

Klasifikasi Dalam Mendeteksi Penyakit Hipoglikemia Dengan Menggunakan Metode *Random Forest* dan *Adaboost*

Ariansyah Arifin¹, Marshall Anugrah Najmi², Jovanka Samudra³, Sultan Farel Syah Reza⁴, Desta Sandya Prasvita⁵

Program Studi Informatika / Fakultas Ilmu Komputer
Universitas Pembangunan Nasional Veteran Jakarta

Jl. RS. Fatmawati, Pondok Labu, Jakarta Selatan, DKI Jakarta, 12460, Indonesia.

ariansyaha@upnvj.ac.id¹, marshall@upnvj.ac.id², jovankas@upnvj.ac.id³, sultanfarel@upnvj.ac.id⁴,
desta.sandya@upnvj.ac.id⁵

Abstrak. Hipoglikemia merupakan sebuah gangguan kesehatan yang penyebabnya adalah kadar glukosa pada tubuh seseorang dibawah angka kadar normal dan gangguan kesehatan ini merupakan sebuah gangguan lanjutan dari individu yang memiliki penyakit diabetes. Hipoglikemia sendiri terjadi ketika kadar glukosa pada darah dibawah angka 70 mg/dL dan kadar insulin yang tinggi melebihi angka 100 mg/dL. Penelitian ini melatih model probabilistik menggunakan algoritma *Machine Learning Random Forest* dan *Adaboost* dari glukosa darah dan kadar insulin pasien. Kedua algoritma tersebut termasuk kedalam bagian *ensemble learning* dengan algoritma *random forest* menggunakan teknik *bagging* dan algoritma *adaboost* menggunakan teknik *boosting*. Dengan demikian, penelitian ini akan mendapatkan algoritma terbaik dalam pengklasifikasian penyakit hipoglikemia. Hasil dari penelitian ini dengan sampel ukuran data *test* sebesar 20% dan *train* sebesar 80% didapat bahwa algoritma *random forest* mendapatkan akurasi 92% dan algoritma *adaboost* mendapatkan akurasi 93.75%. Dengan demikian algoritma *adaboost* lebih baik daripada algoritma *random forest* dalam melakukan pengklasifikasian penyakit hipoglikemia.

Kata Kunci. *Machine Learning*, Hipoglikemia, Algoritma, *Random Forest*, *Adaboost*

1 Pendahuluan

Hipoglikemia merupakan sebuah gangguan kesehatan yang terjadi apabila kadar gula pada tubuh seseorang dibawah angka kadar normal. Penyakit hipoglikemia juga merupakan komplikasi dari penyakit diabetes melitus karena penderita diabetes melitus akan menghadapi situasi yang mengharuskan individu mengkonsumsi obat yang bertujuan untuk mengontrol kadar glukosa pada darah agar tetap normal, namun dengan mengkonsumsi obat tersebut akan memberikan efek samping terapi yang dapat menyebabkan komplikasi hipoglikemia. Glukosa darah merupakan sumber energi bagi tubuh, namun disaat kadar glukosa pada tubuh terhitung rendah maka akan membuat tubuh kekurangan energi untuk melakukan aktivitas. Hipoglikemia terjadi ketika kadar glukosa pada darah dibawah angka 70 mg/dL dan kadar insulin yang tinggi melebihi angka 100 mg/dL. Kadar tingkat glukosa darah seseorang bisa saja tiba-tiba mengalami penurunan hingga terlalu rendah karena berbagai alasan, antara lain seperti aktivitas fisik yang berlebihan, gangguan dosis yang kuat untuk insulin atau kurangnya asupan makan.

Peningkatan penyakit diabetes melitus di berbagai negara serta pendeteksian yang lambat dari penyakit hipoglikemia yang berawal dari penyakit diabetes melitus ini mendorong untuk melakukan penelitian untuk pendeteksian penyakit hipoglikemia. Penelitian ini menggunakan *dataset* yang berasal dari Kaggle yang kemudian diolah menggunakan teknologi *machine learning*.

Machine learning merupakan cabang ilmu yang berasal dari kecerdasan buatan dimana merupakan teknologi untuk mengembangkan mesin agar dapat belajar dengan sendirinya tanpa membutuhkan arahan dari penggunanya. *Machine learning* dikembangkan dengan menggabungkan beberapa keilmuan seperti matematika, statistika serta *data mining* sehingga mesin dapat menganalisa suatu data tanpa harus di program secara spesifik.

Dalam perkembangannya, *Machine learning* dijalankan dengan menggunakan 3 metode, seperti *Supervised Learning*, *Unsupervised Learning* dan *Reinforcement Learning*. Klasifikasi merupakan salah satu pendekatan dari metode *Supervised Learning* yang dimana pada metode ini pemberian kelas pada *dataset* dibutuhkan untuk melihat tingkat akurasi kinerjanya serta mempelajari apa hubungan dan ketergantungan antar data. Cara kerja metode ini adalah memasukkan informasi berupa atribut dan kelas sebagai masukan untuk komputer belajar

sehingga akan dapat memprediksi dengan memberikan hasil atau *output*.

Dari hasil studi literatur yang kami lakukan, didapat beberapa penelitian lain yang telah dilakukan, seperti memprediksi hipoglikemia postprandial dengan menggunakan metode *random forest*, *support vector machine*, *KNN*, dan *logistic regression*. Hasil dari penelitian tersebut didapatkan bahwa metode *random forest* merupakan metode yang lebih baik untuk dikembangkan selanjutnya [9]. Pada penelitian lainnya, terdapat penelitian terkait terjadinya hipoglikemia pada pasien dengan diabetes tipe 2 menggunakan metode *SVM*, *random forest*, *KNN* dan *naive bayes* [10]. Kemudian ada juga penelitian tentang hipoglikemia postprandial neonatus yang menggunakan metode *decision tree*, *SVM*, dan *naive bayes* untuk dilakukan perbandingan. Kinerja dari ketiga metode tersebut dibandingkan melalui nilai akurasi, *recall*, *precision* dan *F-Measure*. Hasil dari penelitian tersebut menunjukkan bahwa dengan menggunakan metode *naive bayes* akan lebih unggul dengan nilai akurasi tertinggi, yaitu sebesar 86,40% [4]. Pada penelitian terakhir dijelaskan bahwa mereka menggunakan model *random forest* yang digunakan untuk memprediksi risiko hipoglikemia pada pasien diabetes tipe 1. Pada penelitian tersebut didapatkan nilai sensitivitas sebesar 94% dan 91% untuk 30 dan 60 menit dan nilai spesifisitas sebesar 93% selama 30 menit dan 90% untuk cakrawala prediksi selama 60 menit [2].

Terdapat banyak algoritma yang digunakan untuk melakukan klasifikasi, salah satunya ialah algoritma *random forest* dan *adaboost*. Algoritma *random forest* adalah algoritma yang berasal dari pengembangan algoritma *Classification and Regression Tree* (CART). Algoritma *random forest* sendiri memanfaatkan metode *bootstrap aggregating* dan *random forest selection*. Dengan menggunakan metode tersebut, *random forest* memiliki kelebihan, yaitu dapat menghasilkan nilai tingkat keakurasian yang terhitung tinggi, dapat menyelesaikan permasalahan dengan jumlah data yang besar, serta tidak dilakukannya pemangkasan variabel [8].

Selain *Random Forest Classifier*, terdapat juga algoritma *AdaBoost* sebagai metode klasifikasi. Algoritma *adaboost* pertama kali diperkenalkan oleh Yoav Freund dan Robert Schapire pada tahun 1995. Algoritma *adaboost* merupakan teknik ansambel yang menggunakan *loss function* fungsi *exponensial* yang berfungsi meningkatkan akurasi klasifikasi yang dibuat [13].

Pada penelitian kali ini kami ingin mengklasifikasikan seseorang yang terkena penyakit Hipoglikemia. Kami ingin membuat sebuah model dan melakukan pengecekan tingkat akurasi seseorang terkena penyakit Hipoglikemia menggunakan algoritma *random forest* dan *adaboost*.

2 Landasan Teori

2.1 Machine Learning

Machine learning merupakan sebuah teknik untuk melakukan inferensi terhadap sebuah data dengan pendekatan model matematis yang mereferensikan pola-pola data [6]. *Machine learning* ini memproses dan memprediksi data dengan menggunakan algoritma pembelajaran untuk menampilkan data. *Machine learning* merupakan sebuah cabang dari ilmu kecerdasan buatan yang termasuk sebuah teknologi untuk mengembangkan mesin agar dapat melakukan pembelajaran dengan mandiri tanpa membutuhkan arahan lagi. *Machine learning* dikembangkan menggunakan beberapa keilmuan seperti matematika, statistika serta *data mining* sehingga mesin dapat menganalisa suatu data tanpa harus di program secara spesifik.

2.2 Python

Python ditemukan oleh Guido van Rossum pada tahun 1990 di Belanda dan dinamai menurut program televisi favorit Guido, *Monty Python's Flying Circus*. Van Rossum awalnya membuat Python hanya sebagai hobi, lalu setelahnya Python menjadi bahasa pemrograman yang banyak dipergunakan di industri serta pendidikan karena sintaksnya yang tidak terlalu rumit, ringkas, intuitif serta *library* yang sangat banyak [12].

Python adalah bahasa pemrograman yang berorientasikan objek yang bergerak maju mengikuti perkembangan dan bisa dipergunakan untuk berbagai pengembangan aplikasi. Python memberikan dukungan untuk dapat mengintegrasikan dengan bahasa dan alat pemrograman lainnya. Python diakui menjadi bahasa pemrograman yang menggabungkan kapabilitas, kemampuan, dengan sintaks kode yang sangat jelas dan dilengkapi dengan fungsi-fungsi yang sangat besar dan lengkap.

2.3 Klasifikasi Penyakit Hipoglikemia

Klasifikasi penyakit hipoglikemia merupakan teknik untuk mengamati perilaku berdasarkan atribut *dataset* penyakit hipoglikemia yang sudah diklasifikasikan. Tujuan dari teknik klasifikasi adalah untuk melakukan pengklasifikasian terhadap data sehingga membentuk sebuah pola. Tahap klasifikasi dibagi menjadi dua tahapan, yaitu tahap pembelajaran serta tahap pengujian [11]. Beberapa metode klasifikasi yang digunakan antara lain *Random Forest* dan *AdaBoost*.

2.4 Prediksi

Prediksi merupakan suatu praktik untuk memperkirakan sesuatu yang kemungkinan besar akan terjadi berdasarkan data sebelumnya dan saat ini untuk mengurangi tingkat kesalahan dalam melakukan prediksi [1]. Prediksi tidak harus memberikan respons yang komprehensif tentang apa yang akan terjadi, tetapi lebih merupakan pendekatan yang mendekati.

2.5 Confusion Matrix

Confusion matrix adalah metode yang digunakan untuk menyampaikan informasi yang akan terjadi dari proses klasifikasi secara mandiri oleh mesin yang bermanfaat untuk proses analisis dari seberapa baik *classifier* memprediksi tupel dari kelas yang tidak selaras [7]. Contoh dari *confusion matrix* yang terdiri dari hanya dua kelas maka akan dibedakan menjadi 2 bagian, yaitu kelas positif dan kelas negatif. Klasifikasi *True* positif adalah pengklasifikasian yang tepat kelas positif, sedangkan klasifikasi *true* negatif adalah pengklasifikasian yang tepat untuk kelas negatif kelas negatif. Untuk klasifikasi *false* positif adalah salahnya pengklasifikasian untuk kelas yang negatif dengan diprediksi dengan nilai kelas positif. Selanjutnya, klasifikasi *false* negatif adalah salahnya prediksi kelas positif dengan diprediksi dengan nilai kelas negatif, hal ini dapat dilihat pada tabel 1.

Tabel 1. Confusion Matrix

Confusion Matrix		
Klasifikasi benar	Klasifikasi	
	Positif (+)	Negatif (-)
Positif benar	<i>True</i> Positif	<i>False</i> Positif
Negatif benar	<i>False</i> Negatif	<i>True</i> Negatif

Keterangan:

1. *True* Positif (TP), Jumlah pengklasifikasian data positif yang diklasifikasikan nilai positif.
2. *False* Positif (FP), Jumlah pengklasifikasian data negatif yang diklasifikasikan nilai positif.
3. *False* Negatif (FN), Jumlah pengklasifikasian data positif yang diklasifikasikan nilai negatif.
4. *True* Negatif (TN), Jumlah pengklasifikasian data negatif yang diklasifikasikan nilai negatif.

Dari *confusion matrix* kita mendapatkan persamaan untuk menghitung nilai akurasi, presisi dan *recall* yang dapat lihat dibawah ini:

$$\text{Akurasi} : \frac{Tp + Tn}{Tp + Tn + Fp + Fn} \times 100\% \quad (1)$$

$$\text{Presisi} : \frac{Tp + Tn}{Tp + Fp} \times 100\% \quad (2)$$

$$\text{Recall} : \frac{Tp}{Tp + Fn} \times 100\% \quad (3)$$

2.6 Random Forest

Random forest adalah klasifikasi yang berbasis pohon dan pendekatan pohon regresi dengan menggunakan mekanisme pemisahan biner rekursif untuk mencapai *node* akhir. Algoritma *random forest* memiliki sejumlah manfaat, termasuk kemampuan untuk membuat kesalahan yang relatif rendah, kinerja klasifikasi yang kuat, penanganan yang cepat dari sejumlah besar data pelatihan, dan metode yang efektif untuk menebak data yang

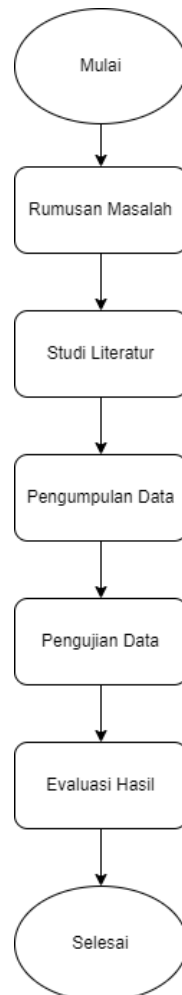
hilang. *Random forest* membangun beberapa pohon yang independen dengan subset yang dipilih secara acak melalui proses *bootstrap* [3].

2.7 AdaBoost Classifier

AdaBoost adalah salah satu algoritma *supervised* yang diterapkan untuk membangun model klasifikasi. Algoritma *adaboost* merupakan teknik ansambel yang menggunakan *loss function* fungsi *exponensial* yang berfungsi meningkatkan akurasi klasifikasi yang dibuat. *Adaboost* pertama kali dipertunjukkan oleh Yoav Freund dan Robert Schapire pada tahun 1995. Mulanya algoritma *adaboost* hanya digunakan pada model regresi, namun dalam perkembangan teknologi, algoritma ini juga diterapkan dalam model statistik lainnya [13]. Pengklasifikasi *adaboost* dapat memberikan informasi tentang fitur relevansinya dengan klasifikasi. Keuntungan lain dari *adaboost classifier* yaitu cepat, sederhana, sangat mudah diterapkan, dan tidak memerlukan parameter kecuali untuk parameter untuk nomor iterasi [5].

3 Metodologi Penelitian

Tahapan dalam melakukan penelitian ini dibagi menjadi 5 tahapan yang terdiri dari perumusan masalah, studi literatur, pengumpulan data, pengujian data, serta evaluasi hasil. Untuk jelasnya dapat dilihat pada ilustrasi gambar 1.



Gambar. 1. *Flowchart* Metode Penelitian

3.1 Rumusan Masalah

Rumusan masalah merupakan kegiatan identifikasi dan merumuskan permasalahan yang sedang terjadi di tempat penelitian. Pada tahap ini didapatkan sebuah ruang lingkup masalah.

3.2 Studi Literatur

Studi literatur digunakan dalam penelitian ini bertujuan untuk memahami permasalahan yang ada pada penyakit hipoglikemia. Selain mencari tentang permasalahan tentang penyakitnya, kita juga mencari dan mengumpulkan jurnal tentang pengklasifikasian penyakit hipoglikemia. Dari hasil literatur yang sudah dilakukan, tim kami mengambil metode *Random Forest* dan *AdaBoost*.

3.3 Pengumpulan Data

Proses pengumpulan data yang digunakan yaitu dengan mencari pada website yang menyediakan berupa data-data. Data hipoglikemia yang digunakan pada proyek ini didapatkan dari *website kaggle.com* (<https://www.kaggle.com/datasets/uciml/pima-indians-diabetes-database>). Banyaknya baris pada data hipoglikemia yang digunakan ini sebanyak 2000 data dengan memiliki atribut dengan jumlah sebanyak 8 jenis.

3.4 Pengujian Data

Pengujian pengklasifikasian yang digunakan yaitu menggunakan 2 metode, yaitu metode *random forest* dan *adaboost*. Pengujian menggunakan data yang dibagi menjadi 2 bagian yang terdiri dari data *testing* dan juga data *training* dengan persentase bagi sebesar 20% data *testing* dan 80% data *training*. Hasil dari pengujian data menggunakan algoritma *random forest* dan *adaboost* menghasilkan sebuah nilai akurasi serta hasil laporan klasifikasi.

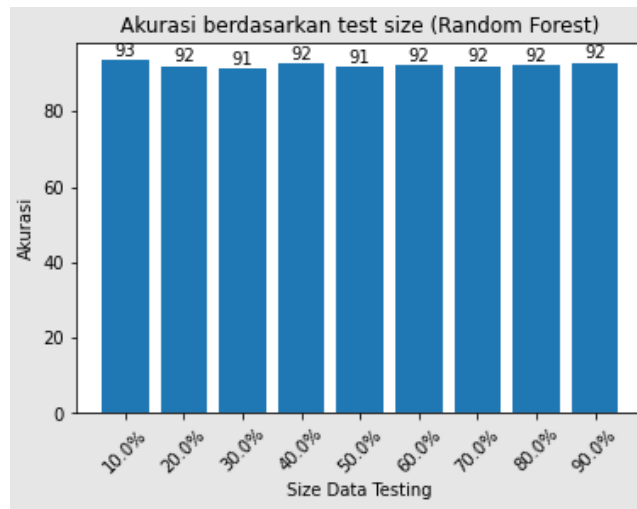
3.5 Evaluasi Hasil

Pada tahap ini akan dilakukan evaluasi terhadap penerapan algoritma *random forest* dengan algoritma *adaboost* diterapkan selama fase pengujian menggunakan parameter uji yang telah ditentukan yaitu kadar glukosa dalam darah dan insulin. Hasil evaluasi akan menunjukkan tingkat akurasi dari kedua algoritma serta nilai *Precision* dan *Recall* yang didapatkan dari *Confusion Matrix*.

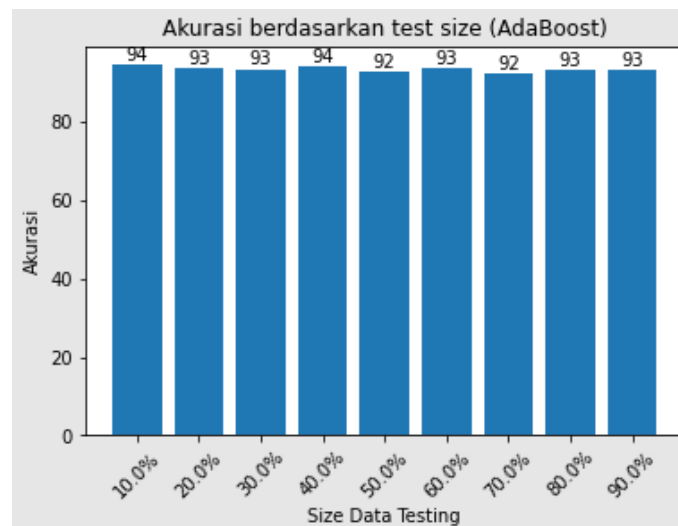
4 Hasil dan Pembahasan

4.1 Penerapan Metode

Berdasarkan *dataset* hipoglikemia yang dikumpulkan dari *website Kaggle.com*, didapat 8 atribut dan total *dataset* sebanyak 2000. Data yang di *download* dari website tersebut memiliki format *csv*. *Dataset* yang sudah dikumpulkan tersebut selanjutnya dibagi menjadi 2 bagian, yaitu data *training* dan data *testing* dengan persentase 80:20 yang dilakukan secara acak. Berikut ini merupakan plot model dari hasil algoritma *random forest* yang dapat dilihat pada gambar 2 dan algoritma *adaboost* yang ada pada ilustrasi gambar 3 dengan pembagian data dari 10% sampai 90%.

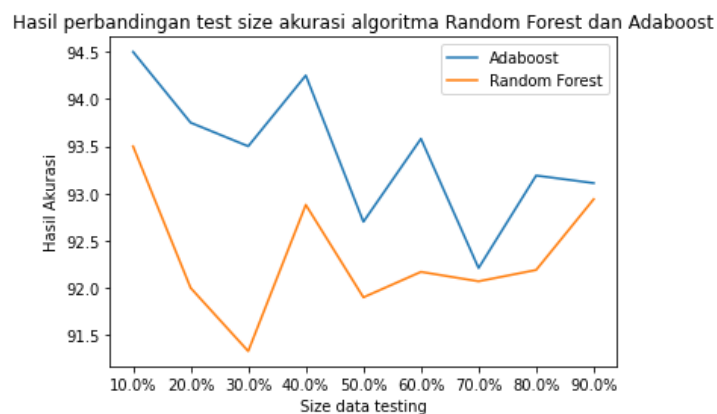


Gambar 2. Hasil Plot menggunakan *random forest*



Gambar 3. Hasil plot menggunakan *AdaBoost*

Jika dibandingkan dari hasil kedua algoritma tersebut terlihat bahwa algoritma *adaboost* lebih unggul daripada algoritma *random forest*, untuk lebih jelas dalam membandingkan kedua hasil algoritma tersebut dibuatlah sebuah *line chart* sebagai visualisasi yang dapat dilihat pada gambar 4.



Gambar. 4. Hasil perbandingan akurasi algoritma *AdaBoost* dan *Random Forest*

4.2 Hasil Klasifikasi

Berdasarkan klasifikasi yang telah dilakukan menggunakan *random forest* dan *adaboost* dengan sampel *test size* sebesar 20% didapatkan data plot di atas serta hasil akurasi, presisi dan *recall* untuk tiap masing-masing algoritma yang dapat dilihat pada tabel 2, 3, 4, 5.

Tabel 2. Hasil akurasi menggunakan *Random Forest*

<i>Accuracy</i>	<i>Precision</i>	<i>Recall</i>
92%	<i>outcome</i> = 0 (89%)	<i>outcome</i> = 0 (99%)
	<i>outcome</i> = 1 (98%)	<i>outcome</i> = 1 (82%)

Tabel 3. Hasil *confusion matrix* metode *Random Forest*

Confusion Matrix		
Klasifikasi benar	Klasifikasi	
	Positif (+)	Negatif (-)
Positif benar	232	3
Negatif benar	29	136

Tabel 4. Hasil akurasi menggunakan *AdaBoost*

<i>Accuracy</i>	<i>Precision</i>	<i>Recall</i>
93.75%	<i>outcome</i> = 0 (94%)	<i>outcome</i> = 0 (95%)
	<i>outcome</i> = 1 (93%)	<i>outcome</i> = 1 (92%)

Tabel 5. Hasil *confusion matrix* metode *AdaBoost*

Confusion Matrix		
Klasifikasi benar	Klasifikasi	
	Positif (+)	Negatif (-)
Positif benar	223	12
Negatif benar	13	52

5 Kesimpulan

Berdasarkan hasil pengklasifikasian dengan menggunakan algoritma *Random Forest* dan *Adaboost* terhadap penyakit Hipoglikemia, didapat kesimpulan bahwa dengan menggunakan algoritma *random forest* masih terdapat suatu pengklasifikasian yang tidak tepat dengan tingkat akurasi 92% sedangkan dengan menggunakan algoritma *adaboost* didapatkan akurasi yang lebih unggul sebesar 93.75%. Oleh karena itu, algoritma *adaboost* merupakan algoritma yang lebih unggul dibandingkan *Random Forest*.

Referensi

- [1] Carlis Hutabarat (2018) 'Penerapan Data Mining Untuk Memprediksi Permintaan Produk Kartu Perdana Internet Menggunakan Algoritma C5.0 (Studi Kasus: Vidha Ponsel)', *Pelita Informatika*, 6(April), pp. 419–424.
- [2] Dave, D. et al. (2020) 'Feature-Based Machine Learning Model for Real-Time Hypoglycemia Prediction', *Journal of Diabetes Science and Technology*, 15(4), pp. 842–855. doi:10.1177/1932296820922622.
- [3] Devella, S., Yohannes, Y. and Rahmawati, F.N. (2020) 'Implementasi Random Forest Untuk Klasifikasi Motif Songket Palembang Berdasarkan SIFT', *JATISI (Jurnal Teknik Informatika dan Sistem Informasi)*, 7(2), pp. 310–320. doi:10.35957/jatisi.v7i2.289.
- [4] Mutua, E. and Nyakango, L. (2020) 'Comparative Analysis of Machine Learning Classification Techniques for Neonatal Postprandial Hypoglycemia Symptoms Screening .', (October), pp. 1–9.
- [5] Prasvita, D.S. (2018) 'Post-processing and band selection for hyperspectral image data classification with AdaBoost.MH', *Proceedings - 2017 International Conference on Sustainable Information Engineering and*

Technology, SIET 2017, 2018-Janua, pp. 6–13. doi:10.1109/SIET.2017.8304100.

- [6] Putra, J.W.G. (2019) ‘Pengenalan konsep pembelajaran mesin dan deep learning’, *Computational Linguistics and Natural Language Processing Laboratory*, 4, pp. 1–235. Available at: <https://www.researchgate.net/publication/323700644>.
- [7] Rahman, M.F. *et al.* (2017) ‘Klasifikasi Untuk Diagnosa Diabetes Menggunakan Metode Bayesian Regularization Neural Network (RBNN)’, *Jurnal Informatika*, 11(1), p. 36. doi:10.26555/jifo.v11i1.a5452.
- [8] Ramadhan, A., Susetyo, B. and Indahwati, I. (2019) ‘Penerapan Metode Klasifikasi Random Forest Dalam Mengidentifikasi Faktor Penting Penilaian Mutu Pendidikan’, *Jurnal Pendidikan dan Kebudayaan*, 4(2), p. 169. doi:10.24832/jpnk.v4i2.1327.
- [9] Seo, W. *et al.* (2019) ‘A machine-learning approach to predict postprandial hypoglycemia’, *BMC Medical Informatics and Decision Making*, 19(1), pp. 1–13. doi:10.1186/s12911-019-0943-4.
- [10] Sudharsan, B., Peeples, M. and Shomali, M. (2015) ‘Hypoglycemia prediction using machine learning models for patients with type 2 diabetes’, *Journal of Diabetes Science and Technology*, 9(1), pp. 86–90. doi:10.1177/1932296814554260.
- [11] Syahdan, Madinatul Munawwarah Ridwan, Ismaya, Andi Muhammad Aminullah, E. (2021) ‘ANALISIS PENERAPAN SISTEM KLASIFIKASI DDC DALAM PENGOLAHAN PUSTAKA’, 2(March), pp. 63–80.
- [12] Yoshua, I.A., Pragantha, J. and Sutrisno, T. (2020) ‘Aplikasi Pengukur Kelayakan Tempat Tinggal Menggunakan Metode Naive Bayes’, *Jurnal Ilmu Komputer dan Sistem Informasi*, 8(1), p. 79. doi:10.24912/jiksi.v8i1.11473.
- [13] Zulhanif (2015) ‘Algoritma AdaBoost Dalam Pengklasifikasian’, *Prosiding Seminar Nasional Matematika Dan Pendidikan Matematika UMS*, pp. 169–198. doi:10.1017/cbo9781139028462.008.