

FA-KNN: HYBRID ALGORITMA UNTUK KLASIFIKASI PENYAKIT DIABETES MELITUS

Dedy Abdianto Nggego*¹, Rahmat Taufik R.L Bau², Nilfred Patawaran³, Marsujitullah⁴
Program Studi Teknik Informatika Universitas Musamus Merauke^{1,3,4}
Program Studi Sistem Informasi Universitas Negeri Gorontalo²
dedyabdianto@unmus.ac.id

Abstrak

Penanganan yang tepat dan tepat waktu dari Diabetes Melitus menjadi sangat penting karena penyakit ini dapat menyebabkan berbagai komplikasi serius. Komplikasi jangka panjang meliputi gangguan pada mata (retinopati), ginjal (nefropati), saraf (neuropati), jantung dan pembuluh darah (kardiovaskular), serta risiko luka yang sulit sembuh hingga amputasi pada ekstremitas. Penelitian ini bertujuan untuk menerapkan Firefly Algorithm (FA) atau algoritma kunang-kunang dan KNN dalam melakukan klasifikasi terhadap penyakit diabetes melitus, dimana FA akan digunakan untuk melakukan pencarian parameter yang paling optimal untuk KNN. Metode penelitian yang digunakan yaitu metode eksperimen dengan melakukan skenario perubahan pada jumlah populasi kunang-kunang dan juga perubahan nilai k-fold validation untuk melakukan pembagian dataset. Hasil akurasi terbaik didapatkan pada populasi 100 dan 150 dengan nilai k=5 yaitu sebesar 76.3% dengan parameter K pada KNN yang diperoleh yaitu 15 dan P adalah 2.

Kata kunci— *firefly algorithm, knn, machine learning, classification.*

Abstract

Proper and timely treatment of Diabetes Mellitus is very important because this disease can cause serious complications. Long-term complications include disorders of the eyes (retinopathy), kidneys (nephropathy), nerves (neuropathy), heart and blood vessels (cardiovascular), as well as the risk of wounds that are difficult to heal up to amputation of the extremities. This study aims to apply the Firefly Algorithm (FA) or the firefly algorithm and KNN in classifying diabetes mellitus, where FA will be used to search for the most optimal parameters for KNN. The research method used is the experimental method by carrying out a scenario of changing the number of firefly populations and also changing the k-fold validation value to do a split dataset. The best accuracy results were obtained in populations 100 and 150 with a value of k = 5, which is 76.3% with the K parameter in the KNN obtained, namely 15 and P is 2.

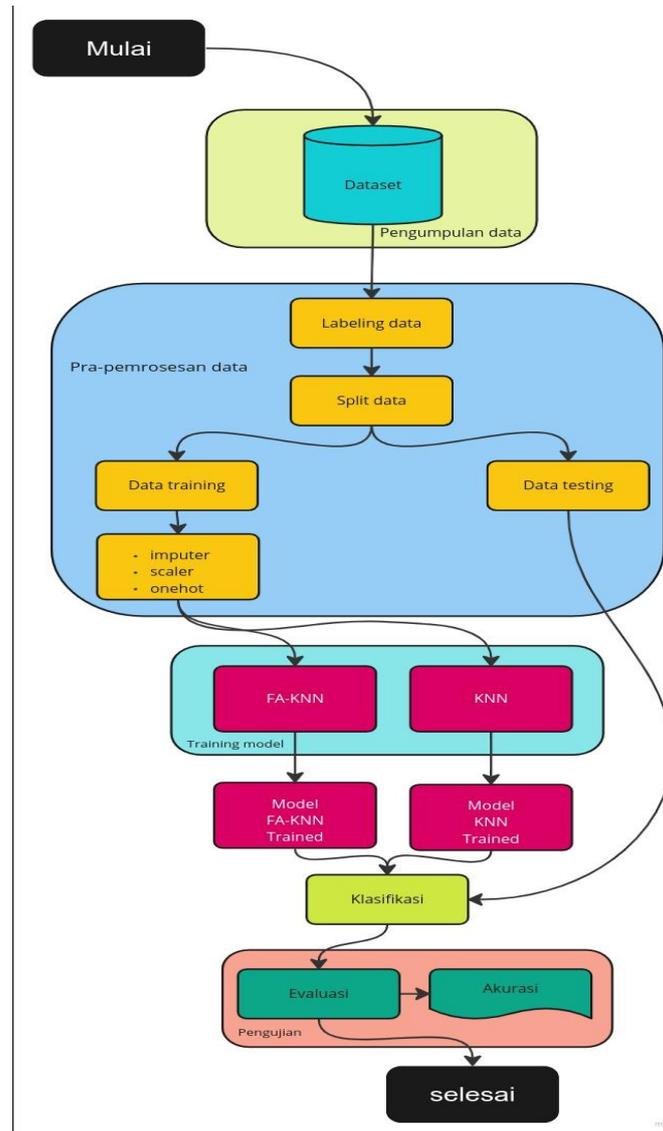
Keywords— *firefly algorithm, knn, machine learning, classification.*

1. PENDAHULUAN

Diabetes adalah penyakit yang sangat umum ditemukan di seluruh dunia. Menurut data International Diabetes Federation pada tahun 2021 penderita diabetes di Indonesia mencapai hampir 20 juta orang, dan diperkirakan akan mencapai lebih dari 23 juta orang pada tahun 2030 [1]. Diabetes terjadi ketika tubuh tidak dapat memproduksi atau menggunakan insulin dengan baik, yang menyebabkan peningkatan kadar gula darah dalam tubuh. Ada beberapa jenis diabetes, yaitu diabetes tipe 1, diabetes tipe 2, dan diabetes gestasional. Diabetes tipe 1 terjadi ketika sistem kekebalan tubuh menyerang dan menghancurkan sel beta di pankreas yang memproduksi insulin. Diabetes tipe 2 terjadi ketika tubuh tidak dapat menggunakan insulin secara efektif dan secara bertahap menghambat pankreas membuat insulin. Sedangkan diabetes gestasional terjadi pada ibu hamil yang tidak memiliki riwayat diabetes sebelumnya. Klasifikasi diabetes penting untuk membantu dokter dalam menegakkan diagnosis dan memberikan pengobatan yang tepat. Diagnosis yang akurat dapat membantu mencegah kemungkinan komplikasi pada penderita diabetes, seperti kerusakan saraf, masalah ginjal, kerusakan mata, dan masalah kardiovaskular [2]. Oleh karena itu, mengembangkan model klasifikasi penyakit untuk diabetes berbasis machine learning dapat membantu meningkatkan pengenalan dan diagnosis penyakit, sehingga membantu dokter memberikan perawatan yang lebih baik dan tepat waktu kepada pasien. Model klasifikasi juga dapat membantu meningkatkan kesadaran akan penyakit dan faktor risiko yang terkait dengan diabetes, yang dapat membantu mencegah dan mengelola penyakit tersebut. Teknik klasifikasi dan juga machine learning telah diterapkan pada berbagai bidang, antara lain pendidikan, jaringan komputer, konstruksi bangunan, kesehatan, dan masih banyak lainnya [3]–[9]. Klasifikasi dan juga penerapan machine learning pada penyakit diabetes juga dilakukan dengan menggunakan metode yang berbeda-beda seperti *Naive Bayes*, *Support Vector Machine*, *Adaboost Classifier*, dan sebagainya [3], [8], [10]–[15]. Salah satu metode klasifikasi yang banyak digunakan adalah *K-Nearest Neighbors* (KNN). KNN merupakan salah satu metode atau algoritma klasifikasi yang cukup baik [16]. Algoritma ini juga dinilai cukup efektif dan memiliki performa yang baik dalam melakukan klasifikasi pada data yang besar [11]. Penelitian [17] menerapkan metode KNN dalam melakukan klasifikasi untuk menentukan prioritas bantuan pembangunan desa, dengan melakukan optimasi pada parameter K . Penentuan nilai K pada metode KNN ini cukup penting untuk dilakukan karena akan mempengaruhi hasil klasifikasi. Hasil klasifikasi akan semakin kabur apabila nilai K terlalu besar, sedangkan untuk nilai K yang bernilai 1 akan mengakibatkan klasifikasi yang dilakukan terasa kaku karena hanya akan menggunakan 1 tetangga paling dekat [18]. Selain melakukan optimasi pada nilai K , hal yang dapat dilakukan untuk meningkatkan akurasi KNN yaitu dengan menerapkan *feature selection* pada sebaran data yang dimiliki [19], [20]. Untuk melakukan pencarian parameter tersebut, dapat dilakukan dengan menggunakan bantuan algoritma metaheuristik seperti *Particle Swarm Optimization* (PSO), *Genetic Algorithm* (GA), dan *Firefly Algorithm* (FA) [6], [21]–[23]. Penelitian ini akan menerapkan FA dan KNN untuk melakukan klasifikasi terhadap penyakit diabetes melitus, dimana FA akan digunakan untuk melakukan pencarian parameter yang paling optimal untuk KNN. Setiap parameter yang dihasilkan oleh FA akan digunakan oleh KNN untuk selanjutnya akan dihitung masing-masing akurasi klasifikasinya sehingga akan ditemukan parameter dengan nilai akurasi paling tinggi.

2. METODE PENELITIAN

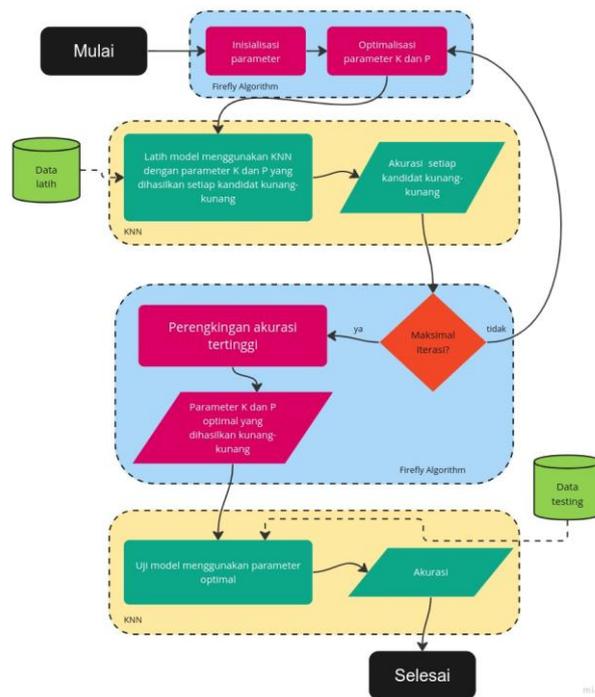
Penelitian ini menggunakan metode eksperimen, dimana training data akan dilakukan dengan menggunakan parameter hasil optimasi. Ada 5 tahapan yang dilakukan dalam penelitian ini yaitu pengumpulan dataset, pre-pemrosesan data, training data, klasifikasi, dan pengujian, seperti yang dituangkan pada Gambar 1.



Gambar 1. Alur Penelitian

Dataset yang digunakan pada penelitian ini diperoleh dari dataset publik resmi *UCI*. Dataset tersebut akan dibagi menjadi data training dan data testing, dimana data training akan dilakukan pra-pemrosesan data dengan melakukan *imputer*, *scaller*, dan *onehot encoding* untuk memperbaiki kesalahan pada data dan mengubah format menjadi lebih teratur. Pada tahap selanjutnya, akan dilakukan training model dengan menggunakan KNN serta training model menggunakan FA-KNN. Selanjutnya kedua model ini akan melakukan klasifikasi, kemudian hasil dari klasifikasi tersebut akan dievaluasi sehingga diperoleh perbandingan akurasi kedua metode tersebut.

Proses *hybrid* metode FA-KNN dapat dilihat pada Gambar 3.



Gambar 3. Proses klasifikasi FA-KNN

3. TINJAUAN PUSTAKA

1. *Firefly Algorithm (FA)*

Firefly algorithm atau algoritma kunang-kunang merupakan salah satu algoritma metaheuristik yang terinspirasi dari perilaku kunang-kunang di alam bebas. Algoritma ini bekerja berdasarkan perilaku kunang-kunang yang mengandalkan daya tarik kecerahan antar kunang-kunang. Aturan algoritma kunang-kunang dirumuskan oleh [24] dengan asumsi sebagai berikut.

1. Seluruh kunang-kunang adalah *unisex*, artinya setiap kunang-kunang akan saling tertarik tanpa melihat jenis kelamin
2. Daya tarik kunang-kunang didasarkan pada intensitas kecerahan, dengan ketentuan tingkat kecerahan kunang-kunang dipengaruhi oleh jarak antara kunang-kunang. Semakin jauh jarak antara kunang-kunang, maka tingkat kecerahan semakin menurun atau menghilang. Kunang-kunang yang lebih redup akan mendekati kunang-kunang yang lebih terang. Apabila di antara kedua kunang-kunang tidak ada yang lebih terang, maka kunang-kunang akan bergerak secara acak.
3. Kecerahan kunang-kunang akan ditentukan oleh fungsi *fitness*.

Ada dua faktor penting yang terdapat pada algoritma kunang-kunang, yaitu intensitas cahaya dan daya tarik antar kunang-kunang [25]. Pada contoh optimasi, intensitas cahaya x diperoleh dari persamaan (1).

$$I(x) = f(x) \tag{1}$$

Daya tarik setiap kunang-kunang dapat dihitung dengan menggunakan persamaan (2).

$$\beta(r) = \beta_0 e^{-\gamma r^2} \tag{2}$$

Dimana β_0 merupakan daya tarik pada jarak ($r = 0$) saat tidak ada jarak antar kunang-kunang, dan kadang-kadang dalam perhitungan matematis dianggap sebagai 1 [26]. γ merepresentasikan nilai penyerapan cahaya. Jarak antara dua kunang-kunang i dan j disimbolkan dengan r . jarak antara dua kunang-kunang i dan j masing-masing adalah jarak kartesian yang dapat dihitung menggunakan hukum jarak *Euclidean*.

$$r_{ij} = \|x_i - x_j\| = \sqrt{\sum_{k=0}^d (x_{i,k} - x_{j,k})^2} \tag{3}$$

Dimana d merupakan dimensi dari permasalahan, sedangkan $x_{i,k}$ merupakan komponen ke- k dari x_i pada kunang-kunang i . Setelah melakukan perhitungan jarak antara kedua kunang-kunang tersebut, misalkan kunang-kunang i memiliki kecerahan lebih kecil dari kunang-kunang j , maka akan terjadi daya tarik menarik dimana kunang-kunang i akan berpindah menuju kunang-kunang j . Dengan adanya pergerakan tersebut, maka posisi dari kunang-kunang akan berubah dan dihitung berdasarkan persamaan (4).

$$x_i = x_i + \beta_0 e^{-\gamma r_{ij}^2} (x_j - x_i) + \alpha \left(rand - \frac{1}{2} \right) \tag{4}$$

Kunang-kunang dengan kecerahan yang rendah akan berpindah menuju kunang-kunang dengan kecerahan yang lebih tinggi. Suku pertama pada persamaan (4) merupakan posisi lama dari kunang-kunang, suku kedua terjadi karena ketertarikan, suku kedua adalah gerakan menuju kunang-kunang yang memiliki kecerahan lebih tinggi dengan koefisien tarikan β , sedangkan suku ketiga merupakan pergerakan acak yang merupakan perkalian generator acak *rand* dengan α , dimana *rand* adalah bilangan riil acak pada interval (0,1) [27]. *Pseudocode* dari algoritma kunang-kunang adalah seperti berikut.

```

Objective function  $f(x)$ ,  $x = (x_1, \dots, x_d)^T$ 
Generate an initial population of  $n$  fireflies  $x_i$ , ( $i = 1, 2, \dots, n$ )
Light intensity  $I_i$  at  $x_i$  is determined by  $f(x_i)$ 
Define light absorption coefficient  $\gamma$ .
While ( $t < MaxGeneration$ )
  for  $i = 1 : n$  all  $n$  fireflies
    for  $j = 1 : n$  all  $n$  fireflies (inner loop)
      if ( $I_i < I_j$ )
        Move firefly  $i$  towards  $j$ 
      end if
      Vary attractiveness with distance  $r$  via  $\exp[-\gamma r^2]$ 
      Evaluate new solutions and update light intensity
    End for j
  End for i
  Rank the fireflies and find the current global best  $g$ 
End while
Post-process result

```

Gambar 2. Pseudocode FA [28]

2. *K-Nearest Neighbors (KNN)*

K-Nearest Neighbors (KNN) merupakan salah satu algoritma klasifikasi dan regresi yang sederhana namun populer dalam pembelajaran mesin. Algoritma ini digunakan untuk memprediksi kelas atau nilai dari sebuah data berdasarkan data latih yang paling mirip dengan data tersebut. KNN bekerja berdasarkan asumsi bahwa data yang memiliki atribut yang mirip cenderung memiliki label yang sama atau nilai yang serupa. Prinsip kerja dari KNN adalah sebagai berikut.

1. Menentukan parameter K: KNN bekerja dengan mencari K tetangga terdekat dari data yang akan diprediksi. Parameter K adalah jumlah tetangga terdekat yang akan diambil dalam proses prediksi. Jumlah K harus merupakan bilangan bulat positif. Pemilihan nilai K yang tepat merupakan aspek penting dalam KNN, karena nilai K yang terlalu kecil dapat menyebabkan model menjadi terlalu sensitif terhadap noise, sementara nilai K yang terlalu besar dapat menyebabkan model menjadi terlalu acak dan mengabaikan pola yang sebenarnya.
2. Menghitung Jarak: Untuk mencari K tetangga terdekat, KNN menghitung jarak antara data yang akan diprediksi (data uji) dengan semua data latih yang ada. Jarak ini bisa dihitung dengan menggunakan metrik jarak seperti Euclidean, Manhattan, atau Minkowski. Umumnya, Euclidean distance digunakan untuk data numerik, sementara metrik jarak lainnya dapat digunakan untuk data dengan atribut yang berbeda.
3. Memilih Tetangga Terdekat: Setelah menghitung jarak, KNN akan memilih K data latih yang memiliki jarak terdekat dengan data uji. Data latih ini akan menjadi tetangga terdekat yang digunakan dalam proses klasifikasi atau regresi.
4. Voting atau Penghitungan Rata-rata: Jika KNN digunakan untuk klasifikasi (Klasifikasi KNN), hasil akhir dari prediksi ditentukan oleh mayoritas label dari tetangga terdekat. Dengan kata lain, label yang paling sering muncul di antara K tetangga tersebut akan menjadi label prediksi. Jika KNN digunakan untuk regresi (Regresi KNN), hasil prediksi akan dihitung dengan mengambil rata-rata nilai dari tetangga terdekat.

Persamaan K-Nearest Neighbors (KNN) digunakan untuk menghitung jarak antara data yang akan diprediksi dengan data latih. Berikut adalah persamaan KNN untuk kasus klasifikasi, dengan menggunakan metrik jarak Euclidean:

$$d_{Euclidean}(x, y) = |x - y| = \sqrt{\sum_i^n (x_i - y_i)^2}$$

Misalkan

1. x_i adalah data latih ke-i dengan atribut (fitur) $x_i = (x_{i1}, x_{i2}, \dots, x_{in})$, di mana n adalah jumlah fitur.
2. y_i adalah label kelas dari data latih ke-i.

Penting untuk dicatat bahwa dalam KNN, nilai K merupakan parameter yang harus ditentukan sebelumnya. Memilih nilai K yang tepat dapat mempengaruhi performa algoritma KNN, dan ini dapat diuji dengan melakukan validasi silang atau teknik lainnya untuk evaluasi model. Dalam implementasi nyata, KNN sering menggunakan variasi metrik jarak selain Euclidean, tergantung pada karakteristik data dan masalah yang dihadapi. Selain itu, ada juga metode pemilihan K yang berbeda, seperti menggunakan cross-validation untuk menentukan K yang optimal [29], [30].

4. HASIL DAN PEMBAHASAN

Dataset yang digunakan pada penelitian ini berasal dari UCI dataset yaitu Pima Indians Diabetes Database. Pada penelitian ini digunakan skenario pengujian yaitu dengan melakukan eksperimen pada populasi dari FA dengan nilai populasi 50, 100, dan 150, kemudian skenario dilakukan pada pengaplikasian *k-fold* dengan nilai *k* masing-masing yaitu 3, 5, dan 7. Tabel 1 merupakan hasil pengujian model FA-KNN untuk mengetahui pengaruh populasi terhadap akurasi yang dihasilkan dengan menggunakan nilai $k=3$.

Tabel 1. Pengujian skenario populasi dengan nilai $k=3$

Populasi	K-Fold	Accuracy
50	3	75.6%
100	3	75.6%
150	3	75.6%

Berdasarkan Tabel 1 hasil dari tersebut menunjukkan bahwa tidak terjadi perubahan akurasi pada masing-masing populasi

Tabel 2. Pengujian skenario populasi dengan nilai $k=5$

Populasi	K-Fold	Accuracy
50	5	75.9%
100	5	76.3%
150	5	76.3%

Hasil pengujian pada Tabel 2 menunjukkan bahwa terjadi peningkatan akurasi pada populasi 100 dan 150 dengan peningkatan akurasi lebih kurang 0.4%.

Tabel 3. Pengujian skenario populasi dengan nilai $k=7$

Populasi	K-Fold	Accuracy
50	7	75.2%
100	7	75.3%
150	7	75.3%

Berdasarkan Tabel 3 diperoleh hasil peningkatan akurasi pada populasi 100 dan 150 sebesar lebih kurang 0.1% akurasi.

Pada pengujian menggunakan KNN tanpa FA diperoleh akurasi seperti dituangkan pada Tabel 4.

Tabel 4. Pengujian menggunakan KNN tanpa FA

K-Fold	Accuracy
3	73.8%
5	75.6%
7	74.8%

Berdasarkan Tabel 4 diperoleh nilai akurasi tertinggi dari KNN yaitu 75.6%, sedangkan pada FA-KNN didapatkan akurasi tertinggi yaitu 76.3% atau 0.7% lebih tinggi dibanding dengan KNN. Hasil ini cukup membuktikan bahwa FA berhasil melakukan optimasi performa KNN.

Hasil akurasi terbaik didapatkan pada populasi 100 dan 150 dengan nilai $k=5$ yaitu sebesar 76.3% dengan parameter K pada KNN yang diperoleh yaitu 15 dan P adalah 2. Ukuran populasi atau jumlah kunang-kunang dapat mempengaruhi performa algoritma. Semakin besar populasi, semakin banyak kemungkinan solusi yang dijelajahi, namun akan meningkatkan kompleksitas perhitungan dan waktu eksekusi. Sebaliknya, populasi yang terlalu kecil dapat menyebabkan kurangnya variasi dalam solusi dan mempengaruhi efektivitas pencarian global. Nilai k pada k -fold validation juga memiliki pengaruh terhadap evaluasi model pada penelitian ini. Nilai k yang lebih kecil akan menghasilkan estimasi akurasi model yang memiliki varian yang lebih tinggi karena model hanya diuji pada beberapa partisi data saja. Akibatnya, perkiraan kinerja model dapat menjadi kurang stabil.

5. KESIMPULAN

Penelitian ini berhasil melakukan hibryd algoritma FA dan KNN untuk klasifikasi penyakit diabetes melitus dengan tingkat akurasi sebesar 76.3%, dan diperoleh parameter terbaik K pada KNN yaitu 15 dan P yaitu 2. Penentuan jumlah populasi pada FA cukup penting untuk mencapai keseimbangan yang baik antara eksplorasi dan eksploitasi, sehingga dapat mencapai solusi yang lebih optimal dalam waktu yang wajar.

DAFTAR PUSTAKA

- [1] "IDF Diabetes Atlas 10th edition 2021," Nov. 2021. [Online]. Available: <https://diabetesatlas.org/data/en/country/94/id.html>
- [2] Perkumpulan Endokrinologi Indonesia, "KONSENSUS PENGELOLAAN DAN PENCEGAHAN DIABETES MELITUS TIPE 2 DI INDONESIA." 2011.
- [3] V. Rawat and Suryakant, "A Classification System for Diabetic Patients with Machine Learning Techniques," *Int. J. Math. Eng. Manag. Sci.*, vol. 4, no. 3, pp. 729–744, Jun. 2019, doi: 10.33889/IJMEMS.2019.4.3-057.
- [4] A. H. M. Ragab, A. Y. Noaman, A. S. Al-Ghamdi, and A. I. Madbouly, "A Comparative Analysis of Classification Algorithms for Students College Enrollment Approval Using Data Mining," in *Proceedings of the 2014 Workshop on Interaction Design in Educational Environments*, Albacete Spain: ACM, Jun. 2014, pp. 106–113. doi: 10.1145/2643604.2643631.
- [5] Y. A. Christobel and P. Sivaprakasam, "A New Classwise k Nearest Neighbor (CKNN) Method for the Classification of Diabetes Dataset," vol. 2, no. 3.
- [6] W. Yunus, "Algoritma K-Nearest Neighbor Berbasis Particle Swarm Optimization Untuk Prediksi Penyakit Ginjal Kronik".
- [7] C. Shahnaz, J. Hossain, S. A. Fattah, S. Ghosh, and A. I. Khan, "Efficient approaches for accuracy improvement of breast cancer classification using wisconsin database," in *2017 IEEE Region 10 Humanitarian Technology Conference (R10-HTC)*, Dhaka: IEEE, Dec. 2017, pp. 792–797. doi: 10.1109/R10-HTC.2017.8289075.
- [8] G. Abdurrahman, "Klasifikasi Penyakit Diabetes Melitus Menggunakan Adaboost Classifier," *JUSTINDO J. Sist. Dan Teknol. Inf. Indones.*, vol. 7, no. 1, pp. 59–66, Mar. 2022, doi: 10.32528/justindo.v7i1.4949.
- [9] J. Chan, D. Jayasuriya, and D. Sundaram, "Machine Learning Application in Healthcare: Breast Cancer Diagnosis and Prognosis," *SSRN Electron. J.*, 2020, doi: 10.2139/ssrn.4211998.

- [10]L. U. Khasanah and Y. N. Nasution, “Klasifikasi Penyakit Diabetes Melitus Menggunakan Algoritma Naïve Bayes Classifier,” 2022.
- [11]N. M. Putry, B. N. Sari, and M. Kom, “KOMPARASI ALGORITMA KNN DAN NAÏVE BAYES UNTUK KLASIFIKASI DIAGNOSIS PENYAKIT DIABETES MELITUS,” vol. 10, no. 1, 2022.
- [12]H. Apriyani and K. Kurniati, “Perbandingan Metode Naïve Bayes Dan Support Vector Machine Dalam Klasifikasi Penyakit Diabetes Melitus,” *J. Inf. Technol. Ampera*, vol. 1, no. 3, pp. 133–143, Dec. 2020, doi: 10.51519/journalita.volume1.iss3.year2020.page133-143.
- [13]A. Yahyaoui, A. Jamil, J. Rasheed, and M. Yesiltepe, “A Decision Support System for Diabetes Prediction Using Machine Learning and Deep Learning Techniques,” in *2019 1st International Informatics and Software Engineering Conference (UBMYK)*, Ankara, Turkey: IEEE, Nov. 2019, pp. 1–4. doi: 10.1109/UBMYK48245.2019.8965556.
- [14]G. T. Reddy *et al.*, “An Ensemble based Machine Learning model for Diabetic Retinopathy Classification,” in *2020 International Conference on Emerging Trends in Information Technology and Engineering (ic-ETITE)*, Vellore, India: IEEE, Feb. 2020, pp. 1–6. doi: 10.1109/ic-ETITE47903.2020.235.
- [15]Md. K. Hasan, Md. A. Alam, D. Das, E. Hossain, and M. Hasan, “Diabetes Prediction Using Ensembling of Different Machine Learning Classifiers,” *IEEE Access*, vol. 8, pp. 76516–76531, 2020, doi: 10.1109/ACCESS.2020.2989857.
- [16]J. Han and M. Kamber, *Data mining: concepts and techniques*, 2nd ed. in The Morgan Kaufmann series in data management systems. Amsterdam; Boston: San Francisco, CA: Elsevier; Morgan Kaufmann, 2006.
- [17]S. Ulya, M. A. Soeleman, and F. Budiman, “Optimasi Parameter K Pada Algoritma K-NN Untuk Klasifikasi Prioritas Bantuan Pembangunan Desa,” *Techno.Com*, vol. 20, no. 1, pp. 83–96, Feb. 2021, doi: 10.33633/tc.v20i1.4215.
- [18]Indrayanti, Devi Sugianti, and M. Adib Al Karomi, *OPTIMASI PARAMETER K PADA ALGORITMA K-NEAREST NEIGHBOUR UNTUK KLASIFIKASI PENYAKIT DIABETES MELLITUS*, Cetakan pertama. Kudus: Badan Penerbit Universitas Muria Kudus, 2017.
- [19]I. Zyout and I. Abdel-Qader, “Classification of Microcalcification Clusters via PSOKNN Heuristic Parameter Selection and GLCM Features,” *Int. J. Comput. Appl.*, vol. 31, no. 2, pp. 34–39, Oct. 2011, doi: 10.5120/3798-5235.
- [20]K. W. Mahardika, Y. A. Sari, and A. Arwan, “Optimasi K-Nearest Neighbour Menggunakan Particle Swarm Optimization pada Sistem Pakar untuk Monitoring Pengendalian Hama pada Tanaman Jeruk”.
- [21]M. Pei, E. D. Goodman, and W. F. Punch, “Feature Extraction Using Genetic Algorithms”.
- [22]L. Zhang, K. Mistry, C. P. Lim, and S. C. Neoh, “Feature selection using firefly optimization for classification and regression models,” *Decis. Support Syst.*, vol. 106, pp. 64–85, Feb. 2018, doi: 10.1016/j.dss.2017.12.001.
- [23]M. A. Kahya, S. A. Altamir, and Z. Y. Algalal, “Improving firefly algorithm-based logistic regression for feature selection,” *J. Interdiscip. Math.*, vol. 22, no. 8, pp. 1577–1581, Nov. 2019, doi: 10.1080/09720502.2019.1706861.
- [24]X.-S. Yang, “Firefly algorithms for multimodal optimization,” in *Stochastic Algorithms: Foundations and Applications: 5th International Symposium, SAGA 2009, Sapporo, Japan, October 26-28, 2009. Proceedings 5*, Springer, 2009, pp. 169–178.
- [25]B. Wang, D.-X. Li, J.-P. Jiang, and Y.-H. Liao, “A modified firefly algorithm based on light intensity difference,” *J. Comb. Optim.*, vol. 31, pp. 1045–1060, 2016.
- [26]E. M. Mashhour, M. El Enas, K. T. Wassif, and A. I. Salah, “Feature selection approach based on firefly algorithm and chi-square,” *Int. J. Electr. Comput. Eng.*, vol. 8, no. 4, p. 2338, 2018.
- [27]X.-S. Yang, “Firefly algorithm, stochastic test functions and design optimisation,” *Int. J. Bio-Inspired Comput.*, vol. 2, no. 2, pp. 78–84, 2010.
- [28]D. A. Nggego, A. Setyanto, and Sukoco, “Implementation of the Firefly Algorithm in the Case of N-Queens Problem,” *J. RESTI Rekayasa Sist. Dan Teknol. Inf.*, vol. 4, no. 2, pp. 219–227, Apr. 2020, doi: 10.29207/resti.v4i2.1506.

- [29]H. Wisnu, M. Afif, and Y. Ruldevyani, “Sentiment analysis on customer satisfaction of digital payment in Indonesia: A comparative study using KNN and Naïve Bayes,” *J. Phys. Conf. Ser.*, vol. 1444, no. 1, p. 012034, Jan. 2020, doi: 10.1088/1742-6596/1444/1/012034.
- [30]G. Guo, H. Wang, D. Bell, Y. Bi, and K. Greer, “KNN model-based approach in classification,” in *On The Move to Meaningful Internet Systems 2003: CoopIS, DOA, and ODBASE: OTM Confederated International Conferences, CoopIS, DOA, and ODBASE 2003, Catania, Sicily, Italy, November 3-7, 2003. Proceedings*, Springer, 2003, pp. 986–996.