

**IMPLEMENTASI MODEL PEMBELAJARAN MESIN DENGAN
METODE ENSAMBEL DAN TEKNIK SELEKSI FITUR PADA PREDIKSI
TINGKAT KEMAMPUAN PEMELIHARAAN PERANGKAT LUNAK**

SKRIPSI

diajukan untuk memenuhi sebagian syarat untuk memperoleh gelar Sarjana
Komputer pada Program Studi S1 Rekayasa Perangkat Lunak



Oleh
Mochamad Nurul Huda
NIM 1905375

**PROGRAM STUDI REKAYASA PERANGKAT LUNAK
KAMPUS UPI DI CIBIRU
UNIVERSITAS PENDIDIKAN INDONESIA
2023**

**IMPLEMENTASI MODEL PEMBELAJARAN MESIN DENGAN
METODE ENSAMBEL DAN TEKNIK SELEKSI FITUR PADA PREDIKSI
TINGKAT KEMAMPUAN PEMELIHARAAN PERANGKAT LUNAK**

Oleh
Mochamad Nurul Huda
NIM 1905375

Sebuah Skripsi yang Diajukan untuk Memenuhi Salah Satu Syarat Memperoleh
Gelar Sarjana Komputer pada Program Studi S1 Rekayasa Perangkat Lunak

© Mochamad Nurul Huda
Universitas Pendidikan Indonesia
2023

Hak Cipta Dilindungi Undang-Undang
Skripsi ini tidak boleh diperbanyak seluruhnya atau sebagian, dengan dicetak
ulang, difoto kopi, atau cara lainnya tanpa izin dari penulis

HALAMAN PENGESAHAN

MOCHAMAD NURUL HUDA

**IMPLEMENTASI MODEL PEMBELAJARAN MESIN DENGAN METODE
ENSAMBEL DAN TEKNIK SELEKSI FITUR PADA PREDIKSI TINGKAT
KEMAMPUAN PEMELIHARAAN PERANGKAT LUNAK**

disetujui dan disahkan oleh pembimbing:

Pembimbing I



Indira Syawanodya, S.Kom., M.Kom.

NIP. 920190219920423201

Pembimbing II



Mochamad Iqbal Ardimansyah, S.T., M.Kom.

NIP. 920190219910328101

Mengetahui

Ketua Program Studi Rekayasa Perangkat Lunak



Mochamad Iqbal Ardimansyah, S.T., M.Kom.

NIP. 920190219910328101

HALAMAN PERNYATAAN KEASLIAN SKRIPSI DAN PERNYATAAN BEBAS PLAGIARISME

Dengan ini saya menyatakan bahwa skripsi dengan judul “Implementasi Model Pembelajaran Mesin Dengan Metode Ensambel dan Teknik Seleksi Fitur Pada Prediksi Tingkat Kemampuan Pemeliharaan Perangkat Lunak” ini beserta seluruh isinya adalah benar-benar karya saya sendiri. Saya tidak melakukan penjiplakan atau pengutipan dengan cara-cara yang tidak sesuai dengan etika ilmu yang berlaku dalam masyarakat keilmuan. Atas pernyataan ini, saya siap menanggung risiko/sanksi apabila di kemudian hari ditemukan adanya pelanggaran etika keilmuan atau ada klaim dari pihak lain terhadap keaslian karya saya ini.

Bandung, Januari 2023

Penulis,



Mochamad Nurul Huda

NIM. 1905375

HALAMAN UCAPAN TERIMA KASIH

Dengan mengucapkan puji dan syukur ke hadirat Allah SWT, karena atas segala rahmat dan karunia-Nya, sehingga penulis dapat menyelesaikan skripsi yang berjudul “Implementasi Model Pembelajaran Mesin Dengan Metode Ensambel dan Teknik Seleksi Fitur Pada Prediksi Tingkat Kemampuan Pemeliharaan Perangkat Lunak” ini dengan baik. Skripsi ini merupakan salah satu syarat untuk mendapatkan gelar sarjana dan menyelesaikan studi di Universitas Pendidikan Indonesia pada program studi Rekayasa Perangkat Lunak.

Dokumen ini berisi segala sesuatu mengenai penelitian yang telah dilakukan oleh penulis. Selesainya skripsi ini tidak terlepas dari bantuan dan dorongan berbagai pihak yang ikut serta memberikan bimbingan, kritik, saran, dukungan, dan motivasi kepada penulis selama proses penelitian dan penyusunan dokumen. Untuk itu penulis mengucapkan terima kasih kepada:

1. Ibu Indira Syawanodya, S.Kom., M.Kom., selaku Dosen Pembimbing Skripsi Utama yang di bawah pengawasannya selalu memberikan bimbingan, saran, dan umpan balik yang bermanfaat.
2. Bapak Mochamad Iqbal Ardimansyah, ST., M.Kom., selaku Dosen Pembimbing Skripsi Kedua dan Ketua Prodi Rekayasa Perangkat Lunak yang di bawah pengawasannya juga selalu memberikan bimbingan, saran, dan umpan balik yang bermanfaat.
3. Ibu Dian Anggraini, S.ST., M.T., selaku Dosen Pembimbing Akademik yang memberikan bimbingan, dan dukungan selama kuliah hingga pada saat penyusunan skripsi ini.
4. Kedua orang tua dan kak penulis, yaitu Bapak Suyatmin, Ibu Walsiyah dan Dinar Khusnul Chotimah yang selalu memberikan doa dan dukungan, baik secara moril maupun materi sehingga penulis dapat menyelesaikan skripsi ini dengan baik.
5. Teman-teman yang tidak dapat penulis sebutkan satu persatu yang memberikan saran, dukungan serta motivasi kepada penulis.

Penulis menyadari bahwa dokumen skripsi ini masih memiliki keterbatasan. Oleh karena itu, penulis mengharapkan kritik dan saran yang bersifat membangun agar tidak terjadi kesalahan yang sama pada saat melakukan penelitian di masa

depan. Akhir kata, penulis memohon maaf sebesar-besarnya apabila terdapat kesalahan baik dari penelitian yang dilakukan maupun cara penulisan dalam dokumen skripsi ini.

Bandung, Januari 2023

Penulis,

Mochamad Nurul Huda

NIM. 1905375

ABSTRAK

Tingkat kemampuan pemeliharaan perangkat lunak merupakan salah satu atribut eksternal dasar dari kualitas perangkat lunak yang mengukur tingkat efektivitas dan efisiensi di mana suatu perangkat lunak dapat dimodifikasi oleh pemelihara perangkat lunak tersebut. Tingkat kemampuan pemeliharaan perangkat lunak diukur menggunakan prediksi sebuah model pembelajaran mesin berdasarkan sejumlah atribut kualitas perangkat lunak untuk mendukung dan membantu dalam pengambilan keputusan pada saat proses pemeliharaan perangkat lunak dilakukan. Sumber himpunan data baru yang terdiri dari lima *dataset* perangkat lunak berorientasi objek Java dengan tujuh belas jenis metrik tingkat kelas digunakan dalam penelitian ini. Model pembelajaran mesin dibangun dengan menggunakan beberapa model individu seperti *Lasso Regression*, *K-Nearest Neighbors*, *Regression Tree*, *Multilayer Perceptron*, *M5Rules*, *Support Vector Machine*, *Artificial Neural Network*, dan dengan menggunakan metode ensambel seperti *Bagging* dan *AdaBoost*. Selain itu, teknik seleksi fitur dipertimbangkan untuk mengidentifikasi fitur terbaik sehingga meningkatkan performa dari model prediksi. Penelitian ini bertujuan untuk menyelidiki performa berbagai sumber himpunan data dalam model pembelajaran mesin. Performa model ini di evaluasi dengan menggunakan tiga metrik evaluasi, yaitu MMRE, MAE, dan Pred. Hasil menunjukkan bahwa ANN menjadi algoritma terbaik pada model individu dengan MMRE 0.88 pada *dataset* Equinox Framework. Metode ensambel terbukti meningkatkan performa dari model dengan ketentuan metode ensambel cocok dengan algoritma individu yang digunakan. Performa terbaik didapatkan metode AdaBoost dengan ANN pada *dataset* Lucene dengan MMRE 0.78. Teknik seleksi fitur juga terbukti meningkatkan beberapa model prediksi dengan penghapusan fitur yang tepat dan algoritma yang digunakan cocok dengan distribusi datanya.

Kata Kunci : *Software Maintainability*; Model Individu; Metode Ensambel; Teknik Seleksi Fitur

ABSTRACT

Software maintainability is one of the primary external attributes of software quality that measures the effectiveness and efficiency with which the software maintainer can modify the software. Software maintainability is measured using the prediction of machine learning models based on several software quality attributes to support and assist in decision-making during the software maintenance process. This study used new datasets consisting of five Java object-oriented software systems with seventeen class-level metrics. Machine learning models are built using several individual models such as Lasso Regression, K-Nearest Neighbors, Regression Tree, Multilayer Perceptron, M5Rules, Support Vector Machine, Artificial Neural Network, and by using ensemble methods such as Bagging and AdaBoost. In addition, feature selection techniques are considered to identify the best features, thereby increasing the prediction model's performance. This research aims to investigate the performance of various dataset sources in machine learning models. The performance of these models is evaluated using three evaluation metrics, namely MMRE, MAE, and Pred. The results show that ANN is the best algorithm for individual models with MMRE 0.88 on the Equinox Framework dataset. The ensemble method is proven to improve the performance of the model, provided that the ensemble method matches the individual algorithms used. The AdaBoost method obtained the best performance with ANN on the Lucene dataset with MMRE 0.78. The feature selection technique is also proven to improve several prediction models with the proper feature removal, and the algorithm matches the data distribution.

Keyword : *Software Maintainability; Individual Models; Ensemble Methods; Feature Selection Techniques*

DAFTAR ISI

HALAMAN JUDUL.....	i
HALAMAN PENGESAHAN.....	iii
HALAMAN PERNYATAAN KEASLIAN SKRIPSI DAN PERNYATAAN BEBAS PLAGIARISME	iv
HALAMAN UCAPAN TERIMA KASIH	v
ABSTRAK.....	vii
ABSTRACT.....	viii
DAFTAR ISI.....	ix
DAFTAR TABEL.....	xi
DAFTAR GAMBAR	xii
DAFTAR LAMPIRAN.....	xv
BAB I PENDAHULUAN.....	16
1.1 Latar Belakang Penelitian	16
1.2 Rumusan Masalah Penelitian	18
1.3 Tujuan Penelitian.....	18
1.4 Manfaat Penelitian.....	19
1.5 Batasan Penelitian	19
1.6 Hipotesis Penelitian.....	19
1.7 Struktur Organisasi Skripsi	20
BAB II KAJIAN PUSTAKA.....	22
2.1 Kualitas Perangkat Lunak	22
2.2 Tingkat Kemampuan Pemeliharaan Perangkat Lunak	23
2.3 Pembelajaran Mesin	28
2.3.1 Lasso Regression.....	29
2.3.2 K-Nearest Neighbors.....	29
2.3.3 Regression Tree.....	30
2.3.4 Multilayer Perceptron.....	30
2.3.5 M5Rules	30
2.3.6 Support Vector Regression	31
2.3.7 Artificial Neural Network	31
2.4 Metode Ensambel	31

2.4.1	Bagging	32
2.4.2	AdaBoost.....	32
2.5	Teknik Seleksi Fitur	32
2.6	Penelitian Terkait	33
BAB III METODE PENELITIAN.....		43
3.1	Desain Penelitian	43
3.2	Sumber Himpunan Data	44
3.3	Instrumen Penelitian.....	49
3.4	Prosedur Penelitian.....	53
3.5	Analisis Data	54
BAB IV TEMUAN DAN PEMBAHASAN.....		55
4.1	Pra-Pemrosesan Data.....	55
4.2	Implementasi Model Individu	57
4.3	Implementasi Model Ensambel	64
4.3.1	Bootstrap Aggregating (Bagging).....	64
4.3.2	Adaptive Boosting (AdaBoost).....	71
4.4	Implementasi Model Dengan Seleksi Fitur	74
4.4.1	Dataset Eclipse JDT Core	78
4.4.2	Dataset Eclipse PDE UI	81
4.4.3	Dataset Equinox Framework.....	84
4.4.4	Dataset Lucene	87
4.4.5	Dataset Mylyn	90
BAB V SIMPULAN, IMPLIKASI, DAN REKOMENDASI		95
5.1	Simpulan.....	95
5.2	Implikasi dan Rekomendasi	96
DAFTAR PUSTAKA		97
LAMPIRAN.....		101

DAFTAR TABEL

Tabel 2.1	Karakteristik Kualitas Produk Perangkat Lunak Berdasarkan ISO/IEC 25010:2011.....	22
Tabel 2.2	Ringkasan Pengukuran Tingkat Kemampuan Pemeliharaan Perangkat Lunak.....	24
Tabel 2.3	Metrik Tingkat Kemampuan Pemeliharaan Perangkat Lunak.....	27
Tabel 2.4	Ringkasan Penelitian Terkait	33
Tabel 2.5	Ringkasan Hasil Pengujian Model Individu dan Ensambel.....	35
Tabel 2.6	Ringkasan Hasil Pengujian pada <i>Dataset</i> UIMS	38
Tabel 2.7	Ringkasan Hasil Pengujian pada <i>Dataset</i> QUES.....	38
Tabel 2.8	Hasil Eksperimen Menggunakan Model Individu	41
Tabel 2.9	Hasil Eksperimen Menggunakan Model Ensambel <i>Bagging</i>	41
Tabel 2.10	Hasil Eksperimen Menggunakan Model Ensambel <i>Additive Regression</i>	42
Tabel 3.1	Deskripsi Sistem Perangkat Lunak pada <i>Software Maintainability Prediction Dataset</i>	45
Tabel 3.2	Statistik Deskriptif Metrik CHANGE.....	46
Tabel 3.3	Metrik-Metrik pada <i>Software Maintainability Prediction Dataset</i>	47
Tabel 3.4	Spesifikasi Komputasi Vertex AI yang Digunakan	50
Tabel 3.5	Perangkat Lunak dan <i>Library</i> yang Digunakan	50
Tabel 3.6	Metrik-Metrik Evaluasi Regresi.....	52
Tabel 4.1	Statistik Deskriptif Metrik CHANGE Setelah Reduksi Data	56
Tabel 4.2	Hasil Evaluasi Model Individu	57
Tabel 4.3	Hasil Evaluasi Model Ensambel <i>Bagging</i>	64
Tabel 4.4	Hasil Evaluasi Model Ensambel <i>AdaBoost</i>	72

DAFTAR GAMBAR

Gambar 2.1 Model Kualitas Produk Perangkat Lunak ISO/IEC 25010:2011	23
Gambar 2.2 Kerangka Kerja <i>Maintainability Prediction</i> oleh Kumar dkk. (2019)	37
Gambar 2.3 Kerangka Kerja Penelitian oleh H. A. Alsolai (2020).....	39
Gambar 2.4 Kerangka Kerja Pra-Pemrosesan data oleh H. A. Alsolai (2020).....	40
Gambar 3.1 Skema Penelitian.....	43
Gambar 3.2 <i>Boxplot</i> Metrik CHANGE.....	46
Gambar 3.3 <i>Goal Question Metric</i> (GQM).....	50
Gambar 3.4 Kerangka Kerja yang Diusulkan	53
Gambar 4.1 Proporsi Data yang Dihapus Selama Reduksi Data	55
Gambar 4.2 <i>Boxplot</i> Metrik CHANGE Setelah Reduksi Data	56
Gambar 4.3 Perbandingan Performa Model Individu Dengan Penelitian Sebelumnya Pada Dataset Eclipse JDT Core.....	59
Gambar 4.4 Perbandingan Performa Model Individu Dengan Penelitian Sebelumnya Pada Dataset Eclipse PDE UI.....	60
Gambar 4.5 Perbandingan Performa Model Individu Dengan Penelitian Sebelumnya Pada Dataset Equinox Framework	61
Gambar 4.6 Perbandingan Performa Model Individu Dengan Penelitian Sebelumnya Pada Dataset Lucene	62
Gambar 4.7 Perbandingan Performa Model Individu Dengan Penelitian Sebelumnya Pada Dataset Mylyn	63
Gambar 4.8 Perbandingan Performa Model Ensambel Bagging Dengan Penelitian Sebelumnya Pada Dataset Eclipse JDT Core.....	66
Gambar 4.9 Perbandingan Performa Model Ensambel Bagging Dengan Penelitian Sebelumnya Pada Dataset Eclipse PDE UI.....	67
Gambar 4.10 Perbandingan Performa Model Ensambel Bagging Dengan Penelitian Sebelumnya Pada Dataset Equinox Framework	68
Gambar 4.11 Perbandingan Performa Model Ensambel Bagging Dengan Penelitian Sebelumnya Pada Dataset Lucene	69
Gambar 4.12 Perbandingan Performa Model Ensambel Bagging Dengan Penelitian Sebelumnya Pada Dataset Mylyn	70

Gambar 4.13 Selisih Performa Antara Model Individu dan Model Ensemble Bagging	71
Gambar 4.14 Selisih Performa Antara Model Individu dan Model Ensemble AdaBoost.....	74
Gambar 4.15 Korelasi Antar Variabel Pada Dataset Eclipse JDT Core	75
Gambar 4.16 Korelasi Antar Variabel Pada Dataset Eclipse PDE UI	75
Gambar 4.17 Korelasi Antar Variabel Pada Dataset Equinox Framework.....	76
Gambar 4.18 Korelasi Antar Variabel Pada Dataset Lucene.....	76
Gambar 4.19 Korelasi Antar Variabel Pada Dataset Mylyn.....	77
Gambar 4.20 Selisih Performa Antara Model Individu Tanpa Seleksi Fitur dan Model Individu Dengan Seleksi Fitur Pada Eclipse JDT Core.....	79
Gambar 4.21 Selisih Performa Antara Model Bagging Tanpa Seleksi Fitur dan Model Bagging Dengan Seleksi Fitur Pada Eclipse JDT Core.....	80
Gambar 4.22 Selisih Performa Antara Model AdaBoost Tanpa Seleksi Fitur dan Model AdaBoost Dengan Seleksi Fitur Pada Eclipse JDT Core	81
Gambar 4.23 Selisih Performa Antara Model Individu Tanpa Seleksi Fitur dan Model Individu Dengan Seleksi Fitur Pada Eclipse PDE UI.....	82
Gambar 4.24 Selisih Performa Antara Model Bagging Tanpa Seleksi Fitur dan Model Bagging Dengan Seleksi Fitur Pada Eclipse PDE UI.....	83
Gambar 4.25 Selisih Performa Antara Model AdaBoost Tanpa Seleksi Fitur dan Model AdaBoost Dengan Seleksi Fitur Pada Eclipse PDE UI	84
Gambar 4.26 Selisih Performa Antara Model Individu Tanpa Seleksi Fitur dan Model Individu Dengan Seleksi Fitur Pada Equinox Framework	85
Gambar 4.27 Selisih Performa Antara Model Bagging Tanpa Seleksi Fitur dan Model Bagging Dengan Seleksi Fitur Pada Equinox Framework	86
Gambar 4.28 Selisih Performa Antara Model AdaBoost Tanpa Seleksi Fitur dan Model AdaBoost Dengan Seleksi Fitur Pada Equinox Framework.....	87
Gambar 4.29 Selisih Performa Antara Model Individu Tanpa Seleksi Fitur dan Model Individu Dengan Seleksi Fitur Pada Lucene	88
Gambar 4.30 Selisih Performa Antara Model Bagging Tanpa Seleksi Fitur dan Model Bagging Dengan Seleksi Fitur Pada Lucene	89

Gambar 4.31 Selisih Performa Antara Model AdaBoost Tanpa Seleksi Fitur dan Model AdaBoost Dengan Seleksi Fitur Pada Lucene	90
Gambar 4.32 Selisih Performa Antara Model Individu Tanpa Seleksi Fitur dan Model Individu Dengan Seleksi Fitur Pada Mylyn	91
Gambar 4.33 Selisih Performa Antara Model Bagging Tanpa Seleksi Fitur dan Model Bagging Dengan Seleksi Fitur Pada Mylyn	92
Gambar 4.34 Selisih Performa Antara Model AdaBoost Tanpa Seleksi Fitur dan Model AdaBoost Dengan Seleksi Fitur Pada Mylyn	93

DAFTAR LAMPIRAN

Lampiran 1. Statistik Deskriptif Pada Semua Dataset	101
Lampiran 2. Distribusi Data Antar Variabel Pada Semua Dataset	104
Lampiran 3. Hasil Evaluasi Model Individu Dengan Seleksi Fitur 12 Metrik yang Sama.....	109
Lampiran 4. Hasil Evaluasi Model Ensambel Bagging Dengan Seleksi Fitur 12 Metrik yang Sama	111
Lampiran 5. Hasil Evaluasi Model Ensambel AdaBoost Dengan Seleksi Fitur 12 Metrik yang Sama	113
Lampiran 6. Hasil Evaluasi Model Individu Dengan Seleksi Fitur 12 Metrik yang Berbeda	115
Lampiran 7. Hasil Evaluasi Model Ensambel Bagging Dengan Seleksi Fitur 12 Metrik yang Berbeda.....	117
Lampiran 8. Hasil Evaluasi Model Ensambel AdaBoost Dengan Seleksi Fitur 12 Metrik yang Berbeda.....	119
Lampiran 9. Hasil Evaluasi Model Individu Dengan Seleksi Fitur 10 Metrik yang Berbeda	121
Lampiran 10. Hasil Evaluasi Model Ensambel Bagging Dengan Seleksi Fitur 10 Metrik yang Berbeda.....	123
Lampiran 11. Hasil Evaluasi Model Ensambel AdaBoost Dengan Seleksi Fitur 10 Metrik yang Berbeda.....	125
Lampiran 12. Visualisasi Hasil Model Regresi	127

DAFTAR PUSTAKA

- Ahmed, M. A., & Al-Jamimi, H. A. (2013). Machine learning approaches for predicting software maintainability: A fuzzy-based transparent model. *IET Software*, 7(6), 317–326.
- Alsolai, H. (2020). *Software maintainability prediction [Data set]*. Zenodo.
- Alsolai, H. A. (2020). *Investigating the Use of Ensemble Techniques in Predicting Object-Oriented Software Maintainability*. University of Strathclyde Glasgow.
- Alsolai, H., & Roper, M. (2020). A systematic literature review of machine learning techniques for software maintainability prediction. *Information and Software Technology*, 119, 106214.
- Alsolai, H., Roper, M., & Nassar, D. (2018). Predicting software maintainability in object-oriented systems using ensemble techniques. *Proceedings - 2018 IEEE International Conference on Software Maintenance and Evolution, ICSME 2018*, 716–721.
- Blessing, L. T. M., & Chalkrabarti, A. (2009). *DRM, a Design Research Methodology*. Springer.
- Breiman, L. (1996). Bagging predictors. *Machine Learning*, 24, 123–140.
- Burkov, A. (2019). *The Hundred-Page Machine Learning Book*. Andriy Burkov.
- Drucker, H., Surges, C. J. C., Kaufman, L., Smola, A., & Vapnik, V. (1997). Support vector regression machines. *Advances in Neural Information Processing Systems*, 1, 155–161.
- Dubey, S. K., Rana, A., & Dash, Y. (2012). Maintainability prediction of object-oriented software system by multilayer perceptron model. *ACM SIGSOFT Software Engineering Notes*, 37(5), 1–4.
- Elish, M. O., & Elish, K. O. (2009). *Application of TreeNet in Predicting Object-Oriented Software Maintainability : A Comparative Study*.
- Freund, Y., & Schapire, R. E. (1997). A Decision-Theoretic Generalization of On-

- Line Learning and an Application to Boosting. *Journal of Computer and System Sciences*, 55(1), 119–139.
- Geron, A. (2019). *Hands-On Machine Learning with Scikit-Learn and TensorFlow: Concepts, Tools, and Techniques to Build Intelligent Systems 2nd Edition* (Second Edi). O'Reilly Media.
- Holmes, G., Hall, M., & Prank, E. (1999). Generating rule sets from model trees. *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 1747, 1–12.
- IEEE Computer Society. (2014). *Guide to the Software Engineering Body of Knowledge Version 3.0 (SWEBOK Guide V3.0)* (P. Bourque & R. E. Fairley (ed.); 4 ed.). IEEE.
- ISO. (2011). ISO/IEC 25010:2011(E), Systems and software engineering — Systems and software Quality Requirements and Evaluation (SQuaRE) — System and software quality models. In *BSI Standards Publication*.
- Kitchenham, B. A., Pickard, L. M., MacDonell, S. G., & Shepperd, M. J. (2001). What accuracy statistics really measure. *IEE Proceedings: Software*, 148(3), 81–85.
- Kramer, O. (2013). Dimensionality Reduction with Unsupervised Nearest Neighbors. *Intelligent Systems Reference Library*, 51, 13–23.
- Kumar, L., Lal, S., & Murthy, L. B. (2019). Estimation of maintainability parameters for object-oriented software using hybrid neural network and class level metrics. In *International Journal of System Assurance Engineering and Management* (Vol. 10, Nomor 5). Springer India.
- Kumar, L., Naik, D. K., & Rath, S. K. (2015). Validating the Effectiveness of Object-Oriented Metrics for Predicting Maintainability. *Procedia Computer Science*, 57, 798–806.
- Kumar, L., & Rath, S. K. (2017). Software maintainability prediction using hybrid neural network and fuzzy logic approach with parallel computing concept. *International Journal of System Assurance Engineering and Management*, 8,

1487–1502.

- Lacerda, G., Petrillo, F., Pimenta, M., & Gaël, Y. (2020). *The Journal of Systems and Software Code smells and refactoring : A tertiary systematic review of challenges and observations*. 167.
- Land, R. (2002). Measurements of Software Maintainability. *Proceedings of the 4th ARTES Graduate Student*.
- Li, W., & Henry, S. (1993). Object-Oriented Metrics that Predict Maintainability. *Systems Software*, 23, 111–122.
- Loh, W. Y. (2011). Classification and regression trees. *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*, 1(1), 14–23.
- MacDonell, S. G. (1997). Establishing relationships between specification size and software process effort in CASE environments. *Information and Software Technology*, 39(1), 35–45.
- Murtagh, F. (1991). Multilayer perceptrons for classification and regression. *Neurocomputing*, 2(5–6), 183–197.
- Otchere, D. A., Ganat, T. O. A., Ojero, J. O., Tackie-Otoo, B. N., & Taki, M. Y. (2022). Application of gradient boosting regression model for the evaluation of feature selection techniques in improving reservoir characterisation predictions. *Journal of Petroleum Science and Engineering*, 208(PE), 109244.
- Pressman, R. S., & Maxim, B. R. (2015). Software Engineering: A Practitioner's Approach 8th Edition. In *McGraw-Hill Education* (8 ed.).
- Quinlan, J. R. (1992). Learning with continuous classes. *Australian Joint Conference on Artificial Intelligence*, 92, 343–348.
- Russell, S., & Norvig, P. (2010). *Artificial Intelligence: A Modern Approach Third Edition* (Vol. 56, Nomor 1). Pearson.
- Tang, J., Deng, C., & Huang, G.-B. (2016). Extreme Learning Machine for Multilayer Perceptron. *IEEE TRANSACTIONS ON NEURAL NETWORKS AND LEARNING SYSTEMS*, 27(4), 809–821.

- Thwin, M. M. T., & Quah, T. S. (2005). Application of neural networks for software quality prediction using object-oriented metrics. *Journal of Systems and Software*, 76(2), 147–156.
- Tibshirani, R. (1996). *Regression Shrinkage and Selection via the Lasso*. 58(1), 267–288.
- Uhrig, R. E. (1995). Introduction to artificial neural networks. *Proceedings of IECON '95 - 21st Annual Conference on IEEE Industrial Electronics*, 15–32.
- Van Koten, C., & Gray, A. R. (2006). An application of Bayesian network for predicting object-oriented software maintainability. *Information and Software Technology*, 48(1), 59–67.
- Wu, X., Kumar, V., Ross, Q. J., Ghosh, J., Yang, Q., Motoda, H., McLachlan, G. J., Ng, A., Liu, B., Yu, P. S., Zhou, Z. H., Steinbach, M., Hand, D. J., & Steinberg, D. (2008). Top 10 algorithms in data mining. In *Knowledge and Information Systems* (Vol. 14, Nomor 1).
- Zhou, Y., & Leung, H. (2007). Predicting object-oriented software maintainability using multivariate adaptive regression splines. *Journal of Systems and Software*, 80(8), 1349–1361.