, Issue 5, May 2011
  - Elberzhager et al., Reducing test effort: A systematic mapping study on existing approaches, Information and Software Technology 54 (2012)

3. Systematic Literature Review (SLR)

- Systematic reviews are a type of literature review that uses systematic methods to collect secondary data, critically appraise research studies, and synthesize findings qualitatively or quantitatively *(Amstrong et al., 2011)*
- A process of identifying, assessing, and interpreting all available research evidence, to provide answers for a particular research question (RQ)
- They are designed to provide a complete, exhaustive summary of current evidence, that is methodical, comprehensive, transparent, and replicable
- SLRs are well established in other disciplines, particularly medicine, biomedic, healthcare
- SLR application in the various fields:
    - Medicine *(Archie Cochrane, 1974)*
    - Computing Field *(Kitchenham & Charter, 2007)*
    - Social Science *(Gough, 2016)*
    - Business Management *(Durach et al., 2017)*

Contoh Systematic Literature Review (SLR)

- Romi Satria Wahono, A Systematic Literature Review of Software Defect Prediction: Research Trends, Datasets, Methods and Frameworks, Journal of Software Engineering, Vol. 1, No. 1, April 2015
- Christian F. Durach Joakim Kembro Andreas Wieland, A New Paradigm for Systematic Literature Reviews in Supply Chain Management, Journal of Supply Chain Management, Vol. 53(4), pp 67–85, 2017
- Matthias Galster, Danny Weyns, Dan Tofan, Bartosz Michalik, and Paris Avgeriou, Variability in Software Systems: A Systematic Literature Review, IEEE Transactions on Software Engineering, Vol 40, No 3, 2014



Contoh Systematic Literature Review (SLR)

Contents lists available at SciVerse ScienceDirect

Information and Software Technology

journal homepage: www.elsevier.com/locate/infsof

ELSEVIER

Systematic literature review of machine learning based software development effort estimatior

Jianfeng Wen [a,*], Shi

[a] Department of Computer Science,
[b] Department of Computer Science,
[c] Institute of Business Intelligence a
[d] Engineering Research Center of Co

A R T I C L E   I N F O

Article history:
Received 28 October 2010
Received in revised form 8 Augus
Accepted 8 September 2011
Available online 16 September 20

Keywords:
Software effort estimation
Machine learning
Systematic literature review

IEEE TRANSACTIONS ON SOFTWARE ENGINEERING, VOL. 38, NO. 6, NOVEMBER/DECEMBER 2

A Systematic Literature Review on Fault Prediction Performance in Software Engineering

Tracy Hall, Sarah Beecham, David Bowes, David Gray, and Steve Counsell

Abstract—*Background*: The accurate prediction of where faults are likely to occur in code can help direct test effort, reduce costs, and improve the quality of software. *Objective*: We investigate how the context of models, the independent variables used, and the modeling techniques applied influence the performance of fault prediction models. *Method:* We used a systematic literature review to identify 208 fault prediction studies published from January 2000 to December 2010. We synthesize the quantitative and qualitative results of 36 studies which report sufficient contextual and methodological information according to the criteria we develop and apply. *Results:* The models that perform well tend to be based on simple modeling techniques such as Naive Bayes or Logistic Regression. Combinations of independent variables have been used by models that perform well. Feature selection has been applied to these combinations when models are performing particularly well. *Conclusion:* The methodology used to build models seems to be influential to predictive performance. Although there are a set of fault prediction studies in which confidence is possible, more studies are needed that use a reliable methodology and which report their context, methodology, and performance comprehensively.

4. Tertiary study

- Is a SLR of SLRs
- To answer a more wider question
- Uses the same method as in SLR
- Potentially less resource intensive
- Examples:
  - Kitchenham et al., Systematic literature reviews in software engineering – A tertiary study, Information and Software Technology 52 (2010)
  - Cruzes et al., Research synthesis in software engineering: A tertiary study, Information and Software Technology 53 (2011)

Tahapan SLR

PLANNING

1. Formulate the review's research question
2. Develop the review's protocol

CONDUCTING

1. Identify the relevant literature
2. Perform selection of primary studies
3. Perform data extraction
4. Assess studies' quality
5. Conduct synthesis of evidence

REPORTING

1. Write up the SLR report/paper
2. Choose the Right Journal

4.5.1 Tahapan Planning

1. Formulate the Review's Research Question
2. Develop the Review's Protocol

1. Formulate the Review's Research Question

- Features of good question:
  - The RQ is meaningful and important to practitioners and researchers.
  - The RQ will lead to changes in current software engineering practice or to increase confidence in the value of current practice
  - The RQ will identify discrepancies between commonly held beliefs and the reality
- RQ can be derived primarily based on researcher's interest
  - An SLR for PhD thesis should identify existing basis for the research work and where it fits in the current body of knowledge

The Research Question (RQ)

- Is the most important part in any SLR
- Is not necessarily the same as questions addressed in your research
- Is used to guide the search process
- Is used to guide the extraction process
- Data analysis (synthesis of evidence) is expected to answer your SLR's RQ

RQ and PICOC

The formulation of RQs about effectiveness of a treatment should focus on 5 elements known as PICOC:

1. **Population (P)** - the target group for the investigation (e.g. people, software etc.)
2. **Intervention (I)** - specifies the investigation aspects or issues of interest to the researchers
3. **Comparison (C)**– aspect of the investigation with which the intervention is being compared to
4. **Outcomes (O)**– the effect of the intervention
5. **Context (C)**– the setting or environment of the investigation

(Petticrew et al., Systematic Reviews in the Social Sciences: A Practical Guide, Blackwell Publishing, 2006)

Example of PICOC (Kitchenham et al., 2007)

Kitchenham et al., A Systematic Review of Cross- vs. Within-Company Cost Estimation Studies, *IEEE Transactions on Software Engineering*, 33 (5), 2007

Population:	Software or web project
Intervention:	Cross-company project effort estimation model
Comparison:	Single-company project effort estimation model
Outcomes:	Prediction or estimate accuracy
Context:	None

Example of PICOC (Wahono, 2015)

Romi Satria Wahono, A Systematic Literature Review of Software Defect Prediction: Research Trends, Datasets, Methods and Frameworks, *Journal of Software Engineering, Vol. 1, No. 1*, pp. 1-16, April 2015

Population	Software, software application, software system, information system
Intervention	Software defect prediction, fault prediction, error-prone, detection, classification, estimation, models, methods, techniques, datasets
Comparison	n/a
Outcomes	Prediction accuracy of software defect, successful defect prediction methods
Context	Studies in industry and academia, small and large data sets

Example of RQs (Kitchenham, 2007)

Kitchenham et al., A Systematic Review of Cross- vs. Within-Company Cost Estimation Studies, *IEEE Transactions on Software Engineering*, 33 (5), 2007

- RQ1: What evidence is there that cross-company estimation models are not significantly different from within-company estimation models for predicting effort for software/Web projects?
- RQ2: What characteristics of the study data sets and the data analysis methods used in the study affect the outcome of within- and cross-company effort estimation accuracy studies?
- RQ3: Which experimental procedure is most appropriate for studies comparing within- and cross-company estimation models?

Example of RQs (Radjenovic et al., 2013)

Radjenovic et al., Software fault prediction metrics: A systematic literature review, *Information and Software Technology,* Vol. 8, No. 55, pp. 1397-1418, 2013

- RQ1: Which software metrics for fault prediction exist in literature?
- RQ2: What data sets are used for evaluating metrics?

Example of RQs (Fu Jia et al., 2020)

Fu Jia et al., Soybean Supply Chain Management and Sustainability: A Systematic Literature Review, *Journal of Cleaner Production*, 2020

- RQ1: What are the drivers and barriers to sustainable soy production and their relationships?
- RQ2: What are the value chain governance mechanisms available for the soybean chain?
- RQ3: What are the consequences of the implementation of these mechanisms?

Example of RQ (Wahono, 2015)

Romi Satria Wahono, A Systematic Literature Review of Software Defect Prediction: Research Trends, Datasets, Methods and Frameworks, *Journal of Software Engineering, Vol. 1, No. 1*, pp. 1-16, April 2015

ID	Research Question
RQ1	Which journal is the most significant software defect prediction journal?
RQ2	Who are the most active and influential researchers in the software defect prediction field?
RQ3	What kind of research topics are selected by researchers in the software defect prediction field?
RQ4	What kind of datasets are the most used for software defect prediction?
RQ5	What kind of methods are used for software defect prediction?
RQ6	What kind of methods are used most often for software defect prediction?
RQ7	Which method performs best when used for software defect prediction?
RQ8	What kind of method improvements are proposed for software defect prediction?
RQ9	What kind of frameworks are proposed for software defect prediction?

2. Develop the Review's Protocol

- A plan that specifies the basic review procedures (method)
- Components of a protocol:
  1. Background
  2. Research Questions
  3. Search terms
  4. Selection criteria
  5. Quality checklist and procedures
  6. Data extraction strategy
  7. Data synthesis strategy

4.5.2 Tahapan Conducting

1. Identify the Relevant Literature
2. Perform Selection of Primary Studies
3. Perform Data Extraction
4. Assess Studies' Quality
5. Conduct Synthesis of Evidence

1. Identifying Relevant Literature

- Involves a comprehensive and exhaustive searching of studies to be included in the review
- Define a search strategy
- Search strategies are usually iterative and benefit from:
  - Preliminary searches (to identify existing review and volume of studies)
  - Trial searches (combination of terms from RQ)
  - Check the search results against list of known studies
  - Consult the experts in the field

Approach to Construct Search String

- Derive major terms used in the review questions based on the PICOC
- List the keywords mentioned in the article
- Search for synonyms and alternative words
- Use the boolean OR to incorporate alternative synonyms
- Use the boolean AND to link major terms

Example of Search String (Wahono, 2015)

Romi Satria Wahono, A Systematic Literature Review of Software Defect Prediction: Research Trends, Datasets, Methods and Frameworks, *Journal of Software Engineering, Vol. 1, No. 1*, pp. 1-16, April 2015

Search String:

(software OR applicati OR systems) AND*

(fault OR defect* OR quality OR error-prone) AND*

(predict OR prone* OR probability OR assess* OR detect* OR estimat* OR classificat*)*

Example of Search String (Salleh et al., 2011)

- The complete search term initially used :

  (student OR undergraduate*) AND (pair programming OR pair-programming) AND ((experiment* OR measurement OR evaluation OR assessment) AND (effective* OR efficient OR successful)*

- A very limited number of results retrieved when using the complete string, thus a much simpler string was derived.
- Subject librarian suggested to revise the search string:

"pair programming" OR "pair-programming"

Example of Search String (Kitchenham et al., 2007)

- Kitchenham et al. (2007) used their structured questions to construct search strings for use with electronic databases:
  - *Population*: software OR application OR product OR Web OR WWW OR Internet OR World-Wide Web OR project OR development
  - *Intervention*: cross company OR cross organisation OR cross organization OR multiple-organizational OR multiple-organisational model OR modeling OR modelling effort OR cost OR resource estimation OR prediction OR assessment
  - *Contrast*: within-organisation OR within-organization OR within-organizational OR within-organisational OR single company OR single organisation
  - *Outcome*: Accuracy OR Mean Magnitude Relative Error
- The search strings were constructed by linking the four OR lists using the Boolean AND

Sources of Evidence (Kitchenham et al., 2007)

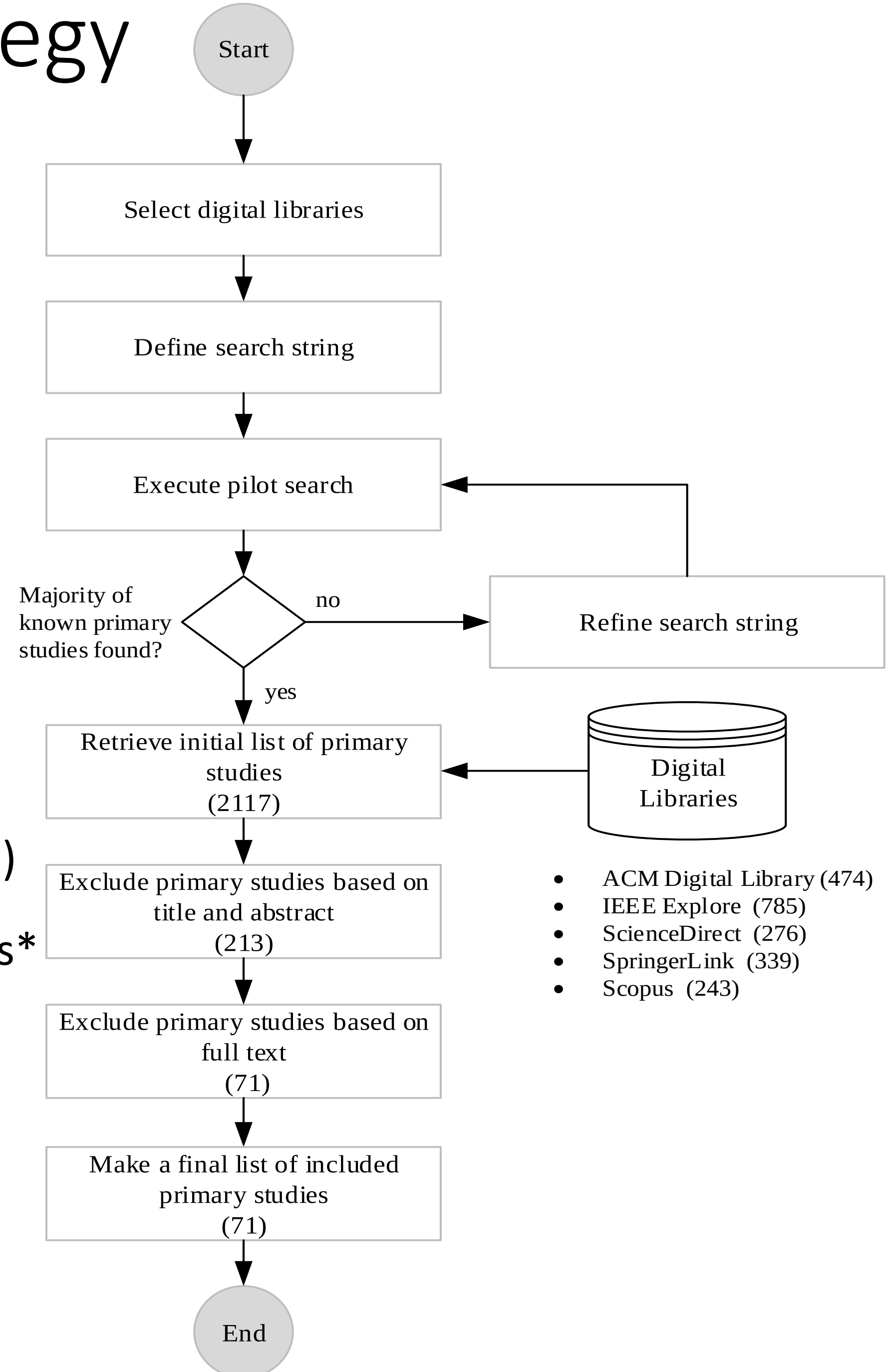
- The search strings were used on 6 digital libraries:
  - Science Direct, SpringerLink, SCOPUS, Web of Science, IEEExplore, ACM Digital library

- Search specific journals and conf. proceedings:
  - Empirical Software Engineering (J)
  - Information and Software Technology (J)
  - Software Process Improvement and Practice (J)
  - International Conference on Software Engineering (C)
  - Journal of Business Research (J)
  - Management Science (J)
  - International Business Review (J)

Studies Selection Strategy (Wahono, 2015)

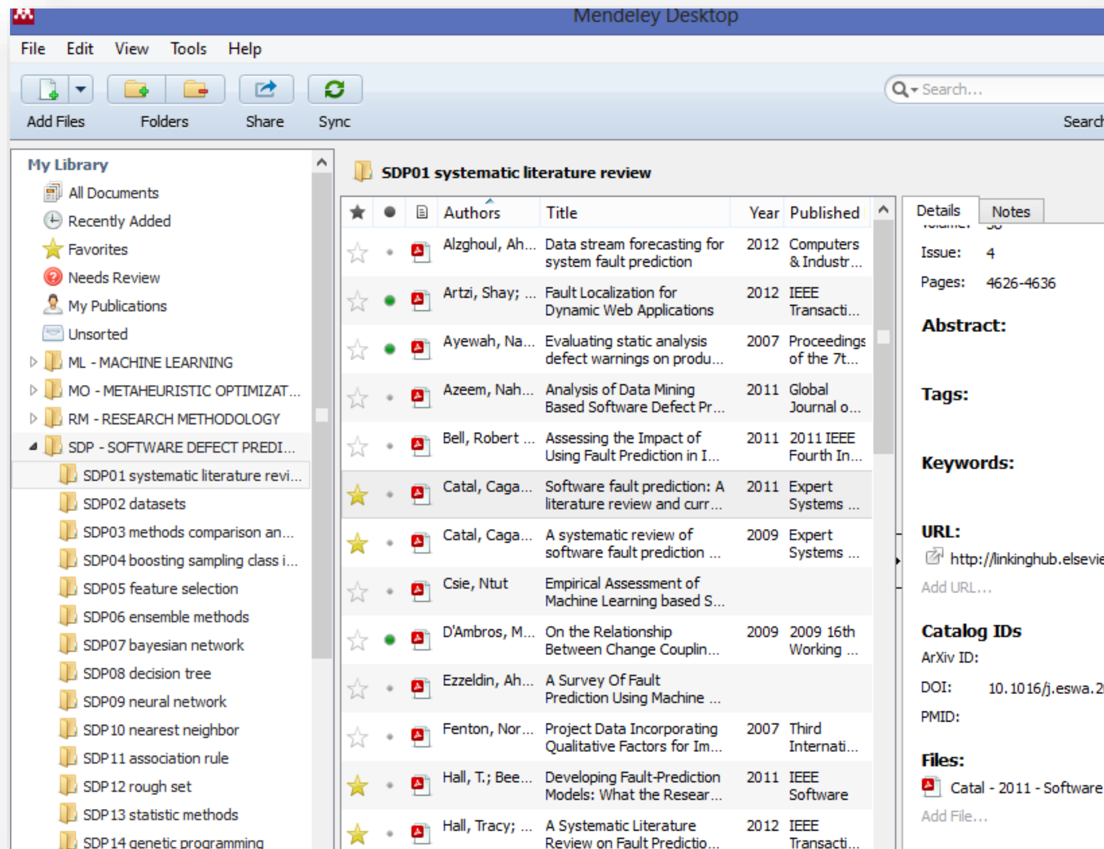
- Publication Year:
  - ✓2000-2013
- Publication Type:
  - ✓Journal
  - ✓Conference Proceedings
- Search String:

  software
  AND
  (fault* OR defect* OR quality OR error-prone)
  AND
  (predict* OR prone* OR probability OR assess* OR detect* OR estimat* OR classificat*)
- Selected Studies:
  - ✓71



Managing Bibliography

- Use relevant Bibliographic package to manage large number of references
- E.g. Mendeley, EndNote, Zotero, JabRef Reference Manager etc.



Documenting the Search

- The process of conducting SLR must be transparent and replicable
- The review should be documented in sufficient detail
- The search should be documented and changes noted
- Unfiltered search results should be saved for possible reanalysis

Data Source	Documentation
Digital Library	Name of Database, Search strategy, Date of search, years covered by search
Journal Hand Searches	Name of journal, Years searched
Conference proceedings	Title of proceedings/Name of conference, Journal name (if published as part of a journal)

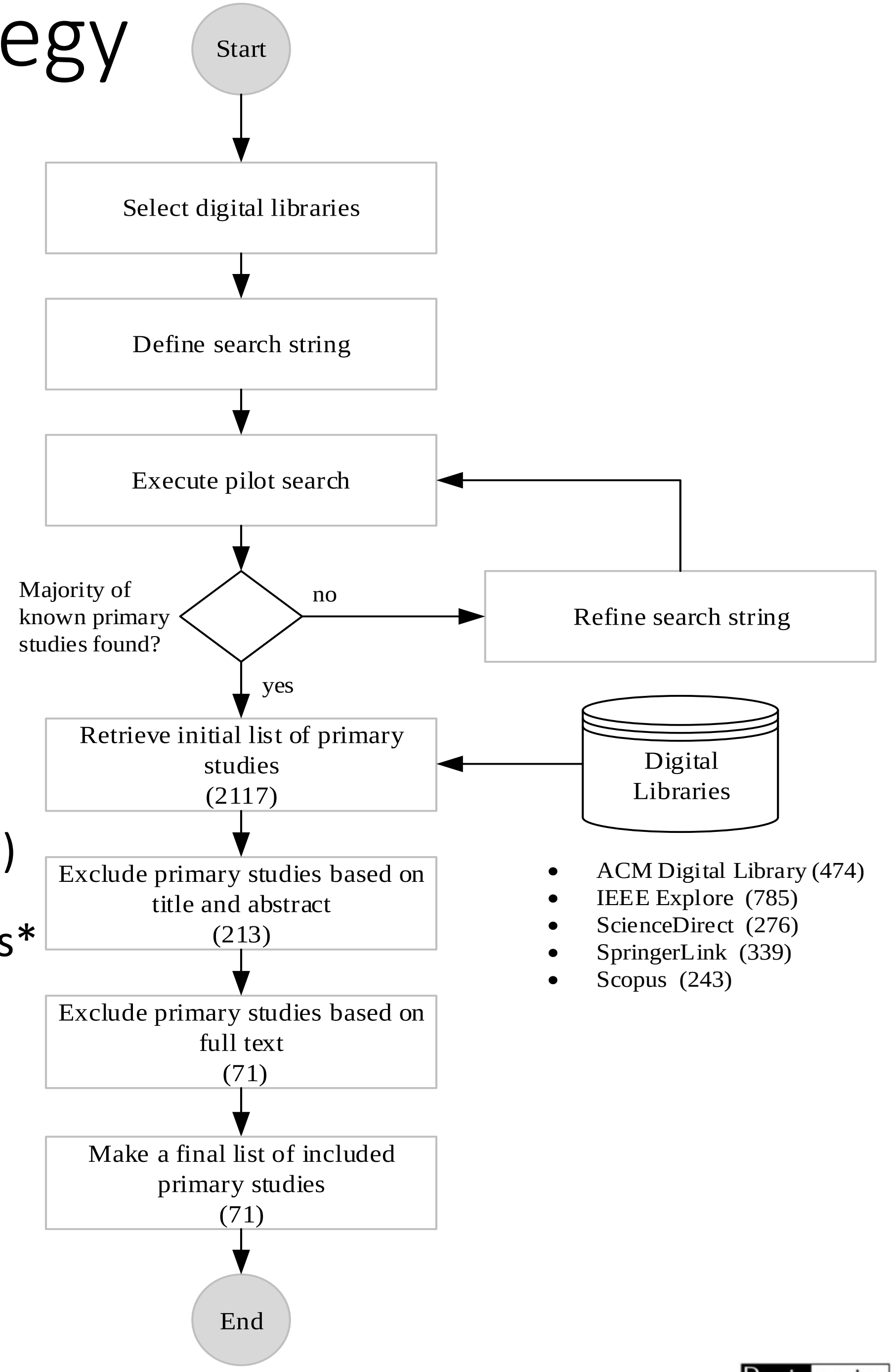
2. Selection of Studies

- Primary studies need to be assessed for their actual relevance
- Set the criteria for including or excluding studies (decided earlier during protocol development, can be refined later)
- Inclusion & exclusion criteria should be based on RQ
- Selection process should be piloted
- Study selection is a multistage process

Studies Selection Strategy (Wahono, 2015)

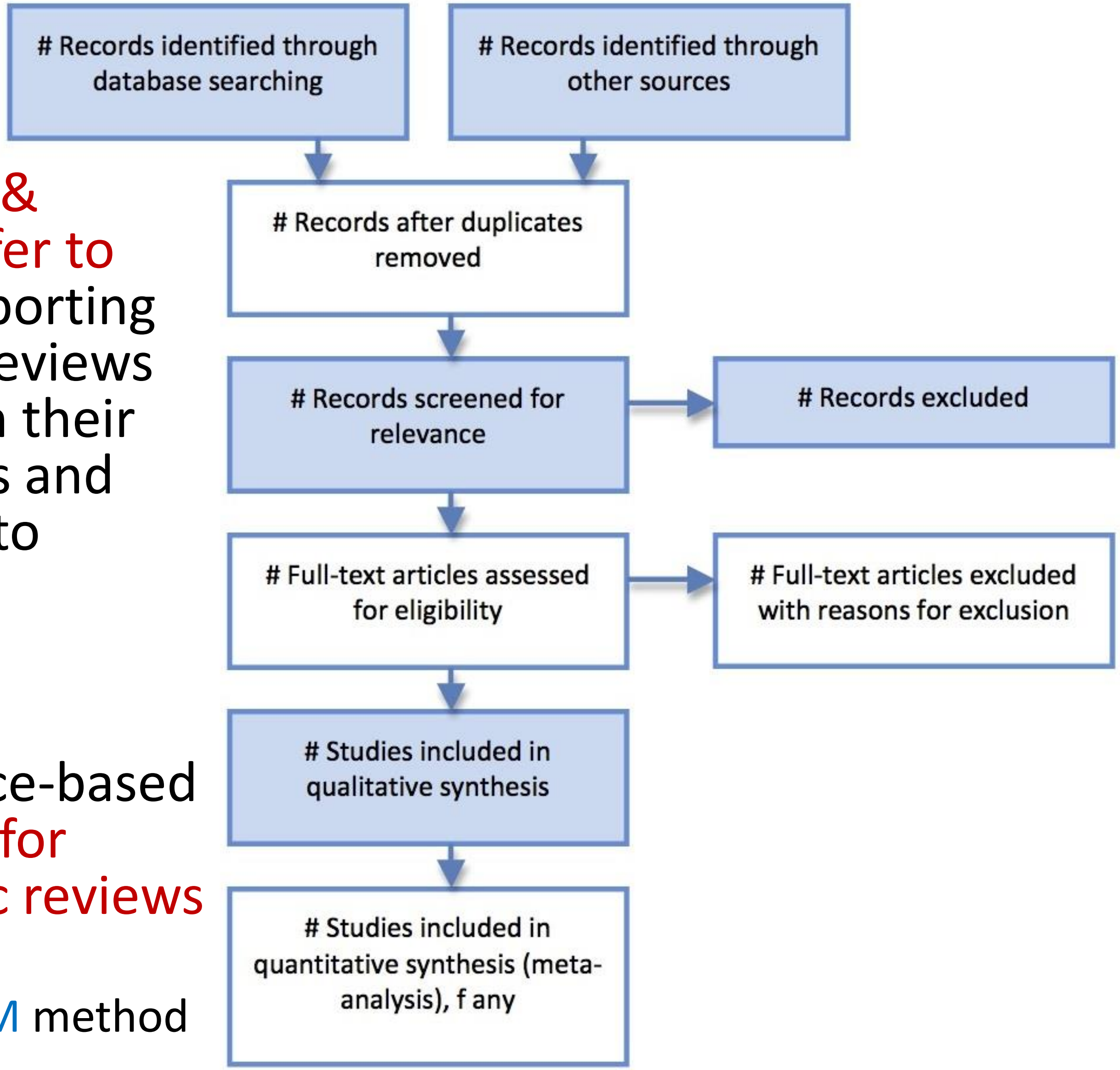
- Publication Year:
  - ✓2000-2013
- Publication Type:
  - ✓Journal
  - ✓Conference Proceedings
- Search String:

  software
  AND
  (fault* OR defect* OR quality OR error-prone)
  AND
  (predict* OR prone* OR probability OR assess* OR detect* OR estimat* OR classificat*)
- Selected Studies:
  - ✓71

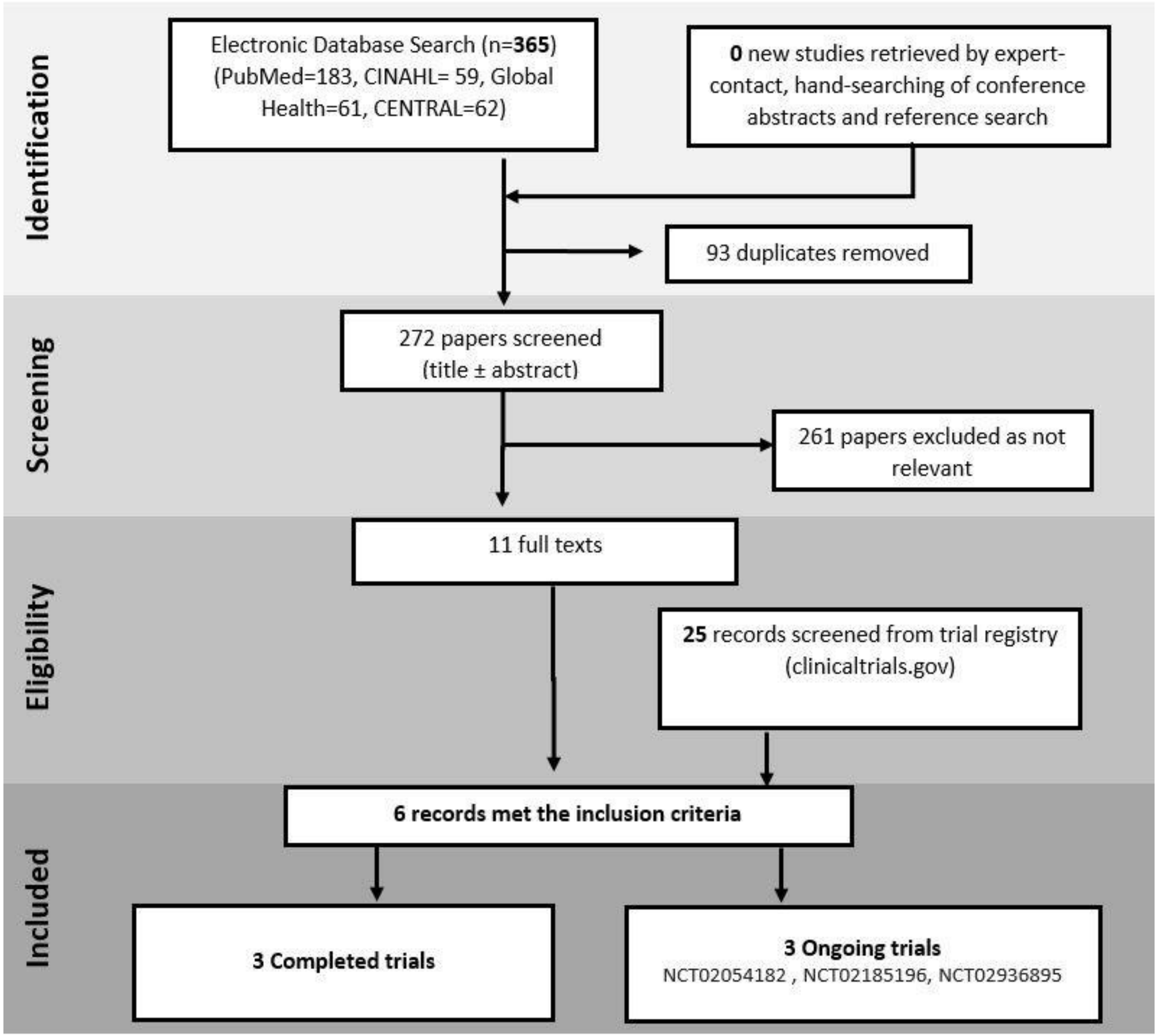


Studies Selection Strategy (PRISMA)

- Many leading medical & healthcare journals refer to PRISMA (Preferred Reporting Items for Systematic Reviews and Meta-Analyses) in their Instructions to Authors and some require authors to adhere to them *(Liberati et al., 2009)*
- (PRISMA) is an evidence-based minimum set of items for reporting in systematic reviews and meta-analyses
  - Replaced the QUOROM method



Studies Selection Strategy (PRISMA)



Selection of Studies (Kitchenham et al., 2007)

- Kitchenham et al. (2007) used the following inclusion criteria:
  - Any study that compared predictions of cross-company models with within-company models based on analysis of single company project data.
- They used the following exclusion criteria:
  - Studies where projects were only collected from a small number of different sources (e.g. 2 or 3 companies)
  - Studies where models derived from a within-company data set were compared with predictions from a general cost estimation model.

Selection of Studies (Wahono, 2015)

Inclusion Criteria	Studies in academic and industry using large and small scale data sets
	Studies discussing and comparing modeling performance in the area of software defect prediction
	For studies that have both the conference and journal versions, only the journal version will be included
	For duplicate publications of the same study, only the most complete and newest one will be included
Exclusion Criteria	Studies without a strong validation or including experimental results of software defect prediction
	Studies discussing defect prediction datasets, methods, frameworks in a context other than software defect prediction
	Studies not written in English

Selection of Studies (*Salleh et al., 2011)*

- **Inclusion criteria**:
  - to include any empirical studies of PP that involved higher education students as the population of interest

- **Exclusion criteria**:
  - Papers presenting unsubstantiated claims made by the author(s), for which no evidence was available.
  - Papers about Agile/XP describing development practices other than PP, such as test-first programming, refactoring etc.
  - Papers that only described tools (software or hardware) that could support the PP practice.
  - Papers not written in English.
  - Papers involving students but outside higher education

3. Assessing Studies' Quality

- To provide more detailed Inclusion/Exclusion criteria
- To check whether quality differences provide an explanation for differences in study results
- As a means of weighting the importance of individual studies when results are being synthesized
- To guide the interpretation of findings and determine the strength of inferences
- To guide recommendations for further research

Assessing Studies' Quality

- Quality relates to the extent to which the study:
  - Minimizes bias, and
  - Maximizes internal and external validity (Khan et al. 2001)
- Quality Concepts Definition (Kitchenham & Charter, 2007)

Terms	Synonyms	Definition
Bias	Systematic error	tendency to produce results that depart systematically from the 'true' results. Unbiased results are internally valid
Internal Validity	Validity	The extent to which the design and conduct of the study are likely to prevent systematic error. Internal validity is a prerequisite for external validity
External Validity	Generalizability, Applicability	The extent to which the effects observed in the study are applicable outside of the study

Assessing Studies' Quality

- Assessing quality of studies:
  - Methodology or design of the study
  - Analysis of studies' findings
- Quality checklist or instrument need to be designed to facilitate quality assessment
- Most quality checklists include questions aimed at assessing the extent to which articles have addressed bias and validity

Study Quality Assessment (Salleh et al., 2011)

Item	Answer
1. Was the article referred? [30]	Yes/No
2. Were the aim(s) of the study clearly stated? [16], [67]	Yes/No/Partially
3. Were the study participants or observational units adequately described? For example, students' programming experience, year of study etc. [44], [68]	Yes/No/Partially
4. Were the data collections carried out very well? For example, discussion of procedures used for collection, and how the study setting may have influenced the data collected [44], [48], [67], [68]	Yes/No/Partially
5. Were potential confounders adequately controlled for in the analysis? 67]	Yes/No/Partially
6. Were the approach to and formulation of the analysis well conveyed? For example, description of the form of the original data, rationale for choice of method/tool/package [48], [67], [68]	Yes/No/Partially
7. Were the findings credible? For example, the study was methodologically explained so that we can trust the findings; findings/conclusions are resonant with other knowledge and experience [48], [44], [68]	Yes/No/Partially

Study Quality Assessment (Kitchenham et al., 2007)

Kitchenham et al. (2007) constructed a quality questionnaire based on 5 issues affecting the quality of the study:

1. Is the data analysis process appropriate?
2. Did studies carry out a sensitivity or residual analysis?
3. Were accuracy statistics based on the raw data scale?
4. How good was the study comparison method?
5. The size of the within-company data set (e.g < 10 projects considered poor quality)

4. Data Extraction

- Involve reading the full text article
- Data extracted from primary studies should be recorded using *data extraction form*
- The form should be designed and piloted when the protocol is defined
- Collect all the information that can be used to answer the RQ and the study's quality criteria
- Both quality checklist and review data can be included in the same form
- In case of duplicates publications (reporting the same data), refer the most complete one
- For validation, a set of papers should be reviewed by 2 or more researchers. Compare results and resolve any conflicts

5. Synthesis of Evidence

- Involves collating and summarizing the results of the included primary studies
- Key objectives of data synthesis (Cruzes & Dyba, 2011):
  - to analyze and evaluate multiple studies
  - to select appropriate methods for integrating or providing new interpretive explanations about them
- Synthesis can be:
  - Descriptive (narrative/non-quantitative)
  - Quantitative (e.g. meta-analysis)

(Cruzes et al., Research Synthesis in Software Engineering: A tertiary study, *Information and Software Technology*, 53(5), 2011)

Descriptive Synthesis (Narrative)

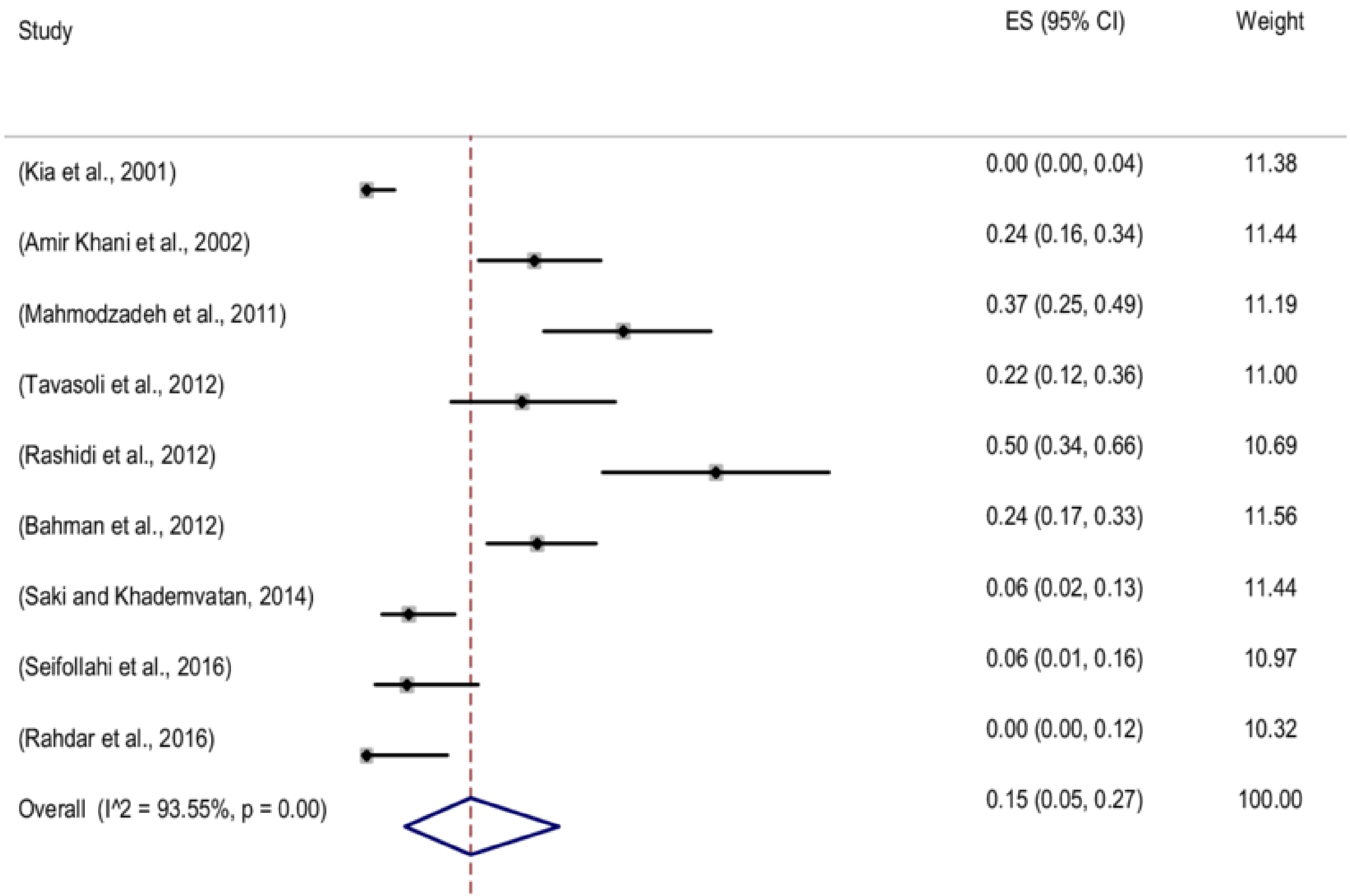
“An approach to the synthesis of findings from multiple studies that relies primarily on the use of words and text to summarize and explain the findings of the synthesis. It adopts a textual approach to the process of synthesis to ‘tell the story’ of the findings from the included studies.” (Popay et al. 2006)

- Use tables to tabulate information extracted from included studies (e.g. population, number of included studies, study quality etc.)
- Tables should be structured to highlight similarity or differences of study outcomes
- Were the findings consistent (homogeneous) or inconsistent?

Quantitative Synthesis (Meta-Analysis)

- Meta-analysis can be used to aggregate results or to pool data from different studies
- The outcome of a meta-analysis is an average effect size with an indication of how variable that effect size is between studies
- Meta-analysis involves three main steps:
  1. Decide which studies to be included in the meta-analysis
  2. Estimate an effect size for each individual study
  3. Combine the effect sizes from the individual studies to estimate and test the combined effect
- Results of the meta-analysis can be presented in a forest plot

Contoh Forest Plot



4.5.3 Tahapan Reporting

1. Write Up the SLR Paper
2. Choose the Right Journal

1. Write Up the SLR Paper

1. Introduction
   - General introduction about the research
   - State the purpose of the review
   - Emphasize the reason(s) why the RQ is important
   - State the significance of the review work and how the project contributes to the body of knowledge of the field
2. Main Body
   1. Review method – briefly describe steps taken to conduct the review
   2. Results – findings from the review
   3. Discussion – implication of review for research & practice
3. Conclusions

2. Choose the Right Journal

- Some journals and conferences include a specific topic on SLR:
  - Information & Software Technology has an editor specializing in systematic reviews
  - Journal of Systems and Software
  - Expert Systems with Applications
  - IEEE Transactions on Software Engineering
  - Journal of Cleaner Production
  - European Journal of Internal Medicine
  - Journal of Supply Chain Management

Listing Jurnal Tujuan dan Nilai SJR/JIF

- Lakukan pendataan journal-journal yang ada di topik SLR yang kita tulis, urutkan berdasarkan rangking SJR atau JIF
- Publikasikan paper SLR kita ke journal yang sesuai dengan kualitas SLR yang kita lakukan
- A paper is an organized description of hypotheses, data and conclusions, intended to instruct the reader. If your research does not generate papers, it might just as well not have been done (Whitesides 2004)

No	Journal Publications	SJR	Q Category
1	IEEE Transactions on Software Engineering	3.39	Q1 in Software
2	Information Sciences	2.96	Q1 in Information Systems
3	IEEE Transactions on Systems, Man, and Cybernetics	2.76	Q1 in Artificial Intelligence
4	IEEE Transactions on Knowledge and Data Engineering	2.68	Q1 in Information Systems
5	Empirical Software Engineering	2.32	Q1 in Software
6	Information and Software Technology	1.95	Q1 in Information Systems
7	Automated Software Engineering	1.78	Q1 in Software
8	IEEE Transactions on Reliability	1.43	Q1 in Software
9	Expert Systems with Applications	1.36	Q2 in Computer Science
10	Journal of Systems and Software	1.09	Q2 in Software
11	Software Quality Journal	0.83	Q2 in Software
12	IET Software	0.55	Q2 in Software
13	Advanced Science Letters	0.24	Q3 in Computer Science
14	Journal of Software	0.23	Q3 in Software
15	International Journal of Software Engineering and Its Application	0.14	Q4 in Software

Contoh SLR

Romi Satria Wahono, A Systematic Literature Review of Software Defect Prediction: Research Trends, Datasets, Methods and Frameworks, Journal of Software Engineering, Vol. 1, No. 1, pp. 1-16, April 2015

Tahapan SLR

PLANNING

1. Formulate the review's research question
2. Develop the review's protocol

CONDUCTING

1. Identify the relevant literature
2. Perform selection of primary studies
3. Perform data extraction
4. Assess studies' quality
5. Conduct synthesis of evidence

REPORTING

1. Write up the SLR report/paper
2. Choose the Right Journal

Planning

Romi Satria Wahono, A Systematic Literature Review of Software Defect Prediction: Research Trends, Datasets, Methods and Frameworks, Journal of Software Engineering, Vol. 1, No. 1, pp. 1-16, April 2015

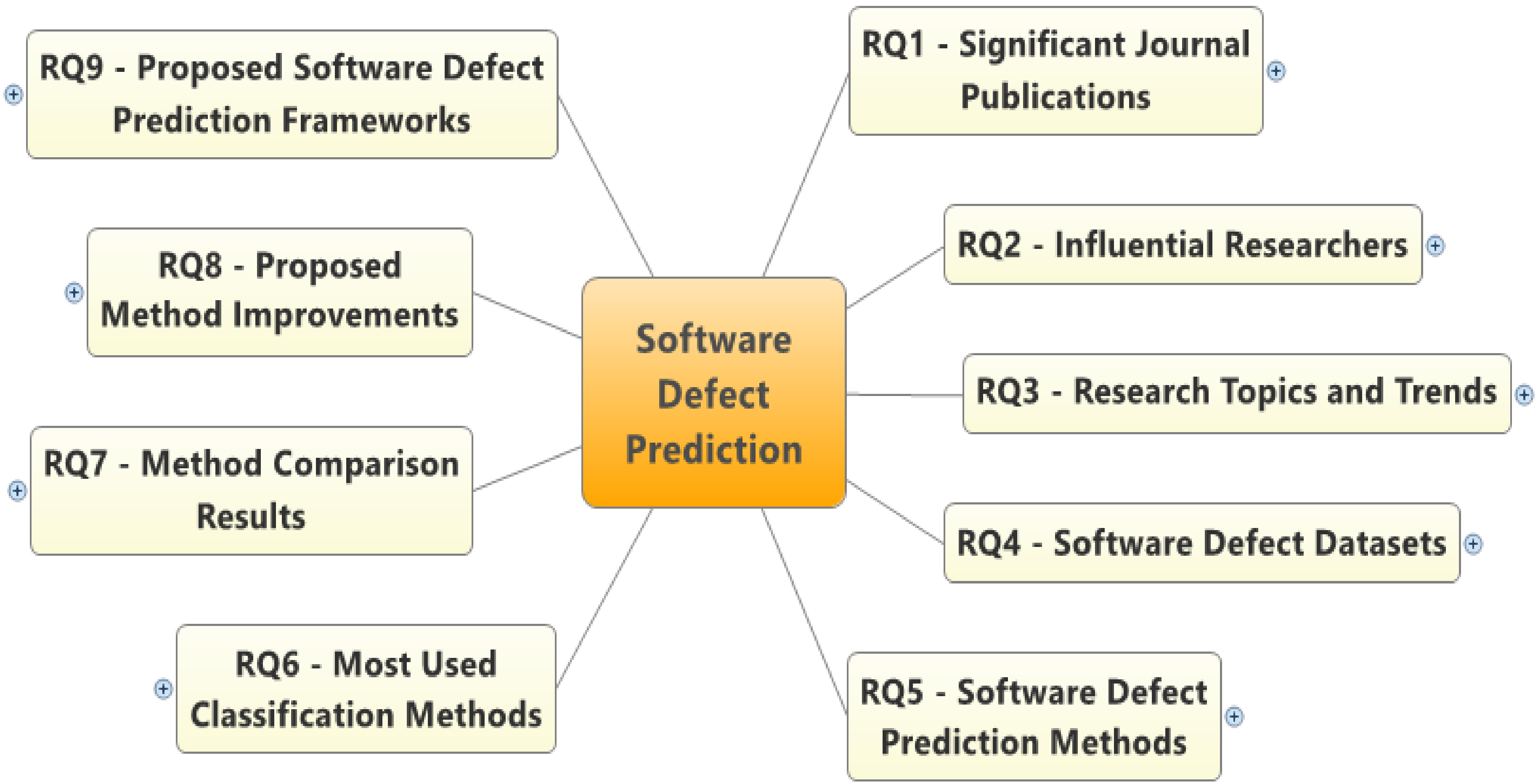
PICOC

Population	Software, software application, software system, information system
Intervention	Software defect prediction, fault prediction, error-prone, detection, classification, estimation, models, methods, techniques, datasets
Comparison	n/a
Outcomes	Prediction accuracy of software defect, successful defect prediction methods
Context	Studies in industry and academia, small and large data sets

Research Question (RQ)

ID	Research Question
RQ1	Which journal is the most significant software defect prediction journal?
RQ2	Who are the most active and influential researchers in the software defect prediction field?
RQ3	What kind of research topics are selected by researchers in the software defect prediction field?
RQ4	What kind of datasets are the most used for software defect prediction?
RQ5	What kind of methods are used for software defect prediction?
RQ6	What kind of methods are used most often for software defect prediction?
RQ7	Which method performs best when used for software defect prediction?
RQ8	What kind of method improvements are proposed for software defect prediction?
RQ9	What kind of frameworks are proposed for software defect prediction?

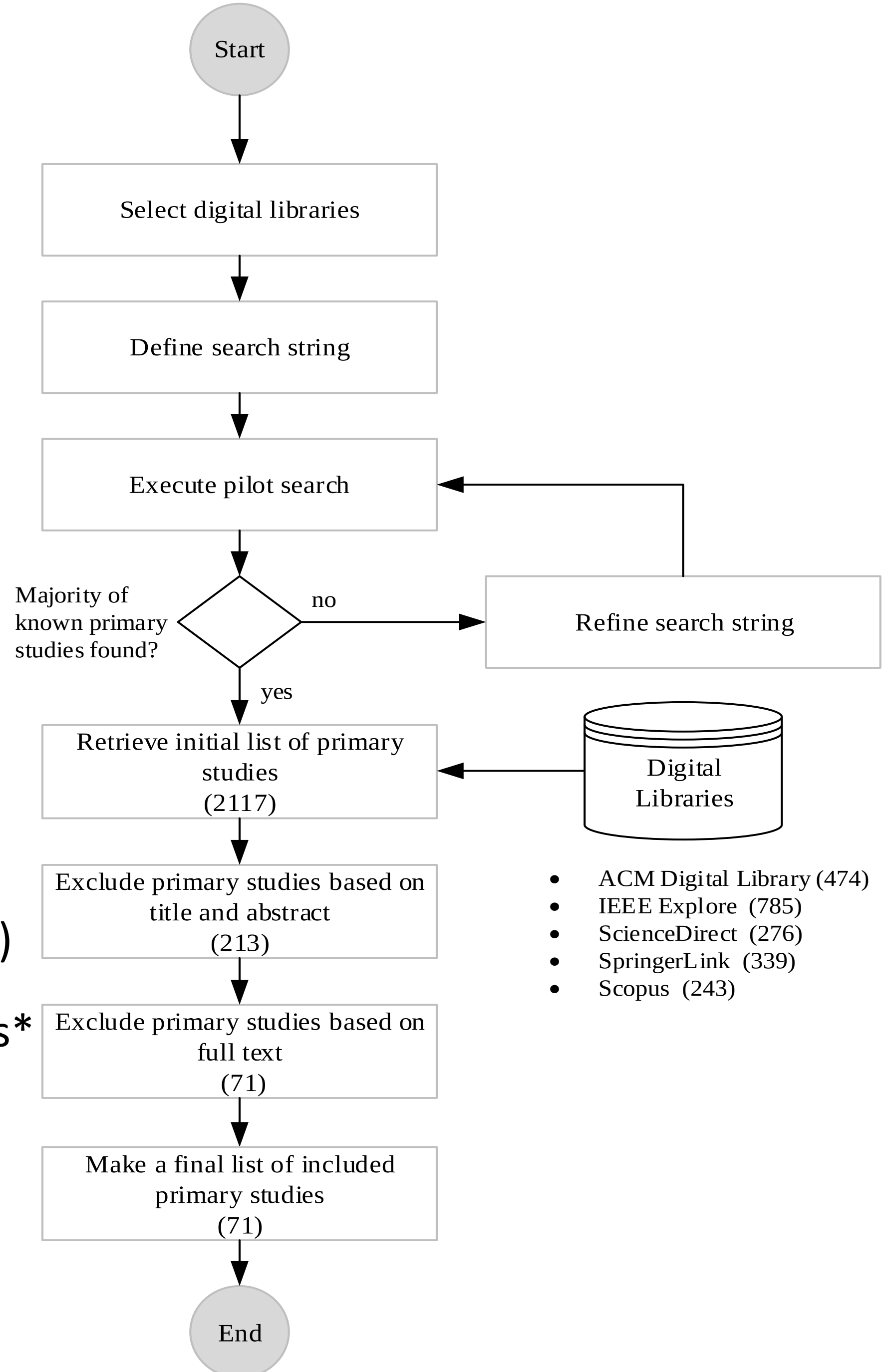
Research Question (RQ)



Studies Selection Strategy

- Publication Year:
  - ✓2000-2013
- Publication Type:
  - ✓Journal
  - ✓Conference Proceedings
- Search String:

  software
  AND
  (fault* OR defect* OR quality OR error-prone)
  AND
  (predict* OR prone* OR probability OR assess* OR detect* OR estimat* OR classificat*)
- Selected Studies:
  - ✓71



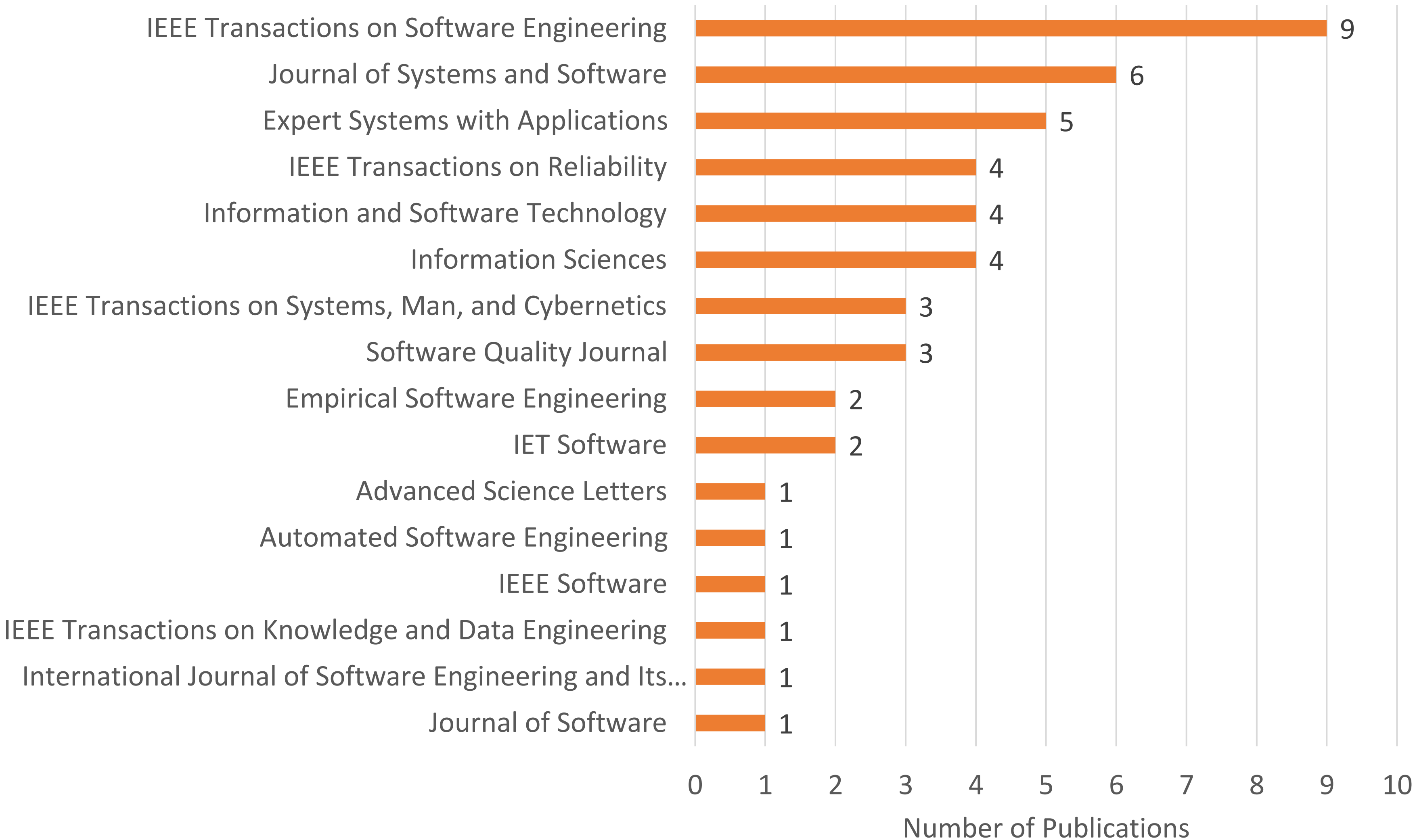
Inclusion and Exclusion Criteria

Inclusion Criteria	Studies in academic and industry using large and small scale data sets
	Studies discussing and comparing modeling performance in the area of software defect prediction
	For studies that have both the conference and journal versions, only the journal version will be included
	For duplicate publications of the same study, only the most complete and newest one will be included
Exclusion Criteria	Studies without a strong validation or including experimental results of software defect prediction
	Studies discussing defect prediction datasets, methods, frameworks in a context other than software defect prediction
	Studies not written in English

Result

Romi Satria Wahono, A Systematic Literature Review of Software Defect Prediction: Research Trends, Datasets, Methods and Frameworks, Journal of Software Engineering, Vol. 1, No. 1, pp. 1-16, April 2015

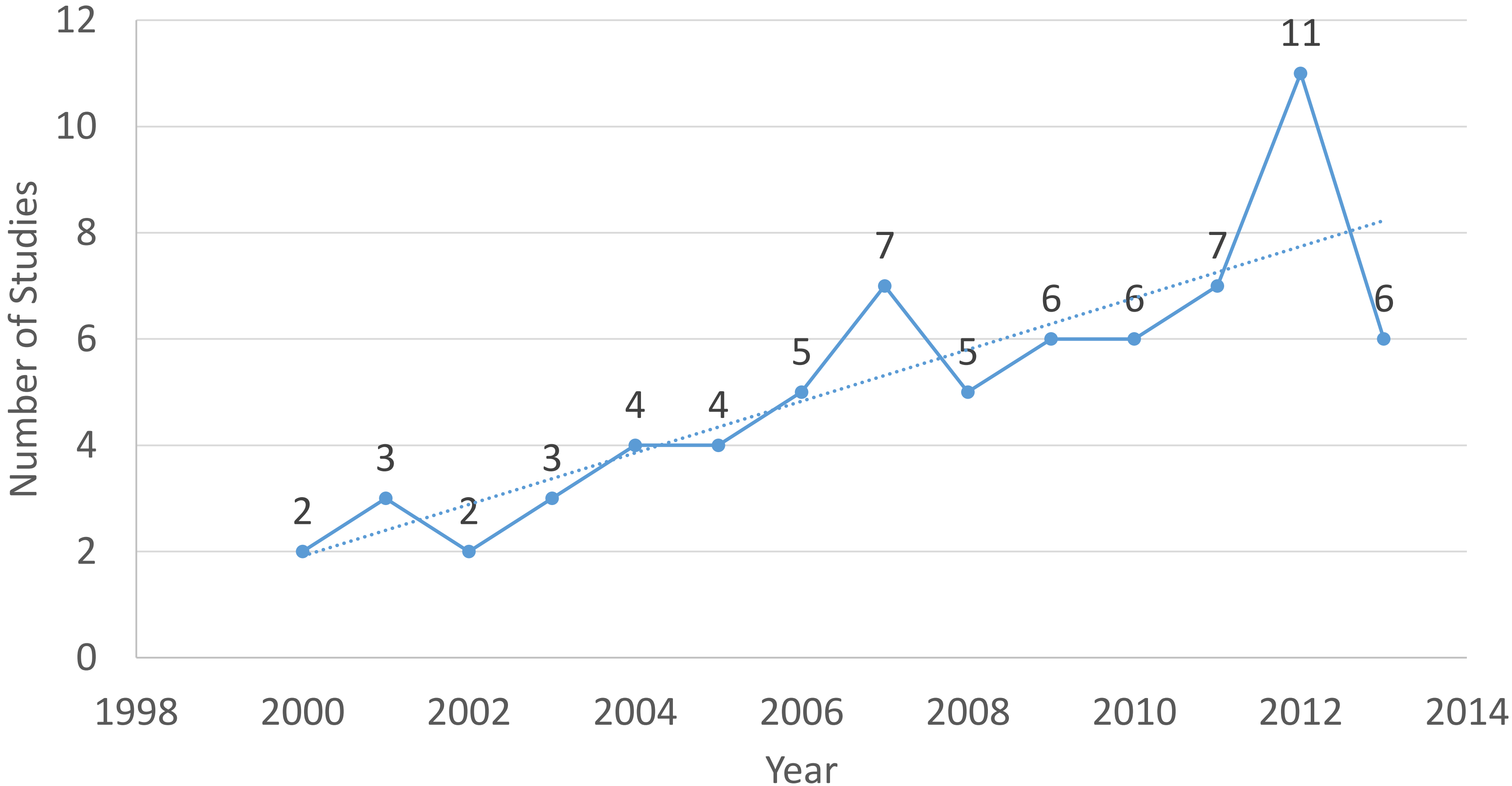
RQ1: Significant Journal Publications



Journal Quality Level of Selected Studies

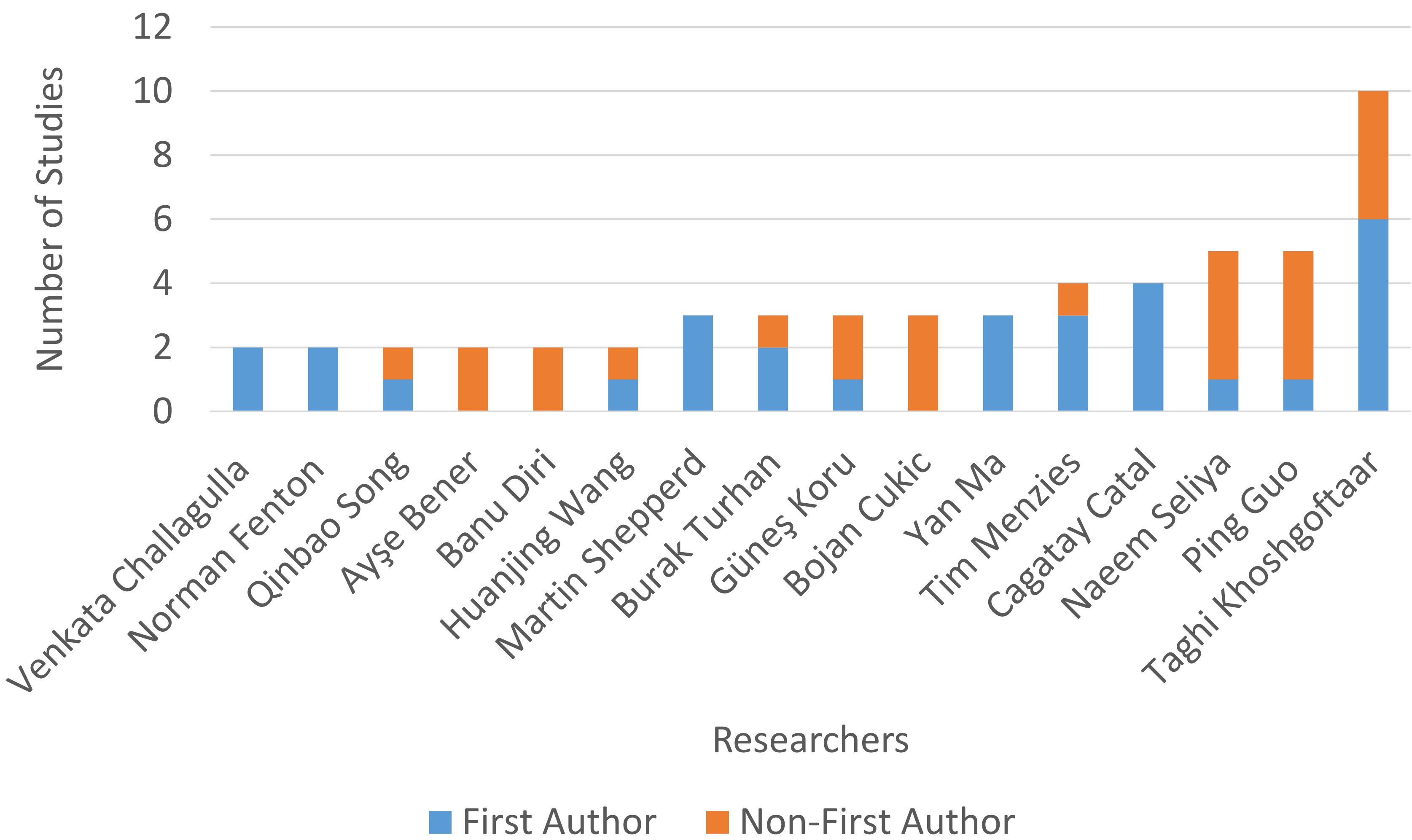
No	Journal Publications	SJR	Q Category
1	IEEE Transactions on Software Engineering	3.39	Q1 in Software
2	Information Sciences	2.96	Q1 in Information Systems
3	IEEE Transactions on Systems, Man, and Cybernetics	2.76	Q1 in Artificial Intelligence
4	IEEE Transactions on Knowledge and Data Engineering	2.68	Q1 in Information Systems
5	Empirical Software Engineering	2.32	Q1 in Software
6	Information and Software Technology	1.95	Q1 in Information Systems
7	Automated Software Engineering	1.78	Q1 in Software
8	IEEE Transactions on Reliability	1.43	Q1 in Software
9	Expert Systems with Applications	1.36	Q2 in Computer Science
10	Journal of Systems and Software	1.09	Q2 in Software
11	Software Quality Journal	0.83	Q2 in Software
12	IET Software	0.55	Q2 in Software
13	Advanced Science Letters	0.24	Q3 in Computer Science
14	Journal of Software	0.23	Q3 in Software
15	International Journal of Software Engineering and Its Application	0.14	Q4 in Software

Distribution of Selected Studies by Year



- The interest in software defect prediction has changed over time
- Software defect prediction research is still very much relevant to this day

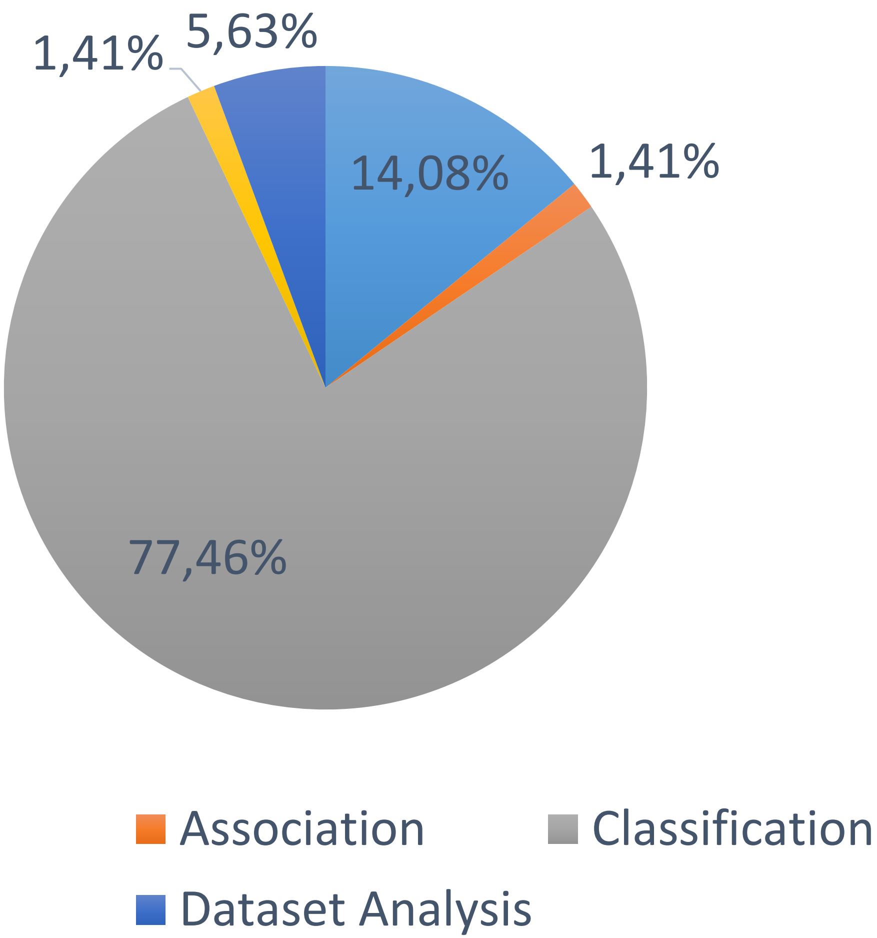
RQ2: Influential Researchers



RQ3: Research Topics and Trends

1. Estimating the number of defects remaining in software systems using estimation algorithm (Estimation)
2. Discovering defect associations using association rule algorithm (Association)
3. Classifying the defect-proneness of software modules, typically into two classes, defect-prone and not defect-prone, using classification algorithm (Classification)
4. Clustering the software defect based on object using clustering algorithm (Clustering)
5. Analyzing and pre-processing the software defect datasets (Dataset Analysis)

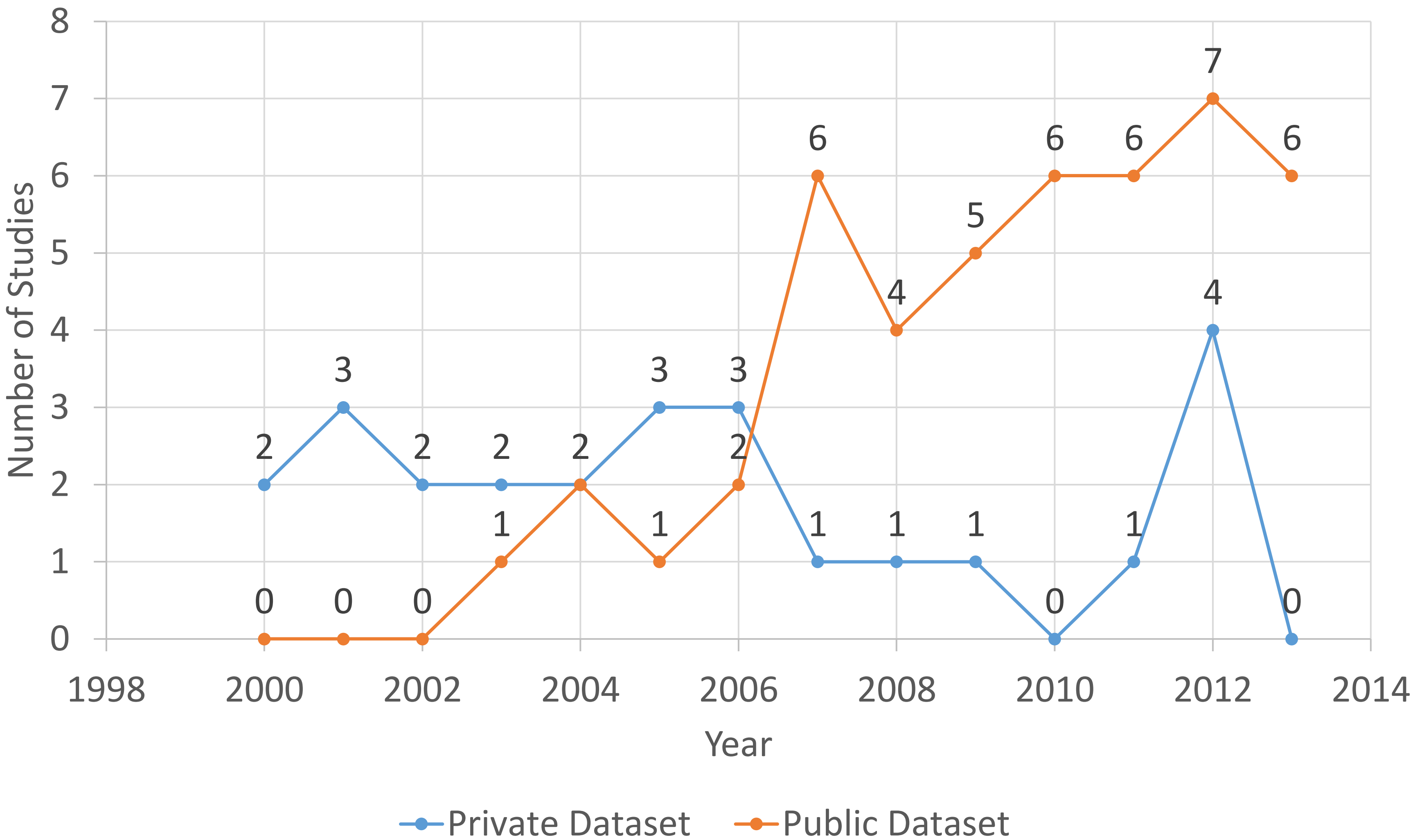
Distribution of Research Topics and Trends



Example Distribution of Research Topics and Trends

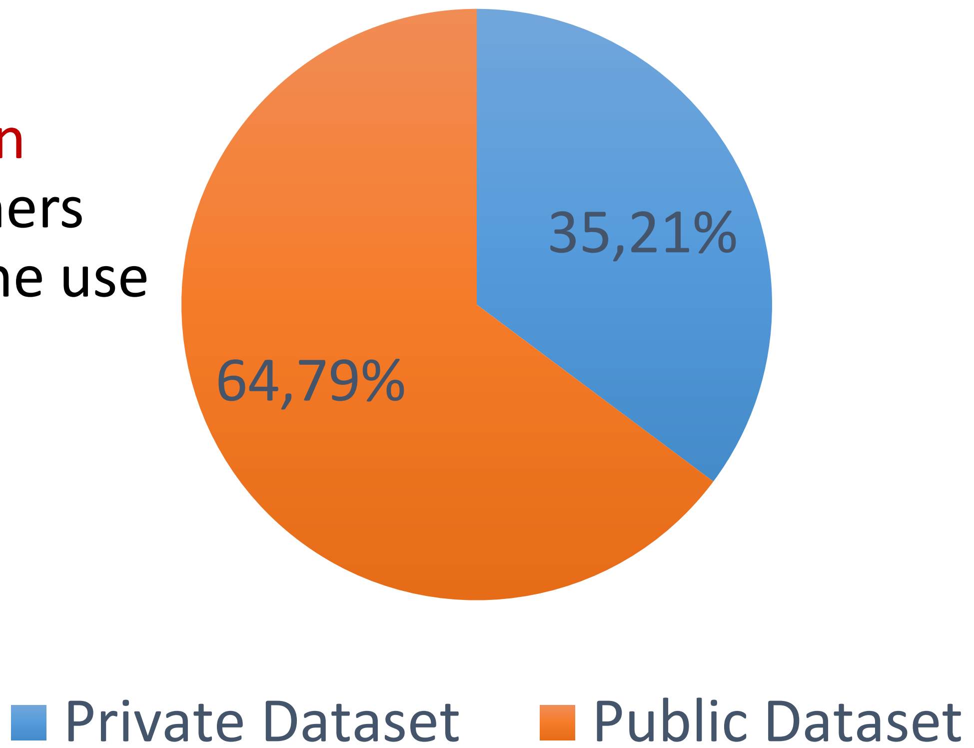
Year	Primary Studies	Publications	Datasets	Topics
2008	(Lessmann et al., 2008)	IEEE Transactions on Software Engineering	Public	Classification
	(Bibi et al., 2008)	Expert Systems with Applications	Private	Estimation
	(Gondra, 2008)	Journal of Systems and Software	Public	Classification
	(Vandecruys et al., 2008)	Journal of Systems and Software	Public	Classification
	(Elish and Elish 2008)	Journal of Systems and Software	Public	Classification
2012	(Gray et al., 2012)	IET Software	Public	Dataset Analysis
	(Ying Ma, Luo, Zeng, & Chen, 2012)	Information and Software Technology	Public	Classification
	(Benaddy and Wakrim 2012)	International Journal of Software Engineering	Private	Estimation
	(Y. Peng, Wang, & Wang, 2012)	Information Sciences	Public	Classification
	(Zhang and Chang 2012)	International Conference on Natural Computation	Private	Estimation
	(Bishnu and Bhattacherjee 2012)	IEEE Transactions on Knowledge and Data Engineering	Private	Clustering
	(Sun, Song, & Zhu, 2012)	IEEE Transactions on Systems, Man, and Cybernetics	Public	Classification
	(Pelayo and Dick 2012)	IEEE Transactions on Reliability	Public	Classification
	(Jin, Jin, & Ye, 2012)	IET Software	Public	Classification
	(Cao, Qin, & Feng, 2012)	Advanced Science Letters	Public	Classification
2013	(Park et al., 2013)	Information Sciences	Public	Classification
	(Dejaeger, Verbraken, & Baesens, 2013)	IEEE Transactions on Software Engineering	Public	Classification
	(Shepperd, Song, Sun, & Mair, 2013)	IEEE Transactions on Software Engineering	Public	Dataset Analysis
	(Wang and Yao 2013)	IEEE Transactions on Reliability	Public	Classification
	(Peters, Menzies, Gong, & Zhang, 2013)	IEEE Transactions on Software Engineering	Public	Dataset Analysis
	(Radjenović et al., 2013)	Information and Software Technology	Public	Dataset Analysis

RQ4: Software Defect Datasets



Distribution of Software Defect Datasets

- The use of public data sets makes the research repeatable, refutable, and verifiable (Catal & Diri 2009a)
- Since 2005 more public datasets were used
- NASA MDP repository have been developed in 2005 and researchers started to be aware regarding the use of public datasets



NASA MDP Dataset

Dataset	Project Description	Language	Number of Modules	Number of *fp* Modules	Faulty Percentage
CM1	Spacecraft instrument	C	505	48	12.21%
KC1	Storage management for ground data	C++	1571	319	15.51%
KC3	Storage management for ground data	Java	458	42	18%
MC2	Video guidance system	C	127	44	34.65%
MW1	Zero gravity experiment related to combustion	C	403	31	10.23%
PC1	Flight software from an earth orbiting satellite	C	1059	76	8.04%
PC2	Dynamic simulator for attitude control systems	C	4505	23	1.01%
PC3	Flight software for earth orbiting satellite	C	1511	160	12.44%
PC4	Flight software for earth orbiting satellite	C	1347	178	12.72%

Code Attributes		Symbols	Description
LOC counts	LOC_total		The total number of lines for a given module
	LOC_blank		The number of blank lines in a module
	LOC_code_and_comment	NCSLOC	The number of lines which contain both code and comment in a module
	LOC_comments		The number of lines of comments in a module
	LOC_executable		The number of lines of executable code for a module
	number_of_lines		Number of lines in a module
Halstead	content	μ	The halstead length content of a module $\mu = \mu_1 + \mu_2$
	difficulty	D	The halstead difficulty metric of a module D = 1/L
	effort	E	The halstead effort metric of a module E = V/L
	error_est	B	The halstead error estimate metric of a module $B = E^{2/3}/1000$
	length	N	The halstead length metric of a module $N = N_1+N_2$
	level	L	The halstead level metric of a module $L = (2^* \mu_2)/ \mu_1{}^*N_2$
	prog_time	T	The halstead programming time metric of a module T = E/18
	volume	V	The halstead volume metric of a module $V = N^*\log_2(\mu_1+ \mu_2)$
	num_operands	N_1	The number of operands contained in a module
	num_operators	N_2	The number of operators contained in a module
	num_unique_operands	μ_1	The number of unique operands contained in a module
	num_unique_operators	μ_2	The number of unique operators contained in a module
McCabe	cyclomatic_complexity	v(G)	The cyclomatic complexity of a module v(G) = e – n +2
	cyclomatic_density		v(G) / NCSLOC
	design_complexity	iv(G)	The design complexity of a module
	essential_complexity	ev(G)	The essential complexity of a module
Misc.	branch_count		Branch count metrics
	call_pairs		Number of calls to functions in a module
	condition_count		Number of conditionals in a given module
	decision_count		Number of decision points in a module
	decision_density		condition_count / decision_count
	edge_count		Number of edges found in a given module from one module to another
	essential_density		Essential density is calculated as: (ev(G)-1)/(v(G)-1)
	parameter_count		Number of parameters to a given module
	maintenance_severity		Maintenance Severity is calculated as: ev(G)/v(G)
	modified_condition_count		The effect of a condition affect a decision outcome by varying that condition only
	multiple_condition_count		Number of multiple conditions within a module
	global_data_complexity	gdv(G)	the ratio of cyclomatic complexity of a module’s structure to its parameter count
	global_data_density		Global Data density is calculated as: gdv(G)/v(G)
	normalized_cyclo_cmplx		v(G) / numbe_of_lines
	percent_comments		Percentage of the code that is comments
	node_count		Number of nodes found in a given module

Code Attributes		NASA MDP Dataset								
		CM1	KC1	KC3	MC2	MW1	PC1	PC2	PC3	PC4
LOC counts	LOC_total	√	√	√	√	√	√	√	√	√
	LOC_blank	√	√	√	√	√	√		√	√
	LOC_code_and_comment	√	√	√	√	√	√	√	√	√
	LOC_comments	√	√	√	√	√	√	√	√	√
	LOC_executable	√	√	√	√	√	√	√	√	√
	number_of_lines	√		√	√	√	√	√	√	√
Halstead	content	√	√	√	√	√	√	√	√	√
	difficulty	√	√	√	√	√	√	√	√	√
	effort	√	√	√	√	√	√	√	√	√
	error_est	√	√	√	√	√	√	√	√	√
	length	√	√	√	√	√	√	√	√	√
	level	√	√	√	√	√	√	√	√	√
	prog_time	√	√	√	√	√	√	√	√	√
	volume	√	√	√	√	√	√	√	√	√
	num_operands	√	√	√	√	√	√	√	√	√
	num_operators	√	√	√	√	√	√	√	√	√
	num_unique_operands	√	√	√	√	√	√	√	√	√
	num_unique_operators	√	√	√	√	√	√	√	√	√
McCabe	cyclomatic_complexity	√	√	√	√	√	√	√	√	√
	cyclomatic_density	√		√	√	√	√	√	√	√
	design_complexity	√	√	√	√	√	√	√	√	√
	essential_complexity	√	√	√	√	√	√	√	√	√
Misc.	branch_count	√	√	√	√	√	√	√	√	√
	call_pairs	√		√	√	√	√	√	√	√
	condition_count	√		√	√	√	√	√	√	√
	decision_count	√		√	√	√	√	√	√	√
	decision_density	√		√	√	√	√	√	√	√
	edge_count	√		√	√	√	√	√	√	√
	essential_density	√		√	√	√	√	√	√	√
	parameter_count	√		√	√	√	√	√	√	√
	maintenance_severity	√		√	√	√	√	√	√	√
	modified_condition_count	√		√	√	√	√	√	√	√
	multiple_condition_count	√		√	√	√	√	√	√	√
	global_data_complexity			√	√					
	global_data_density			√	√					
	normalized_cyclo_complx	√		√	√	√	√	√	√	√
	percent_comments	√		√	√	√	√	√	√	√
	node_count	√		√	√	√	√	√	√	√
Programming Language		C	C++	Java	C	C	C	C	C	C
Number of Code Attributes		37	21	39	39	37	37	36	37	37
Number of Modules		344	2096	200	127	264	759	1585	1125	1399
Number of fp Modules		42	325	36	44	27	61	16	140	178
[illegible]		[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]

Code Attribute

```
void main()
{
        //This is a sample code

        //Declare variables
        int a, b, c;

        // Initialize variables
        a=2;
        b=5;

        //Find the sum and display c if greater
than zero
        c=sum(a,b);
        if c < 0                        c > 0
          printf("%d\n", a);
        return;
}                                       c


int sum(int a, int b)
{
        // Returns the sum of two numbers
        return a+b;
}
```

Module	LOC	LOCC	V	CC	**Error**
main()	16	4	5	2	**2**
sum()	5	1	3	1	**0**

LOC: Line of Code
LOCC: Line of commented Code
V: Number of unique operands&operators
CC: Cyclometric Complexity

Code Complexity Measurement

1. Source Lines of Codes
2. Operator and Operand Numbers
   - Halstead
3. Coupling
4. Flow
   - McCabe

McCabe and Halstead

- There are 2 main types of software reliability models:
  1. the deterministic
  2. the probabilistic
- Two well known models of the deterministic type are the Halstead's software metric and the McCabe's cyclomatic complexity metric
  1. Halstead's software metric is used to estimate the number of errors in a program (based on the number of operands and operators in programs)
  2. McCabe's cyclomatic complexity metric (McCabe 1976) is used to determine an upper bound on the model for estimating the number of remaining defects (based on the number of decision points)

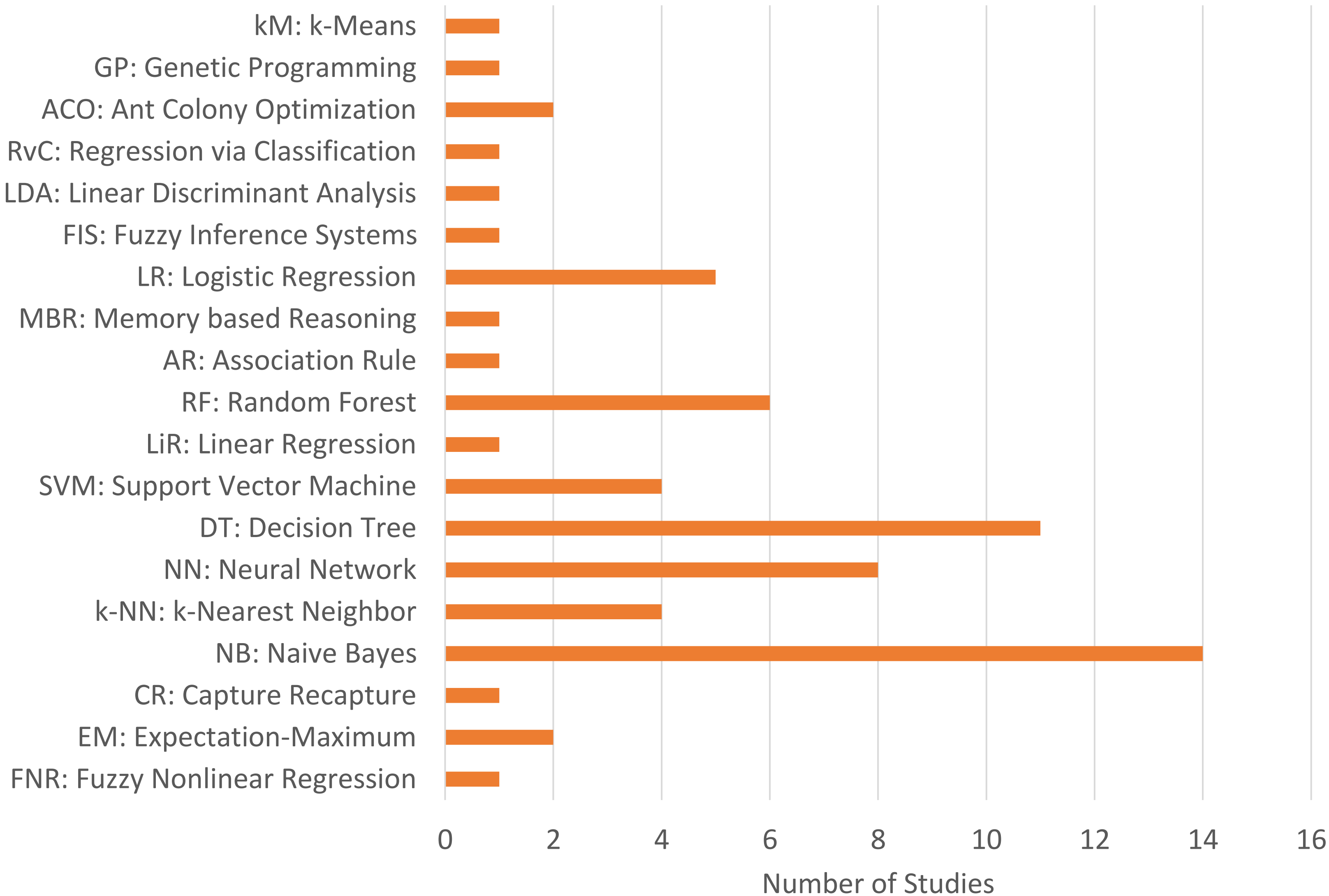
Halstead

- Halstead's theory of software metric is probably the best-known technique to measure complexity in a software program and the amount of difficulty involved in testing and debugging the software
- Halstead (1977) uses the number of distinct operators and the number of distinct operands in a program to develop expressions for the overall program length, volume, and the number of remaining defects in a program

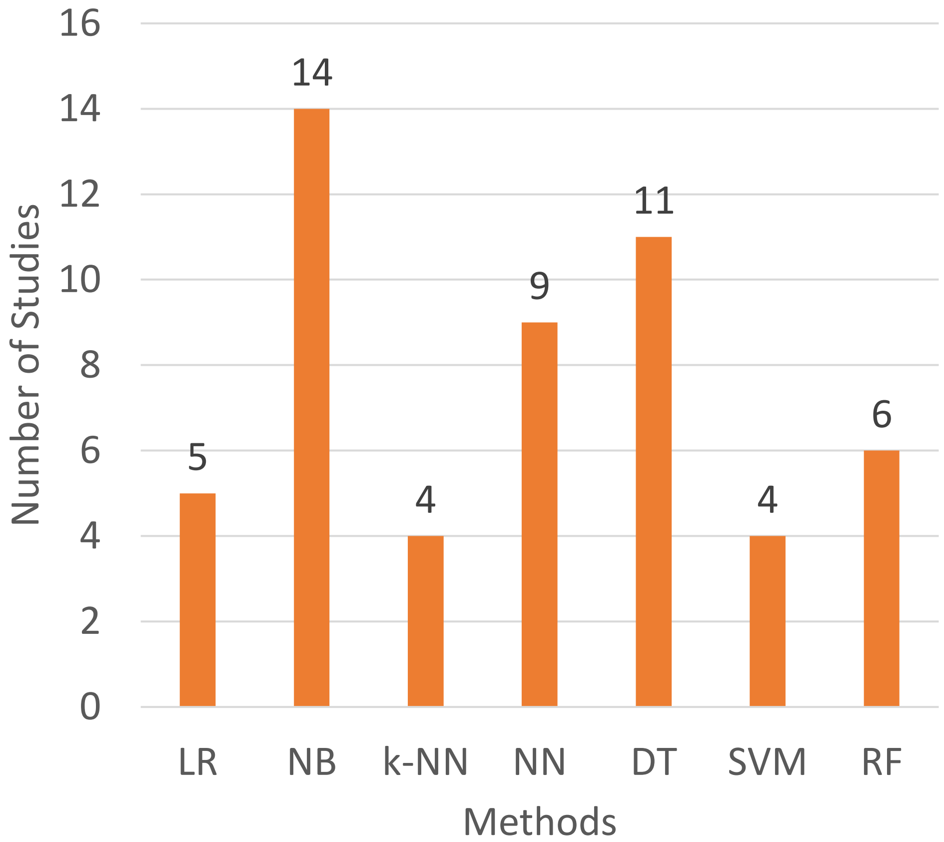
McCabe

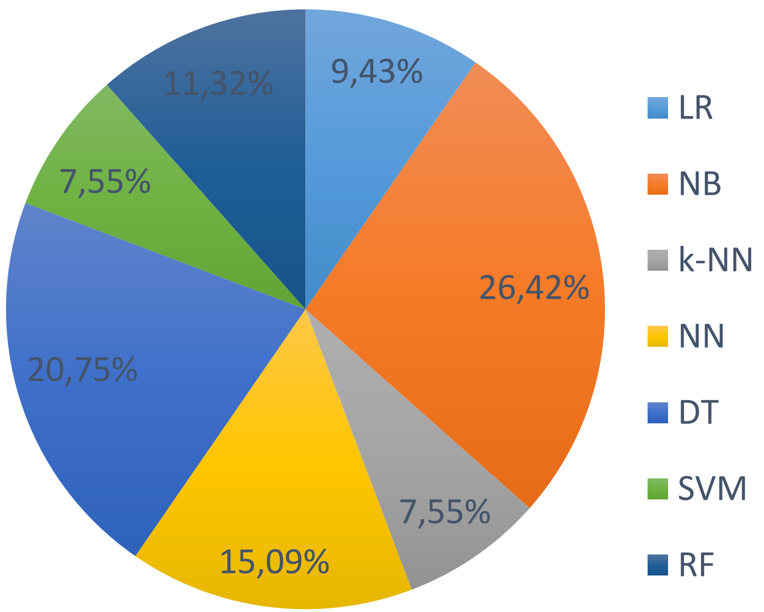
- McCabe developed a metric that allows developers to identify modules that are difficult to test or maintain
- As a result, he developed a software metric that equates complexity to the number of decisions in a program
- Developers can use this measure to determine which modules of a program are overly-complex and need to be re-coded
- The metric can also be used to define the minimal number of test cases that are needed to adequately test the program's paths
- McCabe bases the metric on graph theory

RQ5: Software Defect Prediction Methods



RQ6: Most Used Software Defect Prediction Methods





RQ7: Method Comparison Results

- The comparisons and benchmarking result of the defect prediction using machine learning classifiers indicate that:
  - ✓ Poor accuracy level is dominant (Lessmann et al. 2008)
  - ✓ No significant performance differences could be detected (Lessmann et al. 2008)
  - ✓ No particular classifiers that performs the best for all the data sets (Song et al. 2011) (Hall et al. 2012)

- The accurate and reliable classification algorithms to build a better prediction model is an open issue in software defect prediction

RQ8: Method Improvement Efforts

- Researchers proposed some techniques for improving the accuracy of classifiers for software defect prediction
- Recent proposed techniques try to increase the prediction accuracy of a generated model:
  - ✓ By modifying and ensembling some machine learning methods (Mısırlı et al. 2011) (Tosun et al. 2008)
  - ✓ By using boosting algorithm (Zheng 2010) (Jiang et al. 2011)
  - ✓ by adding feature selection (Gayatri et al. 2010) (Khoshgoftaar & Gao, 2009) (Song et al. 2011)
  - ✓ By using parameter selection for some classifiers (Peng & Wang 2010) (Lin et al. 2008) (Guo et al. 2008)
- While considerable works have been done separately, limited research can be found on investigating them all together

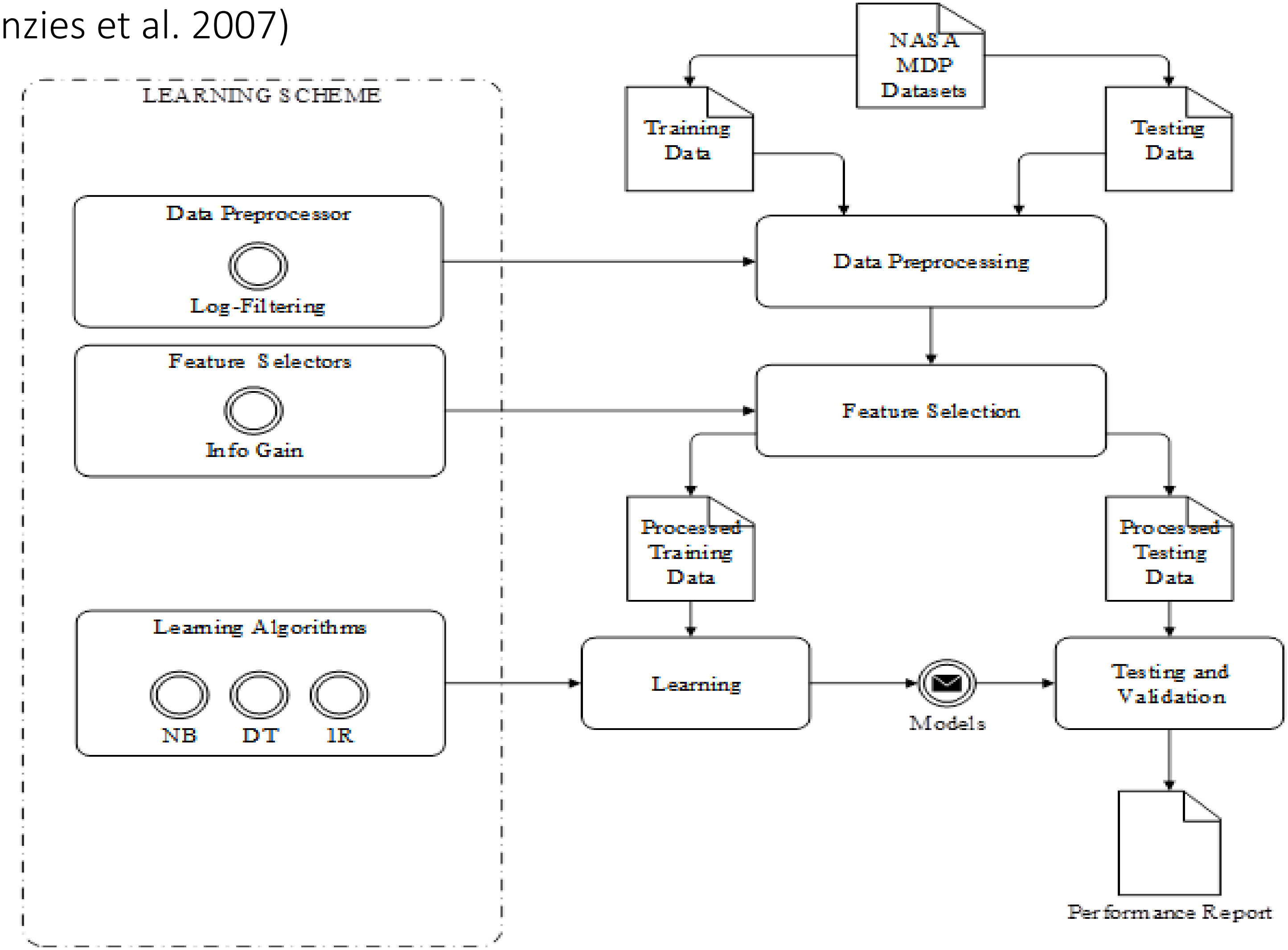
RQ9: Existing Frameworks

Three frameworks have been highly cited and influential in software defect prediction field

Menzies Framework	**Lessmann** Framework	**Song** Framework
(Menzies et al. 2007)	(Lessmann et al. 2008)	(Song et al. 2011)

Menzies Framework

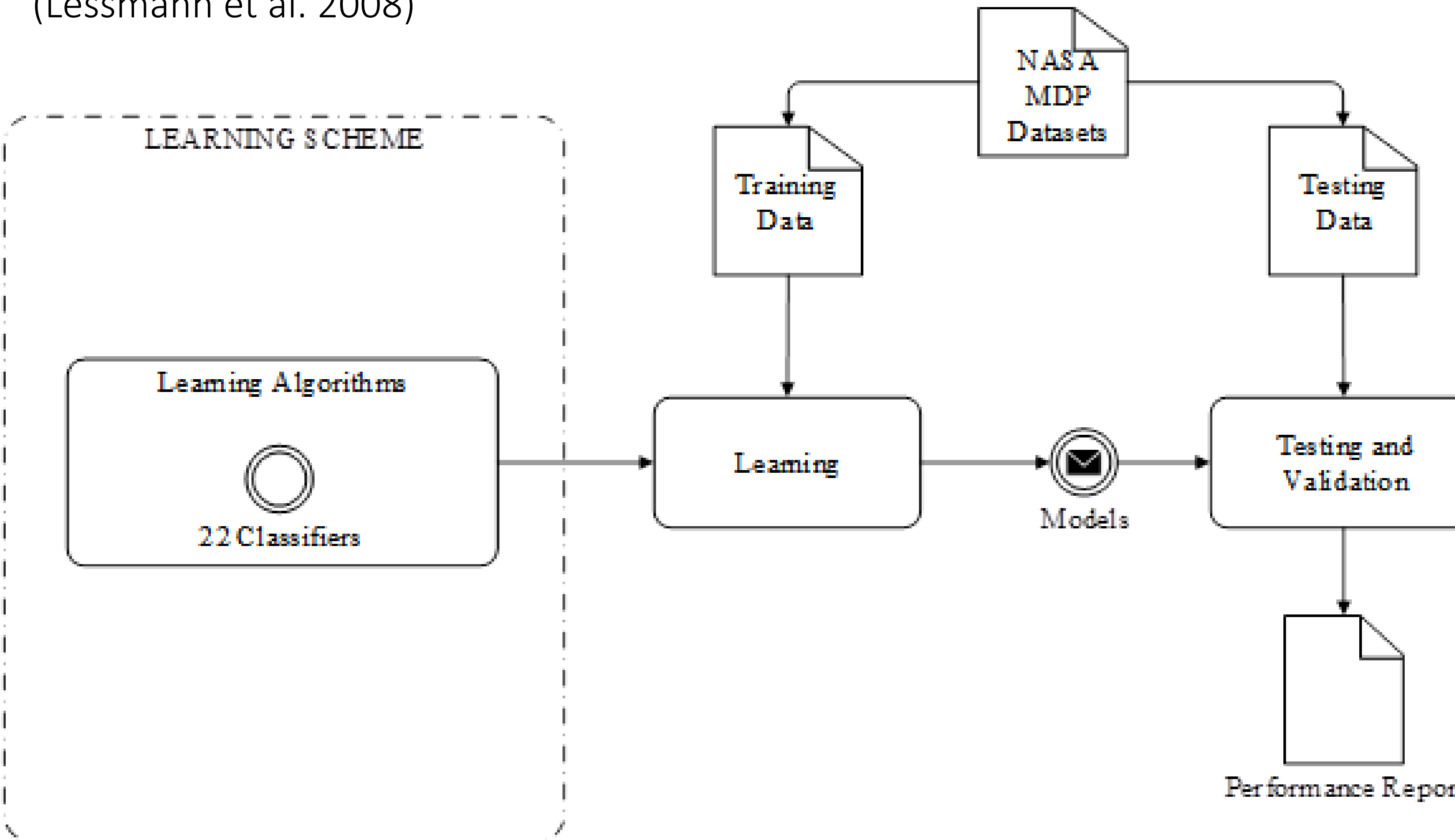
(Menzies et al. 2007)



Framework	Dataset	Data Preprocessor	Feature Selectors	Meta-learning	Classifiers	Parameter Selectors	Validation Methods	Evaluation Methods
(Menzies et al. 2007)	NASA MDP	**Log Filtering**	**Info Gain**	-	**3 algorithms (DT, 1R, NB)**	-	10-Fold X Validation	ROC Curve (AUC)

Lessmann Framework

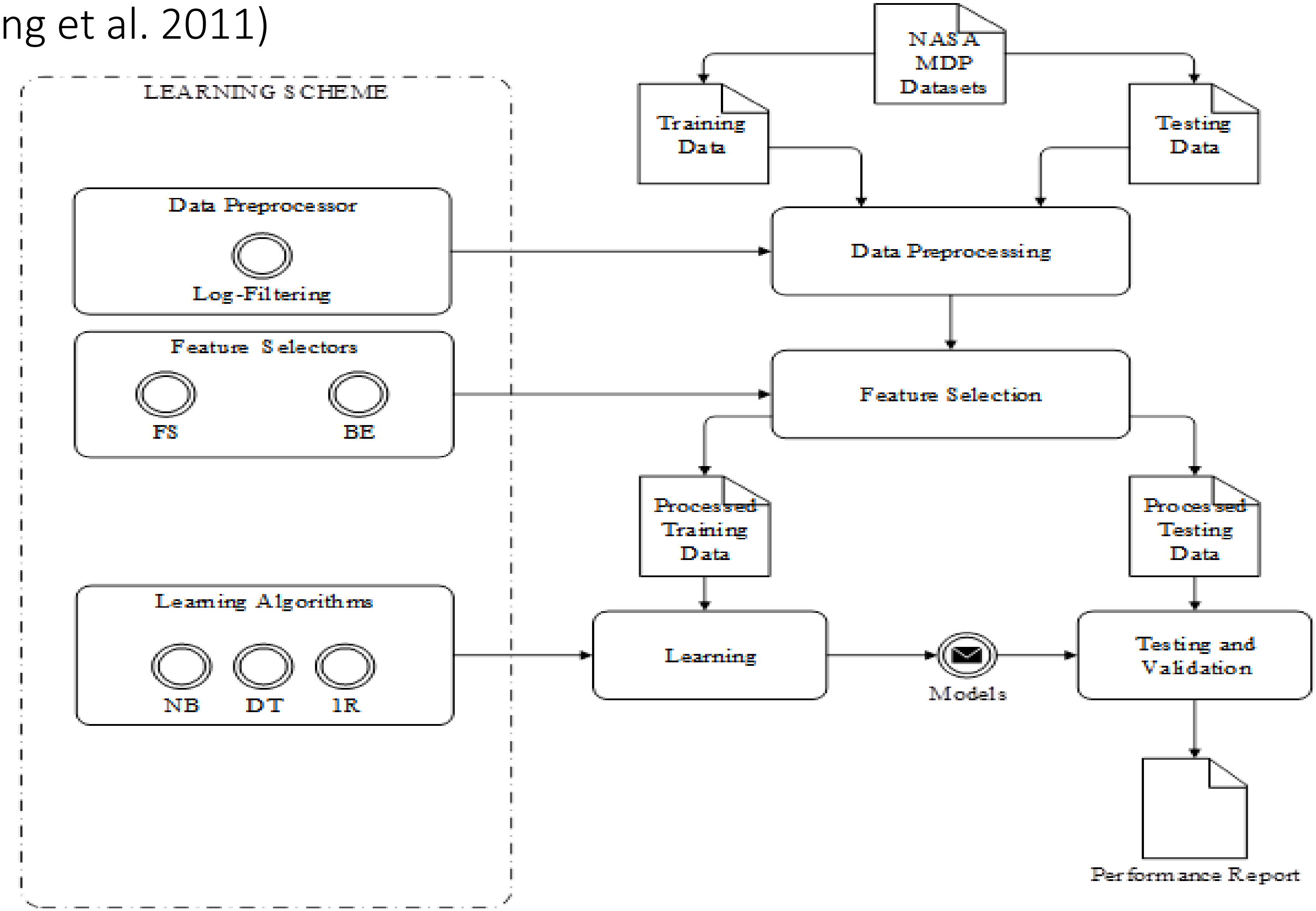
(Lessmann et al. 2008)



Framework	Dataset	Data Preprocessor	Feature Selectors	Meta-learning	Classifiers	Parameter Selectors	Validation Methods	Evaluation Methods
(Lessman et al. 2008)	NASA MDP	-	-	-	**22 algorithms**	-	10-Fold X Validation	ROC Curve (AUC)

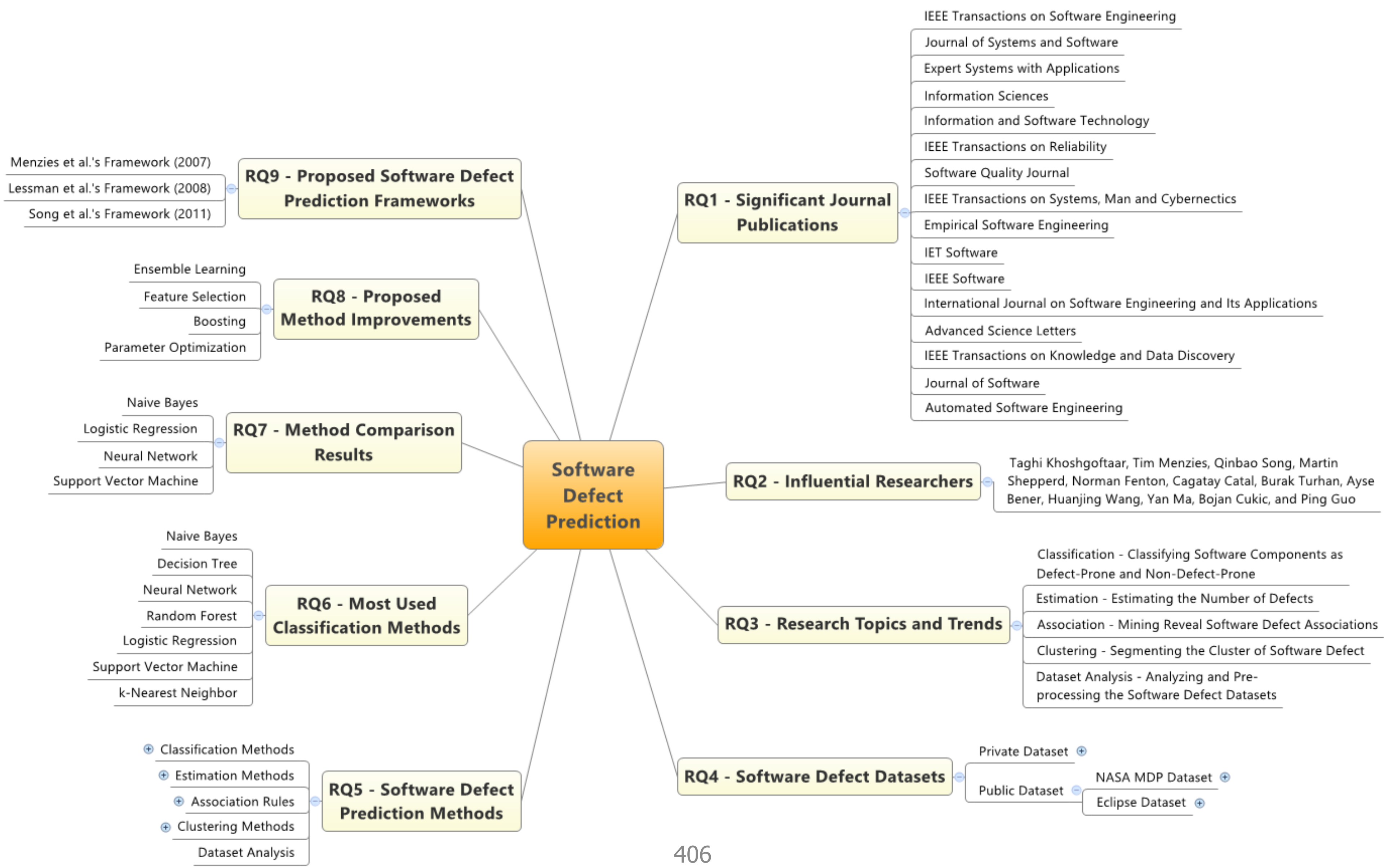
Song Framework

(Song et al. 2011)



Framework	Dataset	Data Preprocessor	Feature Selectors	Meta-learning	Classifiers	Parameter Selectors	Validation Methods	Evaluation Methods
(Song et al. 2011)	NASA MDP	**Log Filtering**	**FS, BE**	-	**3 algorithms (DT, 1R, NB)**	-	10-Fold X Validation	ROC Curve (AUC)

Mind Map of the SLR Results



SLR Melahirkan Research Gaps

Dari Hasil SLR, Kita Menemukan Research Gaps yang Menjadi Kandidat Masalah Penelitian yang Kita Angkat pada Penelitian Kita

Gap Analysis of Framework

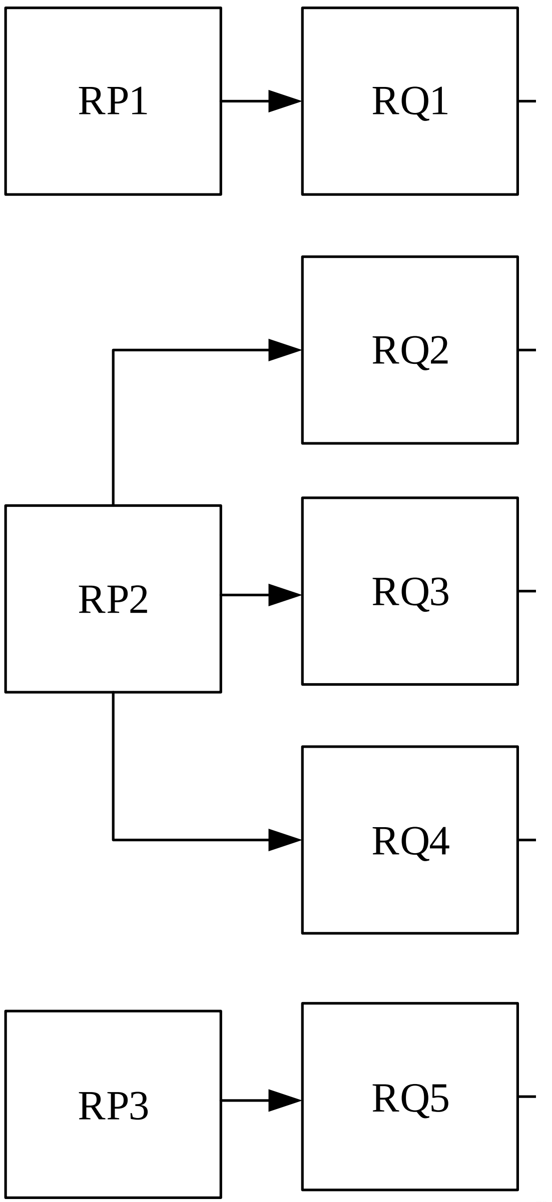
- The comparisons and benchmarking result of the defect prediction using machine learning classifiers indicate that:
  - ✓ Poor accuracy level is dominant (Lessmann et al. 2008)
  - ✓ No significant performance differences could be detected (Lessmann et al. 2008)
  - ✓ No particular classifiers that performs the best for all the data sets (Song et al. 2011) (Hall et al. 2012)
- Noisy attribute predictors and imbalanced class distribution of software defect datasets result in inaccuracy of classification models
- Neural network and support vector machine have strong fault tolerance and strong ability of nonlinear dynamic processing of software fault data, but practicability of neural network and support vector machine are limited due to difficulty of selecting appropriate parameters

Research Problems (RP)

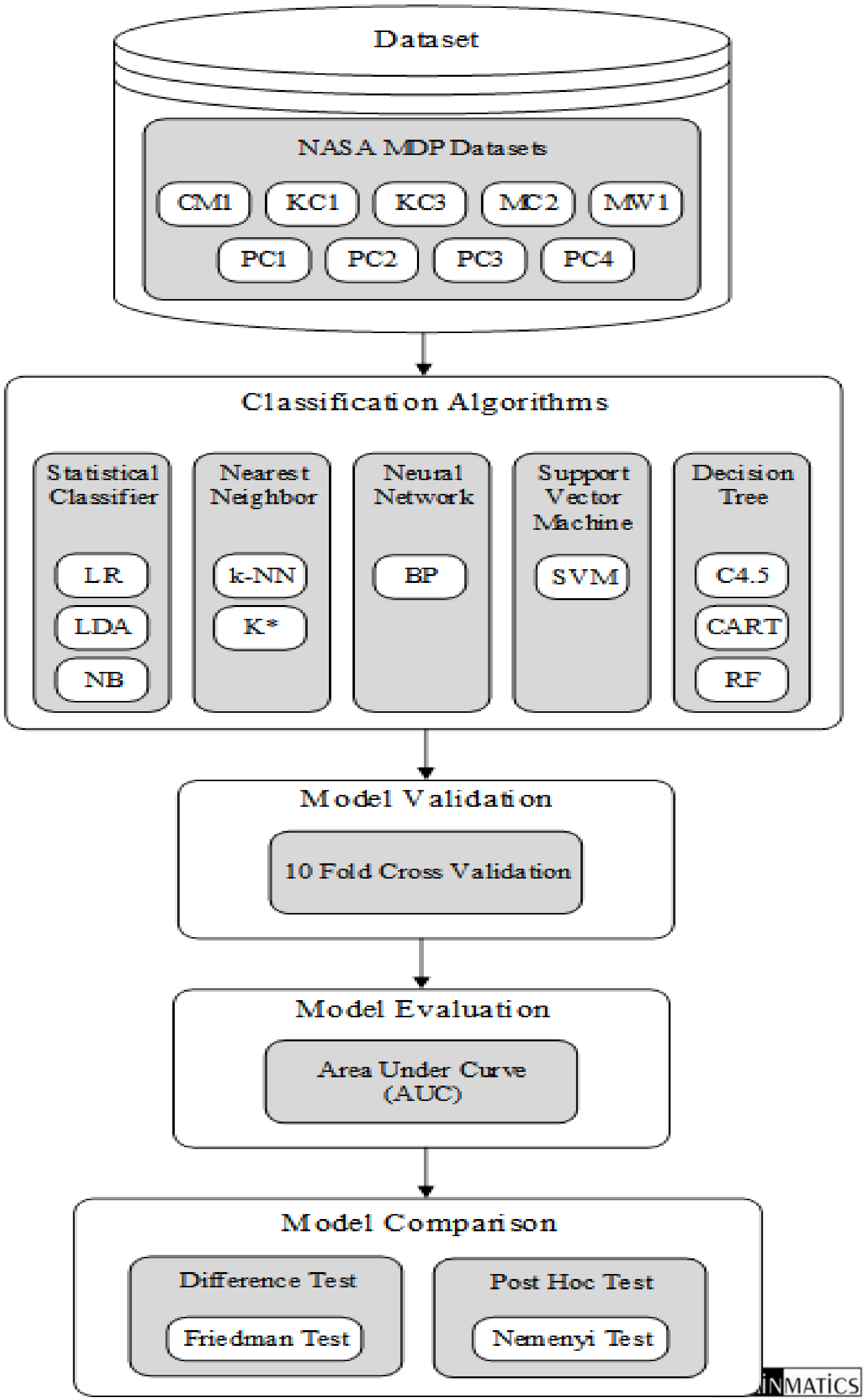
RP1 While many studies on software defect prediction report the comparative performance of the classification algorithms used, but there is no strong consensus on which classifiers perform best when individual studies are looked separately

RP2 Noisy attribute predictors and imbalanced class distribution of software defect datasets result in inaccuracy of classification models

RP3 Neural network has strong fault tolerance and strong ability of nonlinear dynamic processing of software fault data, but practicability of neural network is limited due to difficulty of selecting appropriate parameters



A Comparison Framework of Classification Models for Software Defect Prediction (CF SDP)



AUC and Friedman Test Results

	CM1	KC1	KC3	MC2	MW1	PC1	PC2	PC3	PC4	*M*	*R*
LR	**0.763**	**0.801**	**0.713**	**0.766**	0.726	**0.852**	0.849	**0.81**	0.894	**0.797**	**1.44**
LDA	0.471	0.536	0.447	0.503	0.58	0.454	0.577	0.524	0.61	0.522	8.33
NB	0.734	0.786	0.67	0.739	0.732	0.781	0.811	0.756	0.838	0.761	3
k-NN	0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5	8.778
K*	0.6	0.678	0.562	0.585	0.63	0.652	0.754	0.697	0.76	0.658	5.33
BP	0.713	0.791	0.647	0.71	0.625	0.784	0.918	0.79	0.883	0.762	3.22
SVM	0.753	0.752	0.642	0.761	0.714	0.79	0.534	0.75	**0.899**	0.733	3.33
C4.5	0.565	0.515	0.497	0.455	0.543	0.601	0.493	0.715	0.723	0.567	7.78
CART	0.604	0.648	0.637	0.482	0.656	0.574	0.491	0.68	0.623	0.599	6.89
RF	0.573	0.485	0.477	0.525	**0.74**	0.618	0.649	0.678	0.2	0.549	6.89

- LR is dominant in most datasets
- *R* rank: LR has the highest rank, followed by NB, BP, and SVM
- *M* results: no excellent or good models, and a few fair models

AUC	Meaning	Symbol
0.90 - 1.00	excellent classification	
0.80 - 0.90	good classification	
0.70 - 0.80	fair classification	
0.60 - 0.70	poor classification	
< 0.60	failure	

P-value of Nemenyi Post Hoc Test

	LR	LDA	NB	k-NN	K*	BP	SVM	C4.5	CART	RF
LR	1	**0.0001**	0.986	**0.0001**	0.164	0.965	0.949	**0.000**	**0.005**	**0.005**
LDA	**0.0001**	1	**0.007**	1.000	0.526	**0.013**	**0.017**	1.000	0.992	0.992
NB	0.986	**0.007**	1	**0.002**	0.831	1.000	1.000	**0.028**	0.164	0.164
k-NN	**0.0001**	1.000	**0.002**	1	0.318	**0.004**	**0.005**	1.000	0.949	0.949
K*	0.164	0.526	0.831	0.318	1	0.901	0.927	0.789	0.986	0.986
BP	0.965	**0.013**	1.000	**0.004**	0.901	1	1.000	**0.046**	0.232	0.232
SVM	0.949	**0.017**	1.000	**0.005**	0.927	1.000	1	0.058	0.273	0.273
C4.5	**0.000**	1.000	**0.028**	1.000	0.789	**0.046**	0.058	1	1.000	1.000
CART	**0.005**	0.992	0.164	0.949	0.986	0.232	0.273	1.000	1	1.000
RF	**0.005**	0.992	0.164	0.949	0.986	0.232	0.273	1.000	1.000	1

- If *P value < 0.05* (boldfaced print), it indicate that there is significant different between two classifiers
- Based on significant difference results, there is no significant difference between LR, NB, BP, and SVM models

Research Publication on RQ1

Romi Satria Wahono, Nanna Suryana Herman and Sabrina Ahmad, A Comparison Framework of Classification Models for Software Defect Prediction, Advanced Science Letters, Vol. 20, No. 8, August 2014

Research Result on RQ3

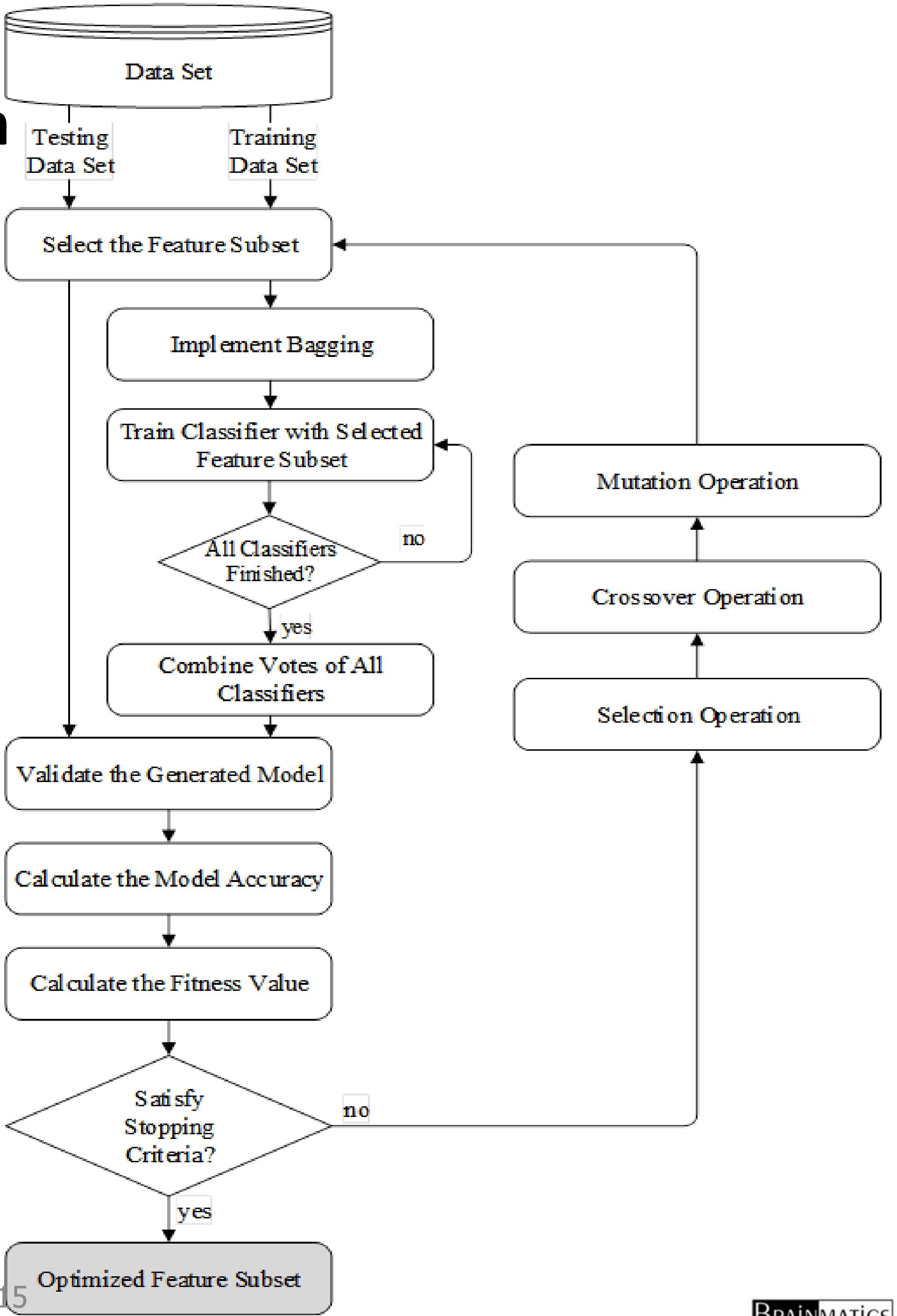
Research Problems (RP)		Research Questions (RQ)		Research Objectives (RO)	
RP2	Noisy attribute predictors and imbalanced class distribution of software defect datasets result in inaccuracy of classification models	RQ2	How does the integration between genetic algorithm based feature selection and bagging technique affect the accuracy of software defect prediction?	RO2	To develop a hybrid genetic algorithm based feature selection and bagging technique for improving the accuracy of software defect prediction
		RQ3	**How does the integration between particle swarm optimization based feature selection and bagging technique affect the accuracy of software defect prediction?**	**RO3**	**To develop a hybrid particle swarm optimization based feature selection and bagging technique for improving the accuracy of software defect prediction**
		RQ4	Which metaheuristic optimization techniques perform best when used in feature selection of software defect prediction?	RO4	To identify the best metaheuristic optimization techniques when used in feature selection of software defect prediction

A Hybrid Genetic Algorithm based Feature Selection and Bagging Technique (GAFS+B)

- Every chromosome is evaluated by the fitness function Equation

$$fitness = W_A \times A + W_F \times \left(P + \left(\sum_{i=1}^{n_f} C_i \times F_i \right) \right)^{-1}$$

- Where
  - A: classification accuracy
  - F_i: feature value
  - W_A: weight of classification accuracy
  - W_F: feature weight
  - C_i: feature cost
- When ending condition is satisfied, the operation ends, otherwise, continue with the next genetic operation



Results: Without GAFS+B

Classifiers		CM1	KC1	KC3	MC2	MW1	PC1	PC2	PC3	PC4
Statistical Classifier	LR	0.763	0.801	0.713	0.766	0.726	0.852	0.849	0.81	0.894
	LDA	0.471	0.536	0.447	0.503	0.58	0.454	0.577	0.524	0.61
	NB	0.734	0.786	0.67	0.739	0.732	0.781	0.811	0.756	0.838
Nearest Neighbor	k-NN	0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5	0.5
	K*	0.6	0.678	0.562	0.585	0.63	0.652	0.754	0.697	0.76
Neural Network	BP	0.713	0.791	0.647	0.71	0.625	0.784	0.918	0.79	0.883
Support Vector Machine	SVM	0.753	0.752	0.642	0.761	0.714	0.79	0.534	0.75	0.899
Decision Tree	C4.5	0.565	0.515	0.497	0.455	0.543	0.601	0.493	0.715	0.723
	CART	0.604	0.648	0.637	0.482	0.656	0.574	0.491	0.68	0.623
	RF	0.573	0.485	0.477	0.525	0.74	0.618	0.649	0.678	0.2

Results: With GAFS+B

Classifiers		CM1	KC1	KC3	MC2	MW1	PC1	PC2	PC3	PC4
Statistical Classifier	**LR**	0.753	0.795	0.691	0.761	**0.742**	0.852	0.822	0.813	**0.901**
	LDA	**0.592**	**0.627**	**0.635**	**0.64**	**0.674**	**0.637**	**0.607**	**0.635**	**0.715**
	NB	0.702	**0.79**	**0.677**	0.739	0.724	**0.799**	0.805	**0.78**	**0.861**
Nearest Neighbor	**k-NN**	**0.666**	**0.689**	**0.67**	**0.783**	**0.656**	**0.734**	**0.554**	**0.649**	**0.732**
	K*	**0.71**	**0.822**	0.503	**0.718**	**0.68**	**0.876**	**0.877**	**0.816**	**0.893**
Neural Network	**BP**	**0.744**	**0.797**	**0.707**	**0.835**	**0.689**	**0.829**	0.905	**0.799**	**0.921**
Support Vector Machine	**SVM**	0.667	**0.767**	0.572	**0.747**	**0.659**	0.774	0.139	0.476	**0.879**
Decision Tree	**C4.5**	**0.64**	**0.618**	**0.658**	**0.732**	**0.695**	**0.758**	**0.642**	**0.73**	**0.844**
	CART	**0.674**	**0.818**	**0.754**	**0.709**	**0.703**	**0.819**	**0.832**	**0.842**	**0.9**
	RF	**0.706**	**0.584**	**0.605**	0.483	0.735	**0.696**	**0.901**	**0.734**	**0.601**

- Almost all classifiers that implemented GAFS+B method outperform the original method
- GAFS+B affected significantly on the performance of the class imbalance suffered classifiers

Without GAFS+B vs With GAFS+B

Classifiers		*P* value of t-Test	Result
Statistical Classifier	LR	0.156	Not Sig. ($\alpha > 0.05$)
	LDA	**0.00004**	**Sig. ($\alpha < 0.05$)**
	NB	0.294	Not Sig. ($\alpha > 0.05$)
Nearest Neighbor	**k-NN**	**0.00002**	**Sig. ($\alpha < 0.05$)**
	K*	**0.001**	**Sig. ($\alpha < 0.05$)**
Neural Network	**BP**	**0.008**	**Sig. ($\alpha < 0.05$)**
Support Vector Machine	**SVM**	**0.03**	**Sig. ($\alpha < 0.05$)**
Decision Tree	**C4.5**	**0.0002**	**Sig. ($\alpha < 0.05$)**
	CART	**0.0002**	**Sig. ($\alpha < 0.05$)**
	RF	**0.01**	**Sig. ($\alpha < 0.05$)**

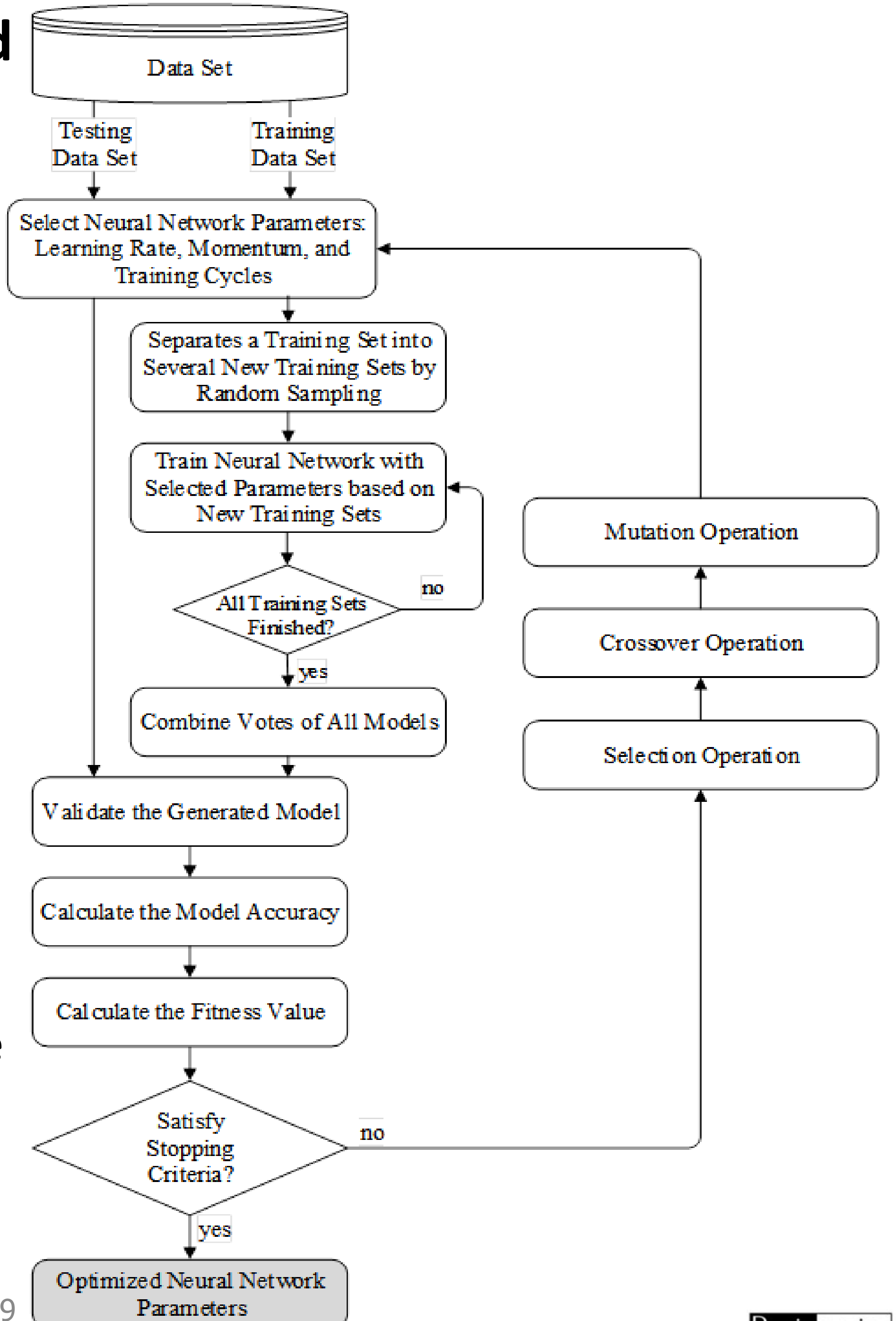
- Although there are two classifiers (LR and NB) that have no significant difference (*P value* > 0.05), the remaining eight classifiers (LDA, k-NN, K*, BP, SVM, C4.5, CART and RF) have significant difference (*P value* < 0.05)
- The proposed GAFS+B method makes an improvement in prediction performance for most classifiers

A Hybrid Genetic Algorithm based Neural Network Parameter Optimization and Bagging Technique for Software Defect Prediction (NN GAPO+B)

- Every chromosome is evaluated by the fitness function Equation

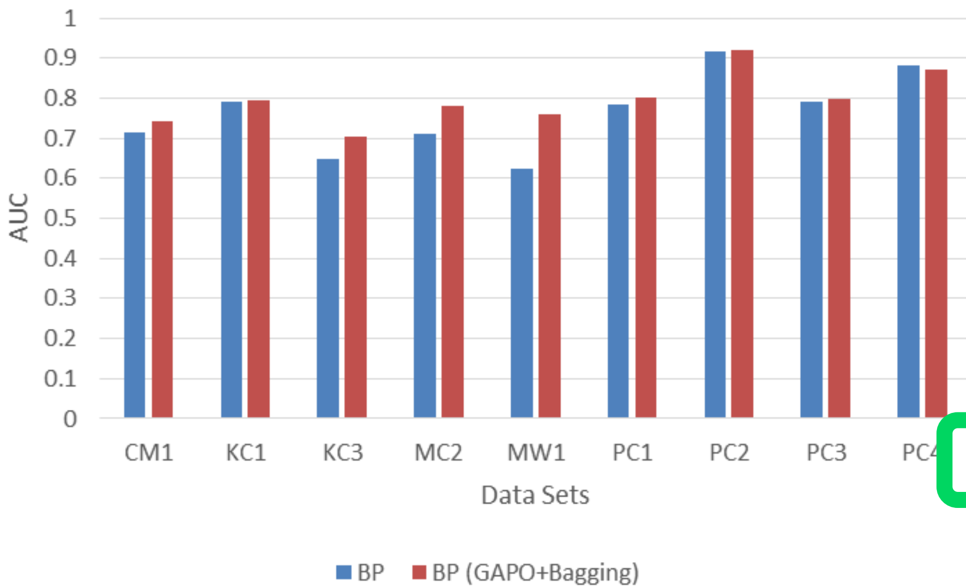
$$fitness = W_A \times A + W_P \times \left(S + \left(\sum_{i=1}^{n} C_i \times P_i \right) \right)^{-1}$$

- Where
  - A: classification accuracy
  - P_i: parameter value
  - W_A: weight of classification accuracy
  - W_p: parameter weight
  - C_i: feature cost
  - S: setting constant
- When ending condition is satisfied, the operation ends and the optimized NN parameters are produced. Otherwise, the process will continue with the next generation operation



Results: NN GAPO+B

Classifiers	CM1	KC1	KC3	MC2	MW1	PC1	PC2	PC3	PC4
NN	0.713	0.791	0.647	0.71	0.625	0.784	0.918	0.79	0.883
NN GAPO+B	**0.744**	**0.794**	**0.703**	**0.779**	**0.76**	**0.801**	**0.92**	**0.798**	0.871

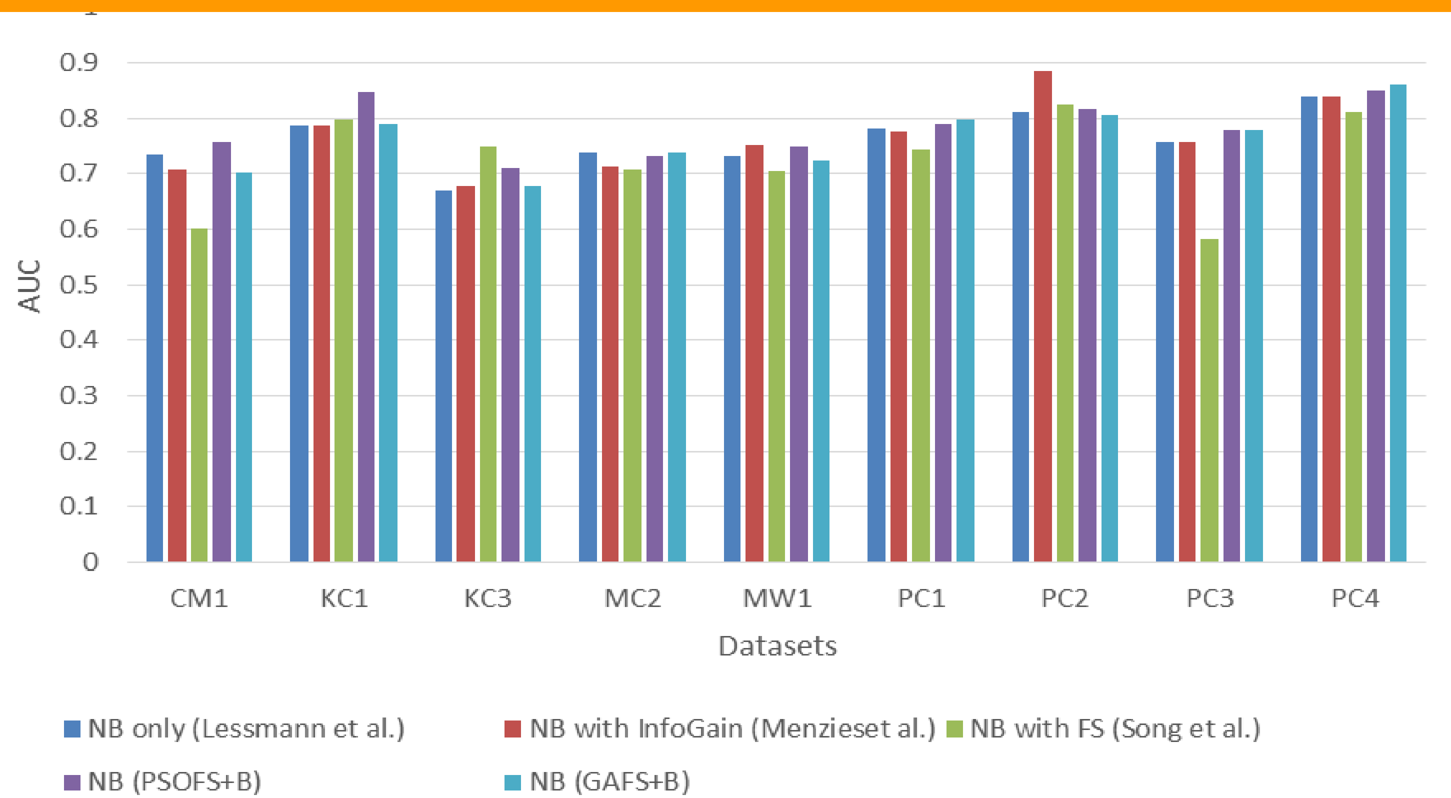


	Variable 1	Variable 2
Mean	0.762333333	0.796666667
Variance	0.009773	0.004246
Observations	9	9
Pearson Correlation	0.923351408	
Hypothesized Mean Difference	0	
df	8	
t Stat	-2.235435933	
P	**0.02791077**	

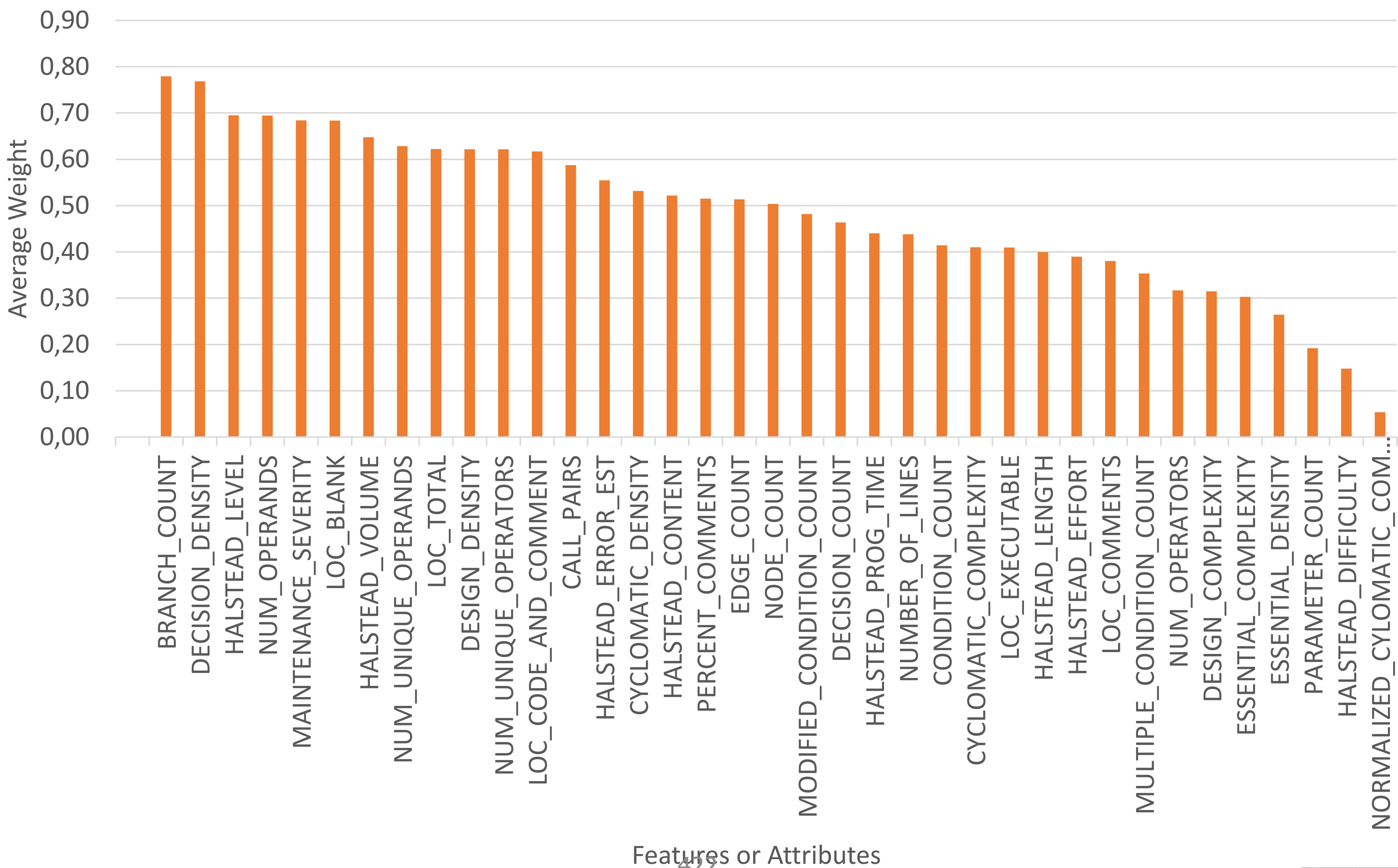
- NN GAPO+B outperforms the original method in almost all datasets
- The proposed (NN GAPO+B) method makes an improvement in prediction performance for back propagation neural network (*P*<0.05)

Framework Comparison

	CM1	KC1	KC3	MC2	MW1	PC1	PC2	PC3	PC4
NB only (Lessmann et al.)	0.734	0.786	0.67	**0.739**	0.732	0.781	0.811	0.756	0.838
NB with InfoGain (Menzies et al.)	0.708	0.786	0.677	0.712	**0.752**	0.775	**0.885**	0.756	0.84
NB with FS (Song et al.)	0.601	0.799	**0.749**	0.707	0.704	0.742	0.824	0.583	0.812
NB (PSOFS+B)	**0.756**	**0.847**	0.71	0.732	0.748	0.79	0.818	**0.78**	0.85
NB (GAFS+B)	0.702	0.79	0.677	**0.739**	0.724	**0.799**	0.805	**0.78**	**0.861**

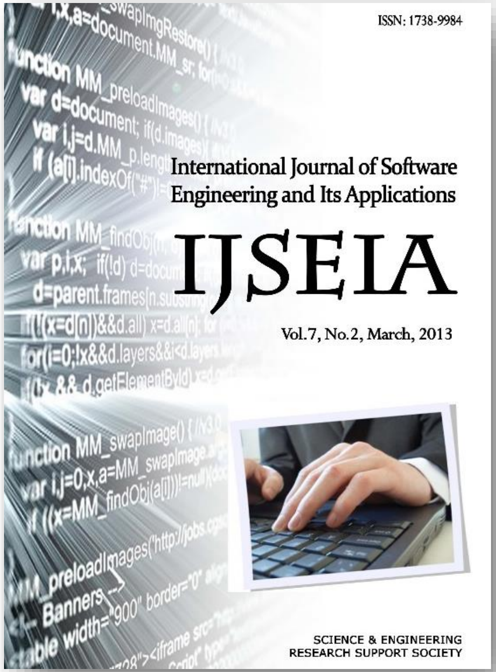


Weighted Average of Relevant Attributes



Research Publication on RQ3

Romi Satria Wahono and Nanna Suryana, *Combining Particle Swarm Optimization based Feature Selection and Bagging Technique for Software Defect Prediction*, International Journal of Software Engineering and Its Applications, Vol 7, No 5, September 2013



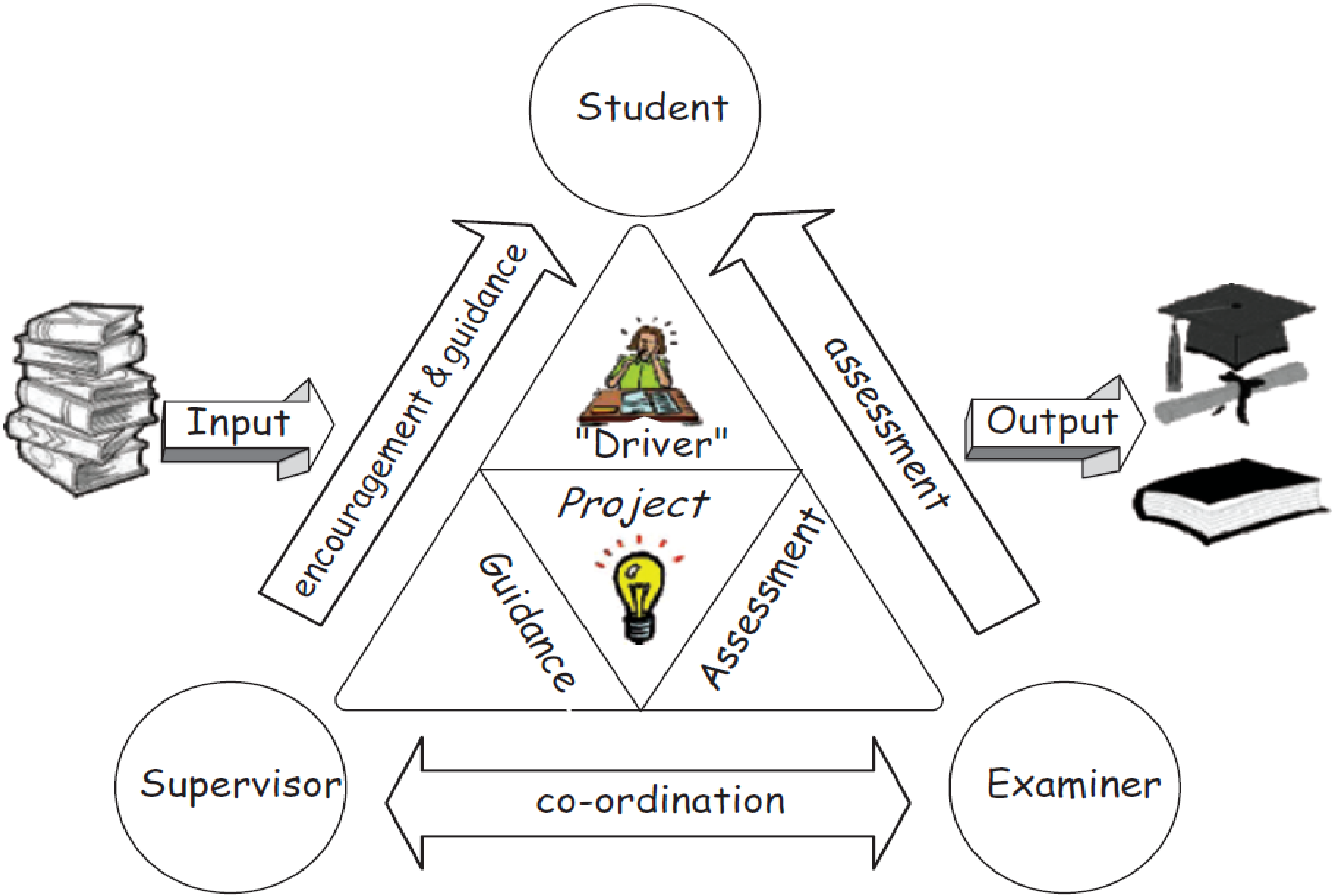
5. Pembimbingan dan Presentasi Penelitian

5.1 Pembimbingan Penelitian

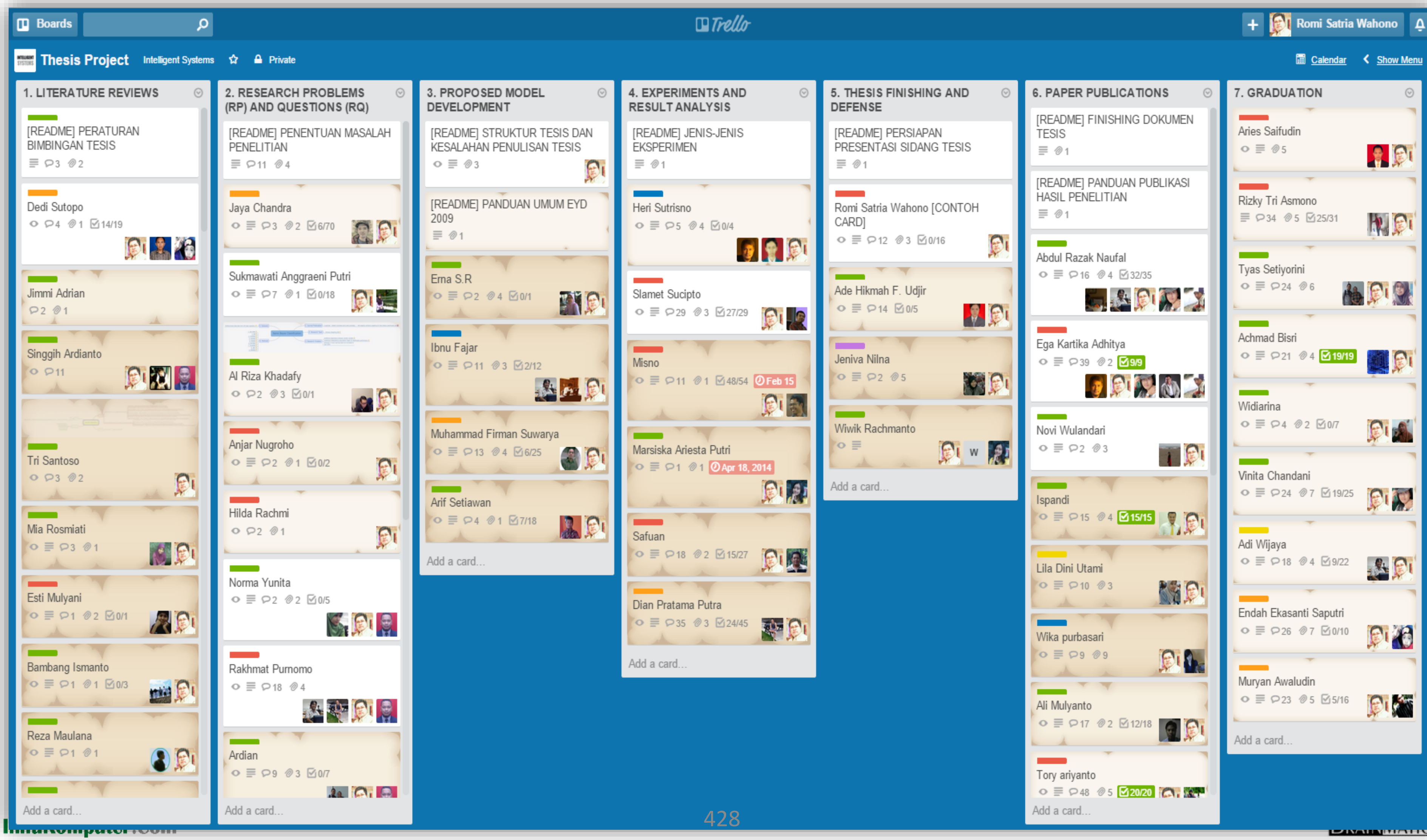
Aktor dalam Penelitian Tesis

1. The student, who identifies, approaches and solves a problem
2. The supervisor, who guides you in your work
3. The examiner, who critically assesses your work

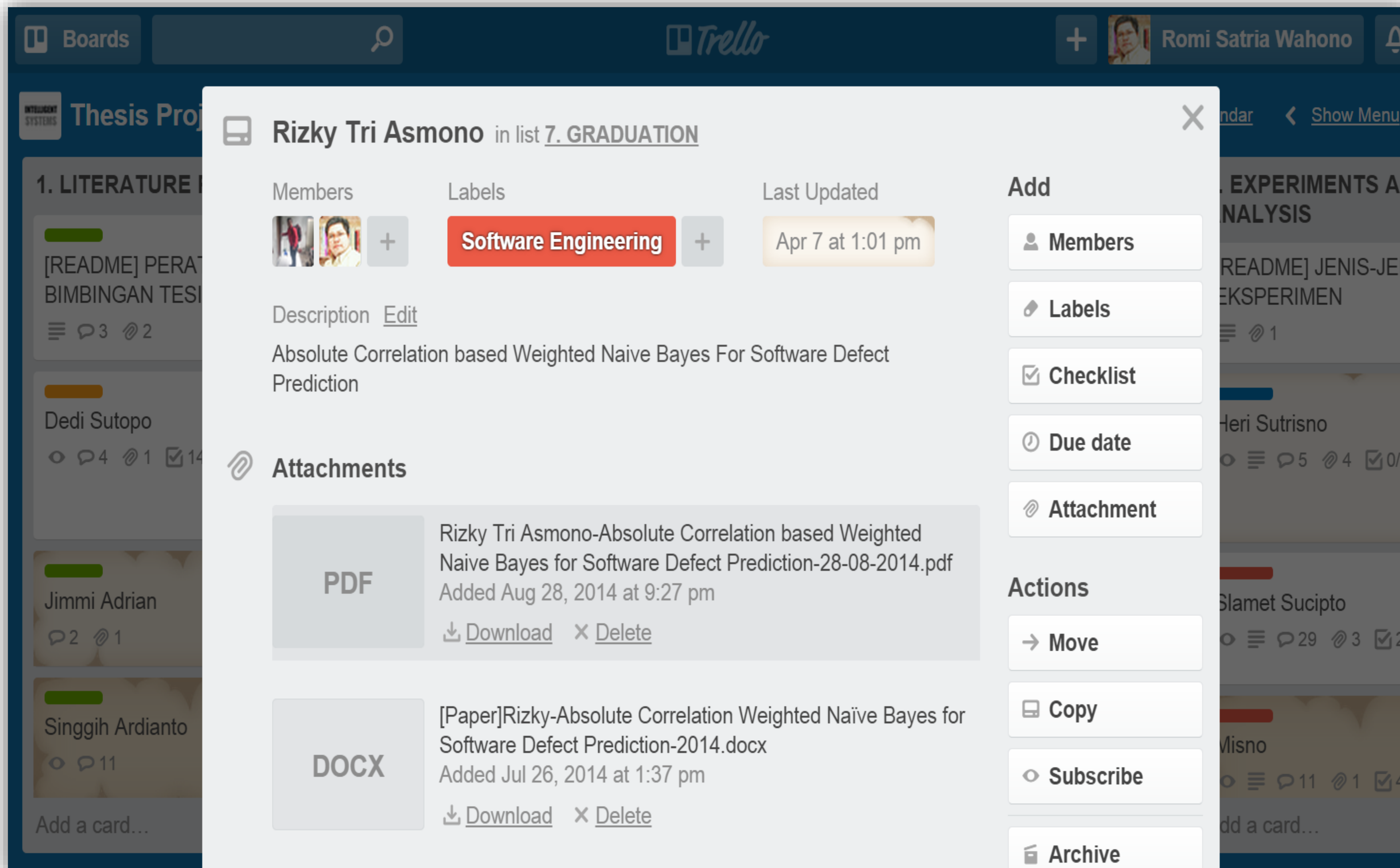
Student – Supervisor – Examiner



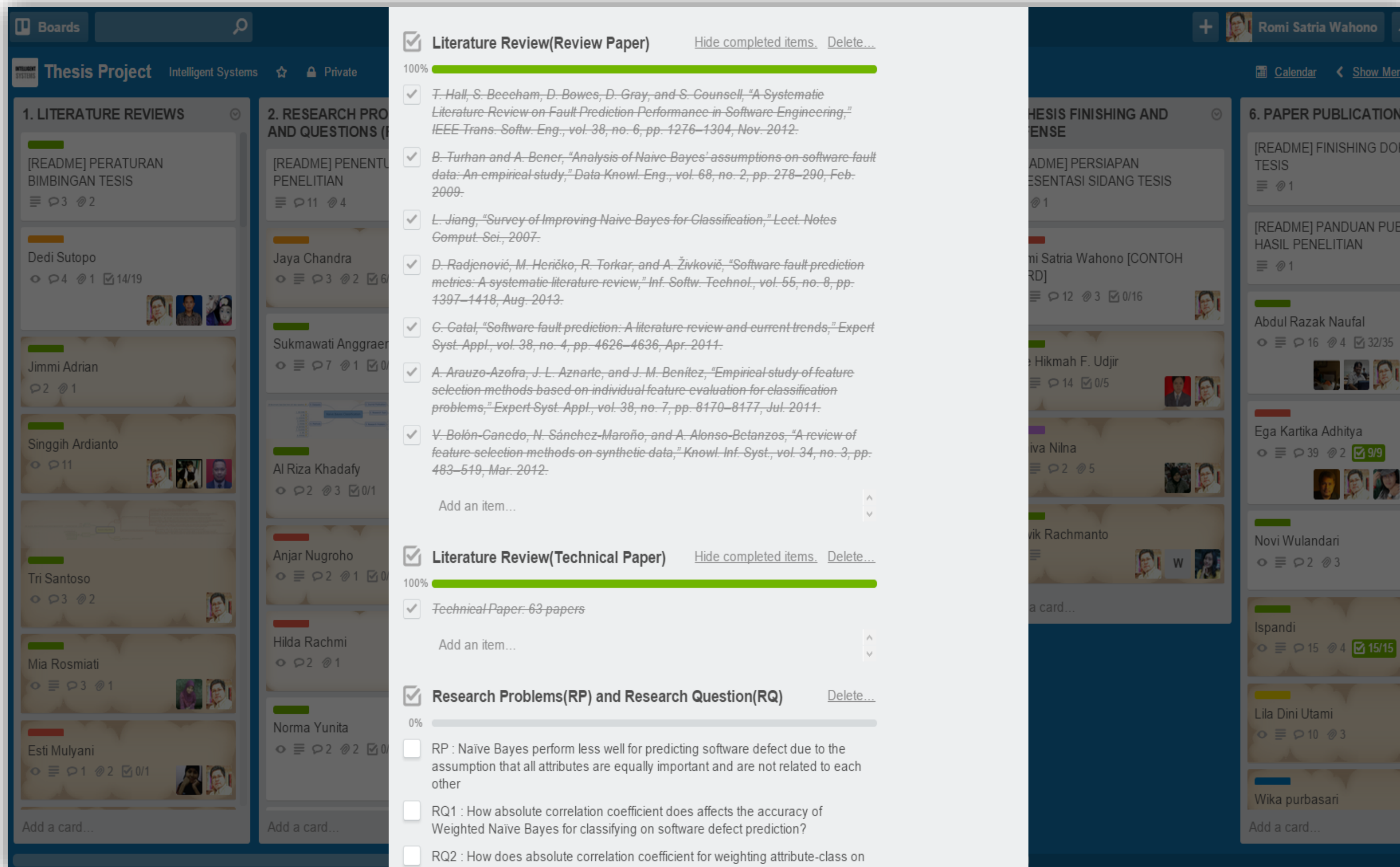
Tools Pengelolaan Penelitian dengan Trello



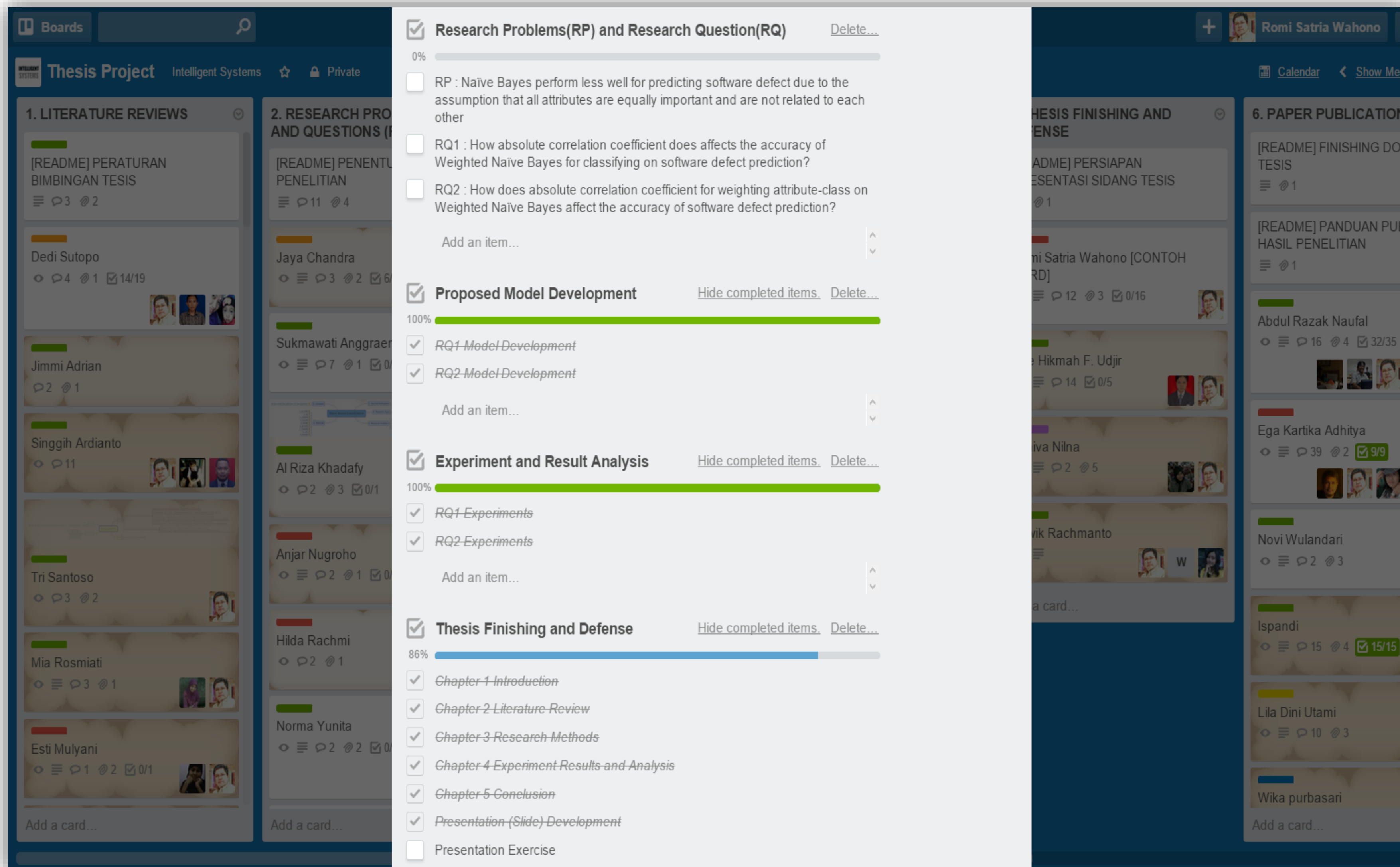
Contoh Card


Boards
Trello
Romi Satria Wahono
Thesis Proj
Show Menu
1. LITERATURE
[README] PERAT
BIMBINGAN TESI
Dedi Sutopo
Jimmi Adrian
Singgih Ardianto
Add a card...
Rizky Tri Asmono in list 7. GRADUATION
Members
Labels
Software Engineering
Last Updated
Apr 7 at 1:01 pm
Description Edit
Absolute Correlation based Weighted Naive Bayes For Software Defect Prediction
Attachments
PDF
Rizky Tri Asmono-Absolute Correlation based Weighted Naive Bayes for Software Defect Prediction-28-08-2014.pdf
Added Aug 28, 2014 at 9:27 pm
Download
Delete
DOCX
[Paper]Rizky-Absolute Correlation Weighted Naïve Bayes for Software Defect Prediction-2014.docx
Added Jul 26, 2014 at 1:37 pm
Download
Delete
Add
Members
Labels
Checklist
Due date
Attachment
Actions
Move
Copy
Subscribe
Archive
EXPERIMENTS AN
NALYSIS
README] JENIS-JEN
EKSPERIMEN
Heri Sutrisno
Slamet Sucipto
Misno

Contoh Card

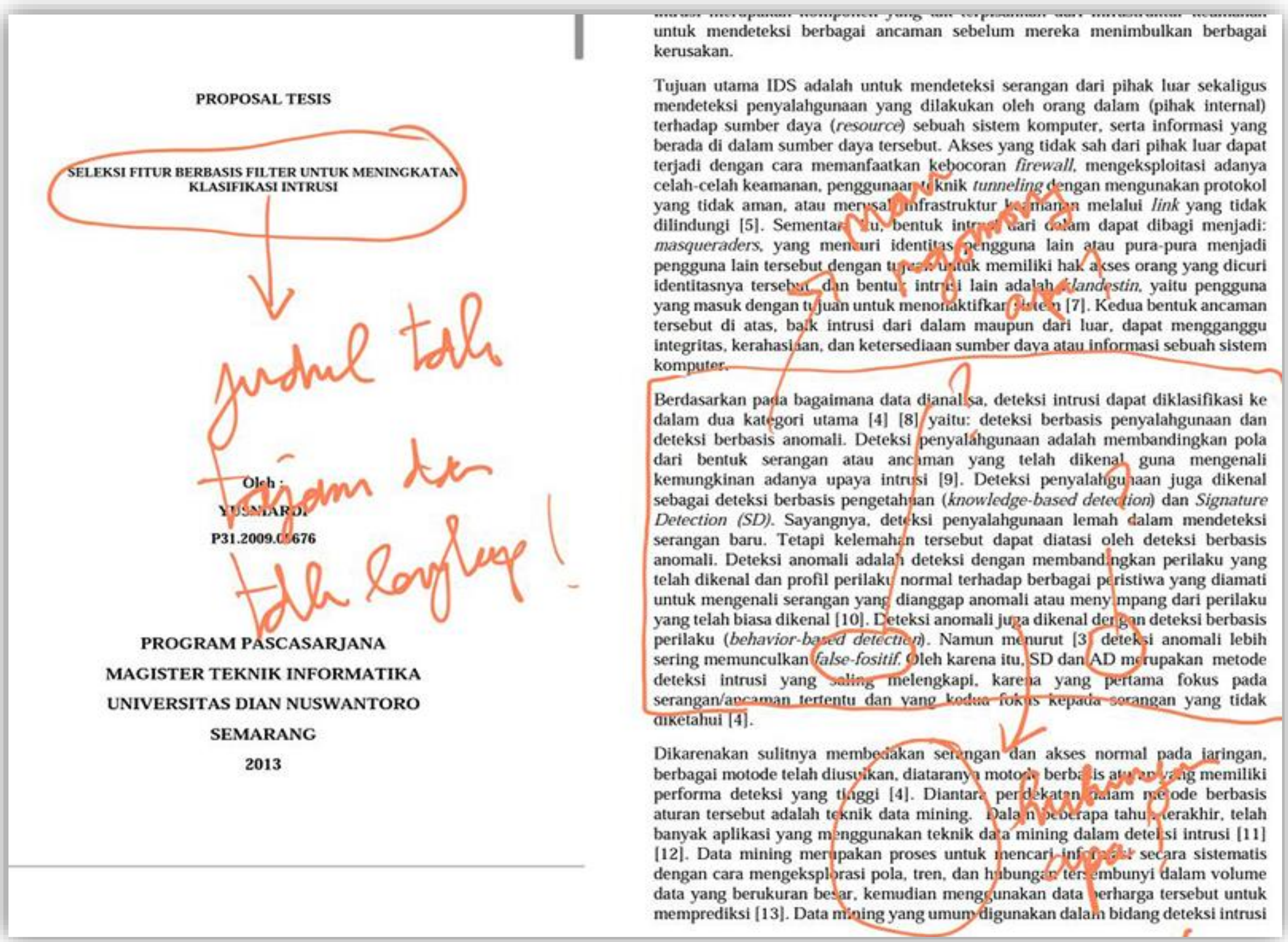

Boards
Thesis Project
Intelligent Systems
Private
Romi Satria Wahono
Calendar
Show Menu
1. LITERATURE REVIEWS
[README] PERATURAN BIMBINGAN TESIS
Dedi Sutopo
Jimmi Adrian
Singgih Ardianto
Tri Santoso
Mia Rosmiati
Esti Mulyani
Add a card...
2. RESEARCH PRO AND QUESTIONS (I
[README] PENENTU PENELITIAN
Jaya Chandra
Sukmawati Anggraer
Al Riza Khadafy
Anjar Nugroho
Hilda Rachmi
Norma Yunita
Add a card...
Literature Review(Review Paper)
Hide completed items.
Delete...
100%
T. Hall, S. Beecham, D. Bowes, D. Gray, and S. Counsell, "A Systematic Literature Review on Fault Prediction Performance in Software Engineering," IEEE Trans. Softw. Eng., vol. 38, no. 6, pp. 1276–1304, Nov. 2012.
B. Turhan and A. Bener, "Analysis of Naive Bayes' assumptions on software fault data: An empirical study," Data Knowl. Eng., vol. 68, no. 2, pp. 278–290, Feb. 2009.
L. Jiang, "Survey of Improving Naive Bayes for Classification," Lect. Notes Comput. Sci., 2007.
D. Radjenović, M. Heričko, R. Torkar, and A. Živkovič, "Software fault prediction metrics: A systematic literature review," Inf. Softw. Technol., vol. 55, no. 8, pp. 1397–1418, Aug. 2013.
C. Catal, "Software fault prediction: A literature review and current trends," Expert Syst. Appl., vol. 38, no. 4, pp. 4626–4636, Apr. 2011.
A. Arauzo-Azofra, J. L. Aznarte, and J. M. Benítez, "Empirical study of feature selection methods based on individual feature evaluation for classification problems," Expert Syst. Appl., vol. 38, no. 7, pp. 8170–8177, Jul. 2011.
V. Bolón-Canedo, N. Sánchez-Maroño, and A. Alonso-Betanzos, "A review of feature selection methods on synthetic data," Knowl. Inf. Syst., vol. 34, no. 3, pp. 483–519, Mar. 2012.
Add an item...
Literature Review(Technical Paper)
Hide completed items.
Delete...
100%
Technical Paper: 63 papers
Add an item...
Research Problems(RP) and Research Question(RQ)
Delete...
0%
RP : Naïve Bayes perform less well for predicting software defect due to the assumption that all attributes are equally important and are not related to each other
RQ1 : How absolute correlation coefficient does affects the accuracy of Weighted Naïve Bayes for classifying on software defect prediction?
RQ2 : How does absolute correlation coefficient for weighting attribute-class on
HESIS FINISHING AND ENSE
ADME] PERSIAPAN ESENTASI SIDANG TESIS
mi Satria Wahono [CONTOH RD]
Hikmah F. Udjir
iva Nilna
ik Rachmanto
a card...
6. PAPER PUBLICATIONS
[README] FINISHING DOKU TESIS
[README] PANDUAN PUBL HASIL PENELITIAN
Abdul Razak Naufal
Ega Kartika Adhitya
Novi Wulandari
Ispandi
Lila Dini Utami
Wika purbasari
Add a card...

Contoh Card



Research Problems(RP) and Research Question(RQ)
Delete...
0%
RP : Naïve Bayes perform less well for predicting software defect due to the assumption that all attributes are equally important and are not related to each other
RQ1 : How absolute correlation coefficient does affects the accuracy of Weighted Naïve Bayes for classifying on software defect prediction?
RQ2 : How does absolute correlation coefficient for weighting attribute-class on Weighted Naïve Bayes affect the accuracy of software defect prediction?
Add an item...
Proposed Model Development
Hide completed items.
Delete...
100%
~~RQ1 Model Development~~
~~RQ2 Model Development~~
Add an item...
Experiment and Result Analysis
Hide completed items.
Delete...
100%
~~RQ1 Experiments~~
~~RQ2 Experiments~~
Add an item...
Thesis Finishing and Defense
Hide completed items.
Delete...
86%
~~Chapter 1 Introduction~~
~~Chapter 2 Literature Review~~
~~Chapter 3 Research Methods~~
~~Chapter 4 Experiment Results and Analysis~~
~~Chapter 5 Conclusion~~
~~Presentation (Slide) Development~~
Presentation Exercise
Boards
Thesis Project
Intelligent Systems
Private
Romi Satria Wahono
Calendar
Show Menu
1. LITERATURE REVIEWS
6. PAPER PUBLICATIONS
Add a card...

Contoh Hasil Koreksi



PROPOSAL TESIS

SELEKSI FITUR BERBASIS FILTER UNTUK MENINGKATAN KLASIFIKASI INTRUSI

judul tdk tajam dan tdk lengkap !

Oleh :

YUSNARDI

P31.2009.00676

PROGRAM PASCASARJANA

MAGISTER TEKNIK INFORMATIKA

UNIVERSITAS DIAN NUSWANTORO

SEMARANG

2013

untuk mendeteksi berbagai ancaman sebelum mereka menimbulkan berbagai kerusakan.

Tujuan utama IDS adalah untuk mendeteksi serangan dari pihak luar sekaligus mendeteksi penyalahgunaan yang dilakukan oleh orang dalam (pihak internal) terhadap sumber daya (*resource*) sebuah sistem komputer, serta informasi yang berada di dalam sumber daya tersebut. Akses yang tidak sah dari pihak luar dapat terjadi dengan cara memanfaatkan kebocoran *firewall*, mengeksploitasi adanya celah-celah keamanan, penggunaan teknik *tunneling* dengan mengunakan protokol yang tidak aman, atau merusak infrastruktur keamanan melalui *link* yang tidak dilindungi [5]. Sementara itu, bentuk intrusi dari dalam dapat dibagi menjadi: *masqueraders*, yang mencuri identitas pengguna lain atau pura-pura menjadi pengguna lain tersebut dengan tujuan untuk memiliki hak akses orang yang dicuri identitasnya tersebut, dan bentuk intrusi lain adalah *clandestin*, yaitu pengguna yang masuk dengan tujuan untuk menonaktifkan sistem [7]. Kedua bentuk ancaman tersebut di atas, baik intrusi dari dalam maupun dari luar, dapat mengganggu integritas, kerahasiaan, dan ketersediaan sumber daya atau informasi sebuah sistem komputer.

Berdasarkan pada bagaimana data dianalisa, deteksi intrusi dapat diklasifikasi ke dalam dua kategori utama [4] [8] yaitu: deteksi berbasis penyalahgunaan dan deteksi berbasis anomali. Deteksi penyalahgunaan adalah membandingkan pola dari bentuk serangan atau ancaman yang telah dikenal guna mengenali kemungkinan adanya upaya intrusi [9]. Deteksi penyalahgunaan juga dikenal sebagai deteksi berbasis pengetahuan (*knowledge-based detection*) dan *Signature Detection (SD)*. Sayangnya, deteksi penyalahgunaan lemah dalam mendeteksi serangan baru. Tetapi kelemahan tersebut dapat diatasi oleh deteksi berbasis anomali. Deteksi anomali adalah deteksi dengan membandingkan perilaku yang telah dikenal dan profil perilaku normal terhadap berbagai peristiwa yang diamati untuk mengenali serangan yang dianggap anomali atau menyimpang dari perilaku yang telah biasa dikenal [10]. Deteksi anomali juga dikenal dengan deteksi berbasis perilaku (*behavior-based detection*). Namun menurut [3] deteksi anomali lebih sering memunculkan *false-fositif*. Oleh karena itu, SD dan AD merupakan metode deteksi intrusi yang saling melengkapi, karena yang pertama fokus pada serangan/ancaman tertentu dan yang kedua fokus kepada serangan yang tidak diketahui [4].

Dikarenakan sulitnya membedakan serangan dan akses normal pada jaringan, berbagai motode telah diusulkan, diataranya motode berbasis aturan yang memiliki performa deteksi yang tinggi [4]. Diantara pendekatan dalam metode berbasis aturan tersebut adalah teknik data mining. Dalam beberapa tahun terakhir, telah banyak aplikasi yang menggunakan teknik data mining dalam deteksi intrusi [11] [12]. Data mining merupakan proses untuk mencari informasi secara sistematis dengan cara mengeksplorasi pola, tren, dan hubungan tersembunyi dalam volume data yang berukuran besar, kemudian menggunakan data berharga tersebut untuk memprediksi [13]. Data mining yang umum digunakan dalam bidang deteksi intrusi

Contoh Peraturan Bimbingan

Peraturan Bimbingan Penelitian Tesis

Romi Satria Wahono

1. Pahami dan lakukan Tahapan Penelitian mengikuti tulisan di: *http://romisatriawahono.net/2013/01/23/tahapan-memulai-penelitian-untuk-mahasiswa-galau*
2. Lakukan literature review untuk menentukan Research Field, Topic dan merumuskan Masalah Penelitian. Contoh research field dan topic bisa diakses melalui: *http://romisatriawahono.net/research*
3. Download paper: sci-hub.org, libgen.org, new-eresources.pnri.go.id, pustaka.ristek.go.id
   Request paper: *http://facebook.com/groups/intelligentsystems*
4. Kelola seluruh paper yang didownload dengan menggunakan aplikasi Mendeley (*http://mendeley.com*)
5. Diskusi dilakukan di group Intelligent Systems di Telegram dan FB. Mahasiswa harus mengirim laporan perkembangan setiap hari Senin melalui Board Thesis di *http://trello.com/intelligentsystems*
6. Untuk bimbingan darat, cek jadwal saya di *http://romisatriawahono.net* dan kirim konfirmasi via Telegram melalui nomor *081586220090*

Contoh Peraturan Bimbingan

Monitoring Perkembangan Penelitian Tesis

Romi Satria Wahono

1. Buka email dan accept invitation, lakukan registrasi di *http://trello.com*
2. Masuk ke Board Thesis (*https://trello.com/b/ndSslnk9/thesis*), buat Card (Add a Card), isi nama Card dengan nama lengkap kita, ikuti contoh card Romi Satria Wahono
3. Edit Card yg sudah dibuat, masukan judul penelitian di Card Description. Tentukan Label sesuai field penelitian kita
4. Buat checklist (Add Checlist) dan isi Item sesuai dengan format yang ada di Card Romi Satria Wahono
5. Update profile, foto dan biografi dengan lengkap. Assign diri sendiri dan Romi Satria Wahono di Card anda
6. Setiap hari Senin harus melakukan updating di Card masing-masing berdasarkan perkembangan penelitian
    1. Updating berupa: edit Checklist/Item atau upload tesis yg ditulis (Attach File). Tulis di Activity, setelah upload file. Gunakan format nama-judul-tahun, dan hanya boleh file pdf
    2. Pindahkan Card ke posisi paling atas setelah update dilakukan
    3. Progress tesis harus direview dan disetujui oleh minimal satu orang senior di group Telegram, sebelum saya review
    4. Komentar tentang progress akan ada di Activity di card

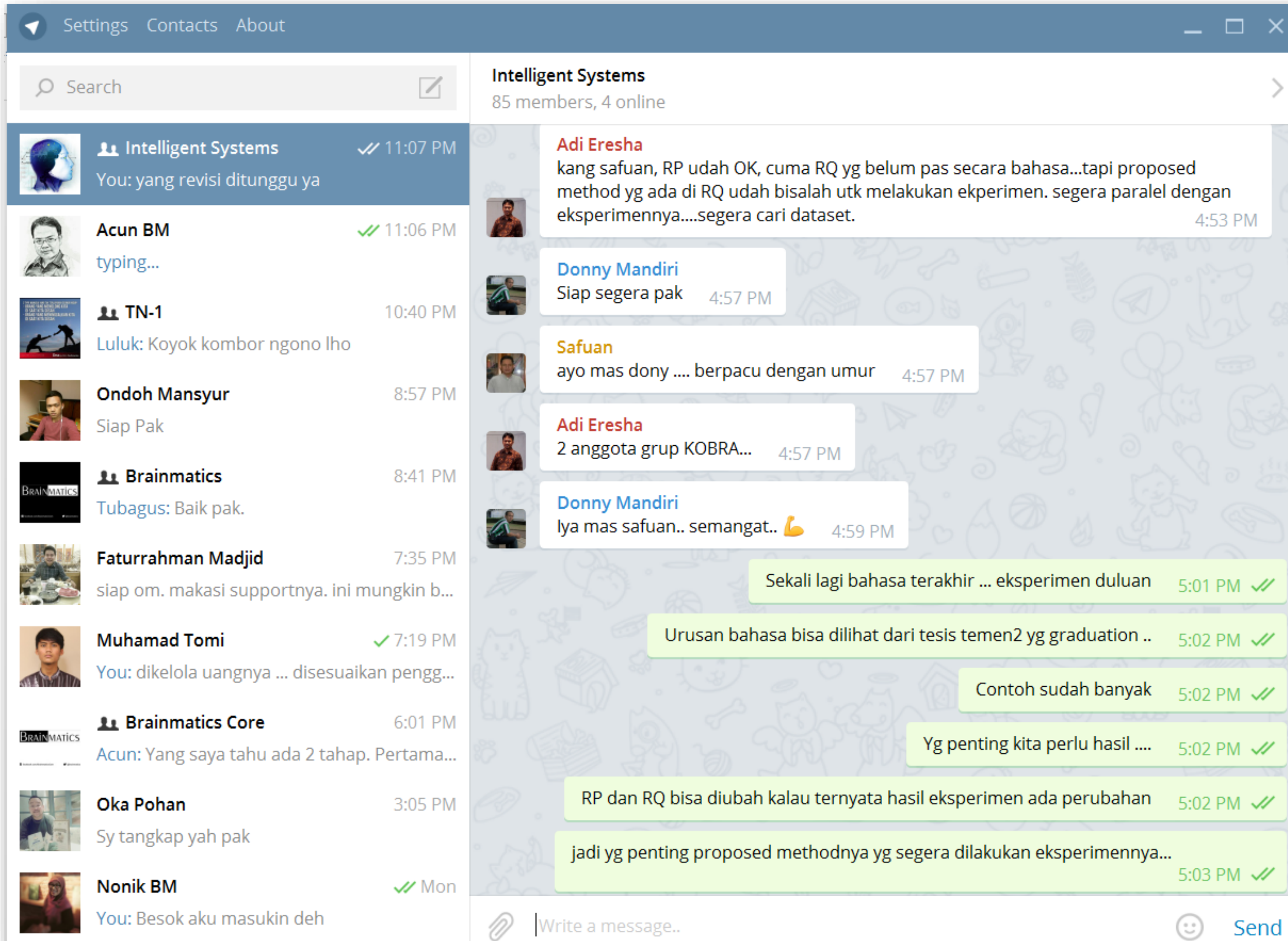
Contoh Peraturan Bimbingan

Finishing Dokumen Tesis

Romi Satria Wahono

1. Setelah sidang tesis, lakukan revisi secepatnya supaya tidak menjadi beban terlalu lama
2. Mahasiswa wajib menulis satu paper ke Journal of Intelligent Systems atau Journal of Software Engineering.
3. Dokumen versi terakhir yg harus diupload ke Card di Trello sebelum tanda tangan tesis adalah:
   1. Proposal Tesis versi Terakhir (PDF)
   2. Slide Sidang Proposal versi Terakhir (PPT)
   3. Tesis versi Terakhir (PDF)
   4. Slide Sidang Tesis versi Terakhir (PPT)
   5. Paper untuk publikasi ke IlmuKomputer.Com Journal, 10-15 halaman, dan gunakan template penulisan (DOCX)
   6. Dataset yang Digunakan (ZIP)
   7. Source Code (ZIP)

Diskusi di Grup Telegram



5.2 Presentasi Penelitian

Kiat Mempersiapkan Slide

- Secara umum alur presentasi sebaiknya mengikuti alur tesis
- Gunakan theme slide putih bersih, hindari menggunakan gambar dan animasi yang tidak perlu
- Gunakan font hitam untuk isi slide, dan kata-kata yang penting bisa diberi penekanan dengan font warna merah atau biru
- Ukuran font jangan lebih kecil dari 20pt, usahakan menggunakan 28pt atau 26pt
- Citation sebaiknya diselipkan di kalimat dalam slide (usahakan style APA atau Harvard), dengan font yang diset lebih kecil (18pt or 20pt)
- Slide tidak berisi paragraph penuh, slide hanya poin-poin penting saja, berikan ilustrasi gambar bila diperlukan
- Gunakan slide size standard (4:3) karena secara umum projector mendukung ukuran ini, jangan pernah gunakan size widescreen (16:9)

Kiat Presentasi di Ujian Tesis -1-

- Lakukan presentasi dengan tenang, gunakan suara lantang dan meyakinkan
- Atur suara supaya tidak terlalu tergesa-gesa dan juga tidak terlalu lambat
- Beri penekanan suara terhadap poin-poin yang kita anggap penting
- Jangan pernah membaca slide! Kita akan kehilangan ruh terhadap presentasi kita bila mulai membaca slide
- Slide hanya berisi poin-poin, karena itu hapalkan kalimat yang ingin kita ungkapkan dengan berdasarkan ke poin-poin di slide tersebut

Kiat Presentasi di Ujian Tesis -2-

- Dalam sesi tanya jawab, dengarkan dengan tenang, catat bila perlu apa yang diungkapkan penguji
- Tunggu sampai penguji selesai bicara, jangan memotong dengan jawaban singkat, karena itu membuat jawaban kita tidak komprehensif, dan ide kita gampang dijatuhkan
- Pahami dan hapalkan referensi utama kita, jawab pertanyaan dengan suara lantang dan meyakinkan, dengan landasan yang shahih dan jelas dari referensi yang kita pahami tersebut
- Jangan mendebat dengan tanpa landasan, jangan banyak gunakan “saya duga/pikir”, terima kalau koreksi dari penguji memang logis, sampaikan bahwa akan melakukan revisi di bagian yang telah dikoreksi penguji

Penilaian Presentasi Penelitian
(Berndtsson, 2008)

1. General: Examiners will look at the relevance and appropriateness of the topic you have studied, the significance of the findings and the amount of contribution you have achieved
2. Report: Examiners will look for clarity, consistency, an appropriate use of arguments, a clear differentiation between your own work and that of others in the literature and appropriate referencing
3. Defence: Examiners will assess the types of arguments you have made to support and defend your claims and conclusions. They will also look for your own insight and understanding in the work you have presented
4. Other: Examiners will review the administrative issues of your project. For example, have you followed the regulations correctly? Have you provided the right documentation at the right time?

Penilaian Presentasi Penelitian
(Berndtsson, 2008)

1. General
    1. Relevance of chosen topic
    2. Originality of chosen topic
    3. Significance of findings
    4. Degree to which the work is the student's own work
2. Report
    1. Clarity of presentation
    2. Consistency between different parts of the report
    3. Degree of insight apparent from the arguments presented to support the choices that the student has made
    4. Ability to differentiate between others' thoughts and own
    5. Ability to handle references and citations
    6. General stylistic impression

Penilaian Presentasi Penelitian (Berndtsson, 2008)

3. Defence
   1. Degree of insight apparent from the arguments presented to support claims and conclusions
   2. Degree of insight apparent from discussion in response to relevant questions
4. Other
   1. How the students performed as opponent
   2. Fulfillment of deadlines and other formal requirements

Penilaian Presentasi Penelitian
(Chinneck, 1999)

1. What was the research question?
2. Is it a 'good' question? This involves a comprehensive literature review to ensure that the question is 'useful' – i.e., worth answering. Through the literature review the student will show the context of the question, that the question has not been answered before and the extent to which others may have partly answered the question in the past.
3. Has the student answered the question adequately?
4. Has the student made an adequate contribution to knowledge?

Penilaian Presentasi Penelitian
(Dawson, 2009)

1. General considerations: These considerations occur in all types of projects at all degree levels
2. Foundations of your project: is its existence justified within other literature in the field?
3. The project approach from a technical perspective (i.e., not a project management viewpoint). Were the correct methods used? Were appropriate data gathered?
4. Results and contribution of the project. This is particularly important at postgraduate level where the ultimate contribution of the work is the quality measure used

Penilaian Presentasi Penelitian (Wahono, 2015)

1. Research:
   1. Validasi Masalah dan Kontribusi Penelitian
   2. Metode Penelitian, Evaluasi dan Validasi Metode Yang Diusulkan
   3. Hasil Penelitian dan Penarikan Kesimpulan
2. Report:
   1. Teknik Penulisan Ilmiah
   2. Penerapan Standard Formatting
   3. Citation and Reference
3. Presentation:
   1. Alur dan Tahapan Presentasi
   2. Penguasaan Materi Presentasi
   3. Argumentasi dalam Mempertahankan Ide

Reference

- Abbott, M., & McKinney, J. (2013). Understanding and Applying Research Design. John Wiley & Sons, Inc.
- Berndtsson, M., Hansson, J., & Olsson, B. (2008). Thesis Projects: a Guide for Students in Computer Science and Information Systems (2nd ed.). London: Springer-Verlag
- Blaxter, L., Hughes, C., & Tight, M. (2006). How to Research (3rd ed.). Open University Press
- Blessing, L. T. M., & Chakrabarti, A. (2009). DRM, a Design Research Methodology. Springer-Verlag London
- Cohen, L., Manion, L., & Morrison, K. (2005). Research Methods in Education (5th ed.). Taylor & Francis Group.
- Dawson, C. W. (2009). Projects in Computing and Information Systems A Student's Guide (2nd ed.). Pearson Education Limited
- Jonker, J., & Pennink, B. (2010). The Essence of Research Methodology. Springer-Verlag Berlin Heidelberg
- Lichtfouse, E. (2013). Scientific Writing for Impact Factor Journals. Nova Science Publishers, Inc.

Reference

- Kothari, C. (2004). Research Methodology: Methods and Techniques. New Age International
- Might, M. (2010). The Illustrated Guide to a Ph.D. Matt.might.net. Retrieved from *http://matt.might.net/articles/phd-school-in-pictures/*
- Marczyk, G., DeMatteo, D., & Fertinger, D. (2005). Essentials of Research Design and Methodology. John Wiley & Sons, Inc.
- Rea, L. M., & Parker, R. A. (2014). Designing and Conducting Survey Research: A Comprehensive Guide (4th ed.). John Wiley & Sons, Inc.
- Runeson, P., Host, M., Rainer, A., & Regnell, B. (2012). Case Study Research in Software Engineering: Guidelines and Examples. John Wiley & Sons, Inc.
- Sahu, P. K. (2013). Research Methodology: A Guide for Researchers In Agricultural Science, Social Science and Other Related Fields. Springer.
- Veit, R., Gould, C., & Gould, K. (2013). Writing, Reading, and Research (9th ed.). Cengage Learning.

Terima Kasih

Romi Satria Wahono
romi@romisatriawahono.net
http://romisatriawahono.net
08118228331