



## Penerapan seleksi fitur median weighted information gain dengan K-NN pada dataset label hours software effort estimation

Indra Kurniawan<sup>\*1</sup>, Dwi Retna Sulistyowati<sup>2</sup>, Yasa Maulana<sup>3</sup>

Email: <sup>1</sup> [indrakurniawan.politbang@gmail.com](mailto:indrakurniawan.politbang@gmail.com), <sup>2</sup> [retno.seis@unisnu.ac.id](mailto:retno.seis@unisnu.ac.id), <sup>3</sup> [yasamaulana1@gmail.com](mailto:yasamaulana1@gmail.com)

<sup>1</sup>Prodi Rekayasa Perangkat Lunak, Politeknik Balekambang Jepara

<sup>2</sup>Teknik Industri, Fakultas Sains dan Teknologi, Universitas Islam Nahdlatul Ulama Jepara

Diterima: 25 Juli 2022 | Direvisi: 1 Agustus 2022 | Disetujui: 11 Agustus 2022

©2020 Program Studi Teknik Informatika Fakultas Ilmu Komputer,  
Universitas Muhammadiyah Riau, Indonesia

### Abstrak

Estimasi pengembangan perangkat lunak menjadi kebutuhan dikarenakan setiap pengembangan perangkat lunak memiliki keterbatasan biaya dan waktu dalam menyelesaikan sebuah proyek [1]. Secara spesifik hasil percobaan diperoleh model ML lebih unggul dari pada model non-ML. Dengan hasil secara spesifik yaitu 66% (52 dari 79) menunjukkan keunggulan model ML sedangkan hanya 34% (27 dari 79) hasil percobaan mengunggulkan model non-ML [2]. Beberapa studi yang sudah teridentifikasi, masalah utama yang paling sering ditemui dalam dataset estimasi usaha perangkat lunak yaitu masalah kategori fitur seperti fitur yang tidak relevan [2]. Berdasarkan latar belakang tersebut maka dalam penelitian ini akan dilakukan penerapan metode seleksi fitur Median WIG dengan algoritma k-NN yang akan berfokus pada dataset label *hours Software Effort Estimation* dan dibandingkan dengan algoritma k-NN. Hasil perbandingan yang diperoleh dalam seluruh pengujian dataset *Software Effort Estimation* diketahui bahwa metode usulan seleksi fitur Median-WIG dengan k-NN terbukti mampu meningkatkan akurasi dalam proses estimasi dilihat dari penurunan nilai RMSE yang signifikan pada seluruh dataset yaitu Albrecht sebesar 5.958, Miyazaki sebesar 62.363 dan Kemerer sebesar 131.027. Terjadinya penurunan nilai RMSE pada seluruh dataset ini memperlihatkan peningkatan kinerja setelah dilakukannya metode seleksi fitur pada dataset *Software Effort Estimation*. Terjadinya penurunan nilai RMSE ini memperlihatkan peningkatan kinerja setelah menggunakan metode seleksi fitur.

**Kata kunci:** *software effort estimation, irrelevant feature, median weighted information gain, k-nearest neighbor*

### Implementation of feature selection median weighted information gain with K-NN on dataset label hours software effort estimation

#### Abstract

Software development estimation is a necessity because every software development has limited cost and time in completing a project [1]. Specifically, the experimental results obtained that the ML model is superior to the non-ML model. With specific results, 66% (52 of 79) showed the superiority of the ML model while only 34% (27 of 79) of the experimental results favored the non-ML model [2]. Several studies have identified, the main problem that is most often encountered in the software effort estimation dataset is the problem of feature categories such as irrelevant features [2]. Based on this background, this research will apply the Median WIG feature selection method with the k-NN algorithm which will focus on the dataset label hours *Software Effort Estimation* and compare it with the k-NN algorithm. The comparison results obtained in all tests of the *Effort Estimation Software* dataset, it is known that the proposed method of selecting the Median-WIG feature with k-NN is proven to be able to increase accuracy in the estimation process seen from the significant decrease in RMSE values in all datasets, namely Albrecht of 5,958, Miyazaki of 62,363 and Kemerer amounted to 131,027. The decrease in the RMSE value in this entire dataset shows an increase in performance after the feature selection method on the *Software Effort Estimation* dataset. The decrease in the RMSE value shows an increase in performance after using the feature selection method.

**Keywords:** *software effort estimation, irrelevant feature, median weighted information gain, k-nearest neighbor*

## 1. PENDAHULUAN

Estimasi pengembangan perangkat lunak menjadi kebutuhan dikarenakan setiap pengembangan perangkat lunak memiliki keterbatasan biaya dan waktu dalam menyelesaikan sebuah proyek. Dapat melakukan perancangan dengan benar merupakan landasan untuk semua kegiatan manajemen proyek perangkat lunak. Melakukan estimasi atau memperkirakan berapa jumlah tenaga dan waktu yang dibutuhkan untuk menyelesaikan sebuah sistem tidak mudah dilakukan karena diperlukan pemahaman yang benar tentang sebuah pekerjaan [1].

Selain pada perangkat lunak estimasi juga dapat dilakukan pada beberapa bidang lain seperti pariwisata [3] yang dilakukan untuk memprediksi jumlah wisatawan yang datang. Melakukan estimasi atau memperkirakan berapa jumlah tenaga dan waktu yang dibutuhkan untuk menyelesaikan sebuah sistem tidak mudah dilakukan karena diperlukan pemahaman yang benar tentang sebuah pekerjaan.

Terdapat dua metode dalam melakukan estimasi pengembangan perangkat lunak yaitu metode Non Machine Learning (non-ML) [4] seperti diantaranya COCOMO [5], expert judgement [6], function point (FP) [7] dan use case point (UCP) [8] dan Machine Learning (ML) [4] seperti Metode-metode machine learning yang telah digunakan untuk estimasi usaha pengembangan perangkat lunak dan beberapa penelitiannya lainnya diantaranya k-Nearest Neighbor (k-NN) [9], [10], [11], [12], [13], [14]. Support Vector Machines [9], [15], Naive Bayes (NB) [16], Bayesian Networks (BN) [17], Decision Trees (DT) [10].

Hasil literatur review yang dilakukan Wen et al [2] menunjukkan hasil perbandingan antara model non-ML dan model ML dengan menggunakan metrik MMRE. Dengan hasil sebagian besar percobaan diperoleh model ML lebih unggul dari pada model non-ML. Dengan hasil secara spesifik yaitu 66% (52 dari 79) menunjukkan keunggulan model ML sedangkan hanya 34% (27 dari 79) hasil percobaan mengunggulkan model non-ML.

Dalam penelitian Wen et al [2] pendekatan CBR algorithm k-NN merupakan teknik yang paling banyak digunakan, yaitu sebanyak 37%. Pendekatan CBR dengan menggunakan algoritma k-NN mempunyai hasil yang lebih akurat dalam estimasi usaha perangkat lunak dibandingkan metode ML lainnya.

Dataset penelitian dalam *software effort* untuk estimasi pengembangan perangkat lunak diantaranya Albercht, Miyazaki, China, Desharnais, Maxwell, Kitchenham dan Cocomo. Menurut Wen et al dalam beberapa studi yang sudah teridentifikasi, masalah utama yang paling sering ditemui dalam dataset estimasi usaha perangkat lunak yaitu masalah kategori fitur dalam dataset seperti fitur yang tidak relevan, bobot fitur yang belum optimal, fitur yang terlalu banyak dan fitur yang sama [2].

Secara umum estimasi usaha pengembangan perangkat lunak sangat sulit dilakukan karena proyek perangkat lunak bersifat dinamis dan sulit mendapatkan proyek yang sangat mirip dengan proyek sebelumnya. Sehingga fitur setiap dataset *Software Effort Estimation* berbeda dan mengakibatkan terdapatnya fitur yang tidak relevan.

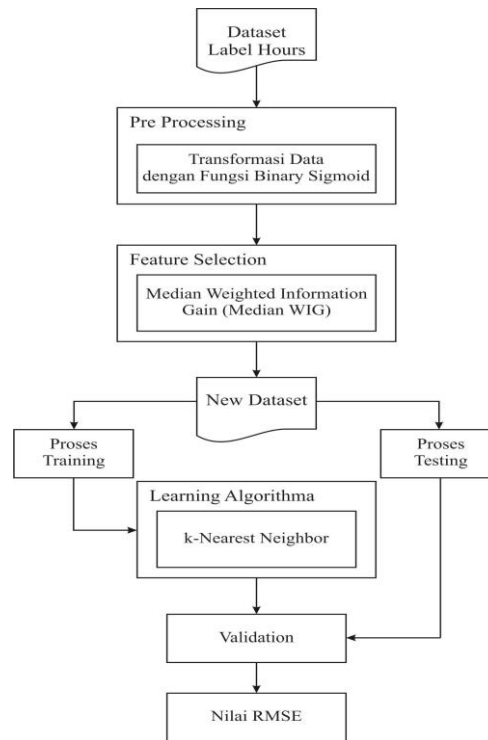
Tujuan dari pendekatan seleksi fitur adalah identifikasi fitur yang paling signifikan untuk penyelesaian masalah dan menghilangkan fitur yang tidak signifikan [18]. Pada penelitian Fatma dkk [19] yang menggunakan pembobotan kriteria sekaligus perankingan metode SAW (Simple Additive Weighting) untuk menentukan hasil terbaik.

Penelitian tentang seleksi fitur menggunakan metode *Weighted Information Gain* (WIG) memperlihatkan hasil yang baik dengan menggunakan bobot fitur untuk pemilihan fitur yang relevan dan menggunakan kriteria median fitur [20]. Tujuan dilakukannya pembobotan yaitu untuk melakukan perankingan fitur mana saja yang memenuhi kriteria (threshold) yang sudah ditentukan sebelumnya akan dipertahankan untuk menjadi kumpulan data baru yang akan digunakan dalam proses pengolahan dengan algoritma.

Berdasarkan latar belakang tersebut maka dalam penelitian ini akan dilakukan penerapan metode seleksi fitur Median WIG dengan algoritma k-NN yang akan berfokus pada dataset label *hours Software Effort Estimation* dan dibandingkan dengan algoritma k-NN saja.

## 2. METODE PENELITIAN

Alur penelitian yang digunakan dalam penelitian ini digambarkan sesuai alur Gambar 1 berikut:



Gambar 1. Tahapan Penelitian

Sesuai alur penelitian pada gambar 1 diatas maka metode yang diusulkan oleh peneliti dalam penelitian ini adalah sebagai berikut:

1. Mempersiapkan dataset *Software Effort Estimation* dengan label *hours*
2. Melakukan proses pre processing yaitu dengan mentransformasi setiap data pada dataset dengan menerapkan fungsi Binary Sigmoid. Yaitu mentransformasi data dalam rentang 0 sampai 1. Menurut Yu dan Xu dalam penelitiannya disebutkan bahwa hasil yang lebih baik akan diperoleh dengan menggunakan rentang kisaran 0.1 sampai 0.9 [21]. Adapun rumus pada persamaan (1) untuk fungsi binary sigmoid (Logsig) yaitu:

$$y' = \frac{x - x_{min}}{x_{max} - x_{min}} \times (0.9 - 0.1) + 0.1 \quad (1)$$

$y'$  = Hasil transformasi data  
 $x$  = Nilai asli  
 $x_{min}$  = Nilai minimal  
 $x_{max}$  = Nilai maksimal

3. Melakukan seleksi fitur dengan metode Median Weighted Information Gain (Median WIG) yaitu seleksi fitur yang relevan dan membentuk dataset baru hasil seleksi fitur dengan cara fitur dengan kriteria (threshold) yaitu nilai fitur yang diatas median (fitur relevan) akan dipertahankan untuk digunakan dan fitur dibawah median (fitur tidak relevan) akan dihilangkan. Dengan langkah-langkah: 1) Hitung nilai entropy setiap data yang sudah di transformasi, 2) Hitung nilai gain setiap data, 3) Hitung median kumpulan data dengan median untuk kumpulan data genap yaitu diambil di titik tengah antara angka dengan kedua nilai di jumlah dan di bagi 2 dan median untuk kumpulan data ganjil diambil dari nilai yang berada di titik tengah 4) Bentuk dataset baru hasil transformasi dan seleksi fitur Median WIG dengan mengeliminasi fitur yang kurang dari nilai median.
4. Melakukan proses training dan testing data dengan dataset *Software Effort Estimation* yang baru hasil transformasi dan seleksi fitur menggunakan algoritma k-NN dengan menentukan nilai  $k = 1$  serta metrik jarak Euclidean distance k-NN (k-NN Euc)
5. Validasi metode yang dilakukan menggunakan leave-one-out cross validation (LOOCV)

Selanjutnya untuk memperkuat hasil metode yang diusulkan dilakukan percobaan dan pengujian metode dengan metode lain dalam dengan tahapan sebagai berikut:

1. Melakukan proses pengujian metode yang diusulkan menggunakan dataset *Software Effort Estimation* yang memiliki label *hours*.
2. Melakukan proses pengujian metode individual k-Nearest Neighbor dataset *Software Effort Estimation* yang memiliki label *hours*.
3. Melakukan pembahasan perbandingan hasil estimasi menggunakan nilai RMSE antara metode individual k-Nearest Neighbor dengan k-Nearest Neighbor dan metode usulan seleksi fitur Median Weighted Information Gain (Median WIG) dengan k-Nearest Neighbor.

### 3. HASIL DAN PEMBAHASAN

#### 3.1. Hasil Pengujian Dataset dengan Metode Median WIG k-NN

Hasil perhitungan pada dataset Albrecht diperoleh nilai median sebesar 0.197 sehingga pada dataset Albrecht terdapat 3 fitur yang akan dihilangkan dikarenakan memperoleh nilai dibawah median yang sudah ditentukan (fitur tidak relevan) yaitu fitur dengan label Input, Inquiry dan File. Serta diperoleh 4 fitur diatas median (fitur relevan) yaitu fitur dengan label Output, FPAdj, RawFPcounts, AdjFP serta 1 label Effort yang akan dipertahankan untuk di jadikan dataset baru Albrecht yang akan digunakan dalam pengujian menggunakan Algoritma k-NN di *rapidminer*.

Pengujian dataset baru menggunakan *rapidminer* dengan algoritma k-NN diperoleh Hasil perhitungan dengan parameter k-NN menggunakan k = 1 dan perhitungan jarak Euclidean distance pada Dataset Albrecht diperoleh hasil perhitungan nilai RMSE sebesar 5.958.

Hasil perhitungan pada dataset Miyazaki diperoleh nilai median sebesar 0.109 sehingga terdapat 3 fitur dataset Miyazaki yang akan dihilangkan dikarenakan memperoleh nilai dibawah median yang sudah ditentukan (fitur tidak relevan) yaitu fitur dengan label KLOC, ESCRN, EFILE. Serta diperoleh 4 fitur diatas median (fitur relevan) yaitu fitur dengan label SCRN, FORM, FILE, EFORM serta 1 label MM yang akan dipertahankan untuk di jadikan dataset baru Miyazaki dalam pengujian menggunakan Algoritma k-NN di *rapidminer*.

Pengujian dataset baru dengan algoritma k-NN menggunakan *rapidminer* diperoleh hasil perhitungan dengan parameter k-NN menggunakan k = 1 dan perhitungan jarak Euclidean distance pada Dataset Miyazaki diperoleh hasil perhitungan nilai RMSE sebesar 62.363.

Hasil perhitungan pada dataset Kemerer diperoleh nilai median sebesar 0.221 sehingga pada dataset Kemerer terdapat 3 fitur yang akan dihilangkan dikarenakan nilai dibawah median yang sudah ditentukan (fitur tidak relevan) yaitu fitur dengan label Language, Hardware, KSLOC. Serta diperoleh 3 fitur diatas median (fitur relevan) yaitu fitur dengan label Duration, AdjFP, RAWFP serta 1 label EffortMM yang akan dipertahankan untuk di jadikan dataset baru Kemerer dalam pengujian menggunakan Algoritma k-NN di *rapidminer*.

Pengujian dataset baru dengan algoritma k-NN menggunakan *rapidminer* diperoleh hasil perhitungan dengan dengan parameter k-NN menggunakan k = 1 dan perhitungan jarak Euclidean distance pada Dataset Kemerer diperoleh hasil perhitungan nilai RMSE sebesar 131.027.

Seluruh hasil pengujian dengan metode seleksi fitur Median WIG dan algoritma k-Nearest Neighbor dengan perhitungan nilai RMSE pada dataset *Software Effort Estimation* yang memiliki label months sebagai berikut:

Tabel 1. Hasil Nilai RMSE Median-WIG k-Nearest Neighbor

Dataset	Median-WIG k-NN
Albrecht	5.958
Miyazaki	62.363
Kemerer	131.027

#### 3.2. Hasil Pengujian Metode Individual Algoritma k-NN

Hasil perhitungan pada seluruh dataset dengan alat bantu *Rapidminer* menggunakan metode k-NN dengan menetapkan parameter k-NN menggunakan k = 1 dan menggunakan perhitungan jarak Euclidean distance diperoleh hasil perhitungan nilai RMSE pada dataset *Software Effort Estimation* dengan label months sebagai berikut:

Tabel 2. Hasil Nilai RMSE Metode Kombinasi k-NN

Dataset	k-NN
Albrecht	6.583
Miyazaki	68.096
Kemerer	143.634

#### 3.3. Pembahasan Komparasi Nilai RMSE antara Median WIG k-NN dengan k-NN

Dalam mengukur kinerja suatu metode terbaik dapat dilakukan dengan membandingkan hasil nilai RMSE dari seluruh metode yang di ujikan dengan menggunakan k = 1 sebagai parameter k-NN dan perhitungan jarak Euclidean Distance. Hasil pengujian yang diperoleh dengan metode k-NN dan metode usulan seleksi fitur Median Weighted Information Gain (Median WIG) dengan k-Nearest Neighbor dengan hasil nilai RMSE seperti pada tabel berikut:

Tabel 3. Komparasi Hasil Pengujian Keseluruhan Metode

Dataset	Median-WIG k-NN	k-NN
Albrecht	<b>5.958</b>	6.583
Miyazaki	<b>62.363</b>	68.096
Kemerer	<b>131.027</b>	143.634

Hasil yang diperoleh dalam pengujian memperlihatkan bahwa metode k-NN memperoleh nilai RMSE yang lebih tinggi dibandingkan dengan metode usulan yaitu integrasi seleksi fitur dan k-NN, hal tersebut dapat menunjukkan bahwa dataset *Software Effort Estimation* memiliki fitur yang kurang relevan yang tidak dapat diatasi dengan metode individual k-NN dan fitur yang kurang relevan tersebut sangat mempengaruhi dari akurasi estimasi yang dilakukan. Sehingga proses membuang fitur yang tidak relevan terbukti mampu menaikkan akurasi dari estimasi.

Hasil perbandingan yang diperoleh dalam seluruh pengujian dataset *Software Effort Estimation* diketahui bahwa metode usulan seleksi fitur Median-WIG dengan k-NN terbukti mampu meningkatkan akurasi dalam proses estimasi dilihat dari penurunan nilai RMSE yang signifikan pada seluruh dataset yaitu Albrecht sebesar 5.958, Miyazaki sebesar 62.363 dan Kemerer sebesar 131.027. Terjadinya penurunan nilai RMSE pada seluruh dataset ini memperlihatkan peningkatan kinerja setelah dilakukannya metode seleksi fitur pada dataset *Software Effort Estimation*. Dataset *Software Effort Estimation* terbukti memiliki fitur yang tidak relevan sehingga perlu dihilangkan untuk memperoleh hasil estimasi yang lebih baik.

Hal ini mempertegas hasil penelitian-penelitian terdahulu tentang fitur yang tidak relevan sangat mempengaruhi akurasi dalam melakukan estimasi. Penggunaan metode seleksi fitur dengan menghilangkan fitur yang tidak relevan sebelum dilakukannya pengujian dengan sebuah algoritma mampu meningkatkan akurasi yang dapat dilihat dari semakin kecil nilai RMSE dari sebuah estimasi memanfaatkan dataset.

#### 4. KESIMPULAN

Kesimpulan yang diperoleh berdasarkan hasil pengujian dalam penelitian ini yaitu:

- 1) Fitur yang tidak relevan sangat berpengaruh pada hasil estimasi dataset label hours *Software Effort Estimation*.
- 2) Metode seleksi fitur lebih baik dari metode individu k-NN dikarenakan dapat mengatasi masalah fitur yang tidak relevan.
- 3) Metode seleksi fitur dapat meningkatkan akurasi estimasi dilihat dari semakin kecil nilai RMSE.

#### DAFTAR PUSTAKA

- [1] A. Trendowicz and R. Jeffery, *Software Project Effort Estimation*. New York: Springer, 2014.
- [2] J. Wen, S. Li, Z. Lin, Y. Hu, and C. Huang, "Systematic literature review of machine learning based software development effort estimation models," *Inf. Softw. Technol.*, vol. 54, no. 1, pp. 41–59, 2012, doi: 10.1016/j.infsof.2011.09.002.
- [3] H. Mukhtar, R. Muhammad, T. Reny Medikawati, and Yoze Rizki, "Peramalan Kedatangan Wisatawan Mancanegara Ke Indonesia Menurut Kebangsaan Perbulannya Menggunakan Metode Multilayer Perceptron," *J. CoSciTech (Computer Sci. Inf. Technol.)*, vol. 2, no. 2, pp. 113–119, 2021, doi: 10.37859/coscitech.v2i2.3324.
- [4] M. Shepperd and S. Macdonell, "Evaluating prediction systems in software project estimation," *Inf. Softw. Technol.*, vol. 54, no. 8, pp. 820–827, 2012, doi: 10.1016/j.infsof.2011.12.008.
- [5] X. Huang, D. Ho, J. Ren, and L. F. Capretz, "Improving the COCOMO model using a neuro-fuzzy approach," *Appl. Soft Comput.*, vol. 7, pp. 29–40, 2007, doi: 10.1016/j.asoc.2005.06.007.
- [6] S. Grimstad and M. Jørgensen, "Inconsistency of expert judgment-based estimates of software development effort," *J. Syst. Softw.*, vol. 80, pp. 1770–1777, 2007, doi: 10.1016/j.jss.2007.03.001.
- [7] G. R. Finnie and G. E. Wittig, "A Comparison of Software Effort Estimation Techniques : Using Function Points with Neural Networks , Case-Based Reasoning and Regression Models," *J. Systems Softw.*, vol. 1212, no. 97, pp. 281–289, 1997.
- [8] S. M. Satapathy, B. P. Acharya, and S. K. Rath, "Early stage software effort estimation using random forest technique based on use case points," *IET Softw.*, vol. 10, no. 1, pp. 10–17, 2016, doi: 10.1049/iet-sen.2014.0122.
- [9] E. K. Adhitya, R. Satria, and H. Subagyo, "Komparasi Metode Machine Learning dan Non Machine Learning untuk Estimasi Usaha Perangkat Lunak," *J. Softw. Eng.*, vol. 1, no. 2, pp. 109–113, 2015.
- [10] A. Bakır, B. Turhan, and A. Bener, "A comparative study for estimating software development effort intervals," *Softw. Qual J.*, vol. 19, pp. 537–552, 2011, doi: 10.1007/s11219-010-9112-9.
- [11] V. Khatibi, B. Dayang, and N. Abang, "A PSO-based model to increase the accuracy of software development effort estimation," *Softw. Qual J.*, vol. 21, pp. 501–526, 2013, doi: 10.1007/s11219-012-9183-x.
- [12] Q. Liu, J. Xiao, and H. Zhu, "Feature selection for software effort estimation with localized neighborhood mutual information," *Cluster Comput.*, no. 1, 2018, doi: 10.1007/s10586-018-1884-x.
- [13] I. Kurniawan, "Kombinasi Median Weighted Information Gain Dengan K-Nearest Neighbor Pada Dataset Label Months Software Effort Estimation," *J. Teknoinfo*, vol. 14, no. 2, p. 138, 2020, doi: 10.33365/jti.v14i2.647.
- [14] I. Kurniawan and A. F. Abror, "Komparasi Metode Kombinasi Seleksi Fitur dan Machine Learning K-Nearest Neighbor Pada Dataset Label Hours Software Effort Estimation," *J. Sist. Inf. dan Telemat. (Telekomunikasi, Multimedia, dan Informasi)*, vol. 10, no. 2, pp. 83–89, 2019.
- [15] A. L. I. Oliveira, P. L. Braga, R. M. F. Lima, and M. L. Cornélio, "GA-based method for feature selection and parameters optimization for machine learning regression applied to software effort estimation," *Inf. Softw. Technol.*, vol. 52, no. 11, pp. 1155–1166, 2010, doi: 10.1016/j.infsof.2010.05.009.
- [16] J. Shivhare and S. K. Rath, "Software Effort Estimation using Machine Learning Techniques," *ISEC*, 2014.
- [17] F. Zare, H. Khademi Zare, and M. S. Fallahnezhad, "Software effort estimation based on the optimal Bayesian belief network," *Appl. Soft Comput.*, vol.

- 49, pp. 968–980, 2016, doi: 10.1016/j.asoc.2016.08.004.
- [18] A. Idri and S. Cherradi, “Improving Effort Estimation of Fuzzy Analogy using Feature Subset Selection,” 2016.
- [19] Y. Fatma, Januar Al Amien, and Riski Hakiki, “Sistem Pendukung Keputusan Penilaian Pegawai Di Klinik Bunda Medical Center (BMC) Dengan Menggunakan Metode Simple Additive Weighting (SAW),” *J. CoSciTech (Computer Sci. Inf. Technol.*, vol. 2, no. 2, pp. 1–9, 2022, doi: 10.37859/coscitech.v2i2.2961.
- [20] T. M. Khoshgoftaar and K. Gao, “Feature Selection with Imbalanced Data for Software Defect Prediction,” in *International Conference on Machine Learning and Applications*, 2009, pp. 235–240, doi: 10.1109/ICMLA.2009.18.
- [21] F. Yu and X. Xu, “A short-term load forecasting model of natural gas based on optimized genetic algorithm and improved BP neural network,” *Appl. Energy*, vol. 134, pp. 102–113, 2014, doi: 10.1016/j.apenergy.2014.07.104.