

Comparative Analysis of Machine Learning Algorithms for Knowledge Discovery in Poverty Level Classification of Indonesian Provinces

Novita Ramadhini¹, Ken Ditha Tania^{2*}, Putri Eka Sevtiyuni³

^{1,2,3} Sistem Informasi, Universitas Sriwijaya

novitaramadhini0@gmail.com¹, kenya.tania@gmail.com², putrieka@unsri.ac.id³

Article Info

Article history:

Received 2026-08-04

Revised 2026-09-25

Accepted 2026-10-17

Keyword:

*Classification,
Data Mining,
Knowledge Discovery,
Machine Learning,
Poverty Data.*

ABSTRACT

Poverty in Indonesia remains a major development challenge that requires an understanding of socio-economic conditions across regions. Previous studies have applied machine learning for poverty classification. However, most have compared a limited number of algorithms and focused on district or municipality-level data. This study aims to compare the performance of four machine learning algorithms, namely Naïve Bayes, Support Vector Machine (SVM), Decision Tree, and Random Forest, in classifying the poverty levels of Indonesian provinces into “Low” and “High” categories. The study employs the CRISP-DM methodology and Synthetic Minority Over-sampling Technique (SMOTE) to address class imbalance. The dataset was obtained from Statistics Indonesia (BPS) for the 2018-2025 period and consists of five predictor variables and one label variable. The models were developed using RapidMiner with 10-Fold Cross Validation and evaluated using accuracy, precision, recall, F1-score, and balanced accuracy. The results show that Random Forest achieved the best performance, with an accuracy of 81.79%, an F1-score macro average of 80.46%, and a balanced accuracy of 81.11%. Furthermore, feature importance analysis of the Random Forest model was used to identify the relative importance of the indicators in distinguishing the poverty categories. The findings provide knowledge regarding the contribution of the indicators to the classification of provincial poverty levels in Indonesia.



This is an open access article under the [CC-BY-SA](https://creativecommons.org/licenses/by-sa/4.0/) license.

I. PENDAHULUAN

Kemiskinan hingga saat ini masih menjadi salah satu permasalahan sosial dan ekonomi di berbagai negara, termasuk Indonesia [1]. Tingkat kemiskinan pada setiap provinsi di Indonesia bervariasi seiring dengan perbedaan jumlah penduduk dan karakteristik wilayahnya yang beragam [2]. Beberapa faktor yang berkaitan dengan kondisi tersebut meliputi ketimpangan ekonomi, keterbatasan akses terhadap pendidikan dan pekerjaan, serta rendahnya kualitas hidup masyarakat yang tercermin dari kesulitan memenuhi kebutuhan dasar, layanan kesehatan, pendidikan, dan peluang ekonomi [3]. Kondisi yang beragam tersebut menunjukkan karakteristik kemiskinan yang berbeda-beda di setiap wilayah, sehingga diperlukan upaya penanggulangan yang mempertimbangkan kondisi dan karakteristik masing-

masing daerah [4]. Oleh karena itu, penanggulangan kemiskinan memerlukan keterlibatan berbagai pihak sebagai salah satu prioritas pembangunan.

Dalam mendukung upaya tersebut, pemerintah memerlukan informasi yang akurat mengenai kondisi kemiskinan di setiap provinsi sebagai dasar dalam menyusun kebijakan penanggulangan kemiskinan [5]. Informasi tersebut dapat dimanfaatkan sebagai dasar untuk mengidentifikasi provinsi berdasarkan tingkat kemiskinan yang relatif tinggi sehingga penyusunan program dan kebijakan penanggulangan kemiskinan dapat dilakukan secara lebih tepat sasaran [6]. Untuk memperoleh informasi tersebut, diperlukan pengukuran tingkat kemiskinan yang tepat agar jumlah penduduk miskin serta kondisi kemiskinan di setiap wilayah dapat diketahui secara akurat [7].

Di Indonesia, pengukuran tingkat kemiskinan dilakukan oleh Badan Pusat Statistik (BPS) menggunakan pendekatan kemampuan memenuhi kebutuhan dasar (*basic needs approach*). Berdasarkan pendekatan tersebut, tingkat kemiskinan dapat diukur menggunakan tiga indikator utama, yaitu *Head Count Index* (P0), Indeks Kedalaman Kemiskinan (P1), dan Indeks Keparahan Kemiskinan (P2) [7]. Ketiga indikator tersebut dikenal sebagai *Foster-Greer-Thorbecke Index* (FGT Index) yang dikembangkan oleh Foster, Greer, dan Thorbecke [8].

Untuk menghasilkan data yang lebih relevan, data kemiskinan yang dikumpulkan dari berbagai indikator harus diolah dan dianalisis. *Knowledge discovery* adalah salah satu pendekatan untuk mendapatkan informasi dari kumpulan data tersebut dengan mencari pengetahuan yang tersembunyi melalui analisis data [9]. Dalam konteks penelitian ini, proses *knowledge discovery* dapat membantu mengidentifikasi pola berdasarkan indikator kemiskinan yang digunakan dalam proses klasifikasi tingkat kemiskinan suatu wilayah. Informasi yang diperoleh dari hasil pengukuran kemiskinan dapat dimanfaatkan untuk mendukung proses analisis dan pengambilan keputusan dalam upaya penanggulangan kemiskinan di Indonesia. Untuk memperoleh pola tersebut, penelitian ini menerapkan teknik klasifikasi yang telah dipilih sebagai metode *data mining* berdasarkan karakteristik yang dimiliki untuk mengelompokkan data ke dalam kategori [1], [5].

Berbagai penelitian telah memanfaatkan teknik klasifikasi untuk mendukung analisis kemiskinan melalui perbandingan performa beberapa algoritma *machine learning*. Salah satunya membandingkan algoritma *Naïve Bayes* dan *Support Vector Machine* dalam mengklasifikasikan tingkat kemiskinan kabupaten dan kota di Indonesia dengan menerapkan metode *Random Oversampling* untuk mengatasi proporsi kelas yang tidak seimbang [10]. Berdasarkan hasil evaluasi, SVM memperoleh akurasi sebesar 81%, *precision* dan *recall* yang relatif baik, serta kinerja klasifikasi yang lebih tinggi dibandingkan *Naïve Bayes* yang memperoleh akurasi sebesar 76%.

Penelitian [2] menunjukkan *Random Forest* memiliki kinerja yang lebih efektif daripada LMT, yaitu tingkat akurasi 100% pada iterasi $k=10$ dan $k=500$, dan 75% pada iterasi $k=100$. Dengan demikian, *Random Forest* terbukti lebih efektif dibandingkan LMT untuk menganalisis data kemiskinan di Indonesia.

Penelitian mengenai klasifikasi tingkat kemiskinan menggunakan metode *machine learning* juga telah dilakukan oleh [10] dengan memanfaatkan data *Somalia Demographic and Health Survey* (SDHS) 2020. Melalui pendekatan tersebut, penelitian ini mengidentifikasi berbagai faktor yang memengaruhi tingkat kemiskinan di Somalia. Penelitian tersebut menggunakan 13 variabel, yaitu wilayah administratif, ukuran rumah tangga, kelompok usia responden, status pekerjaan suami, usia kepala rumah tangga, tempat tinggal, tingkat pendidikan suami, paparan media

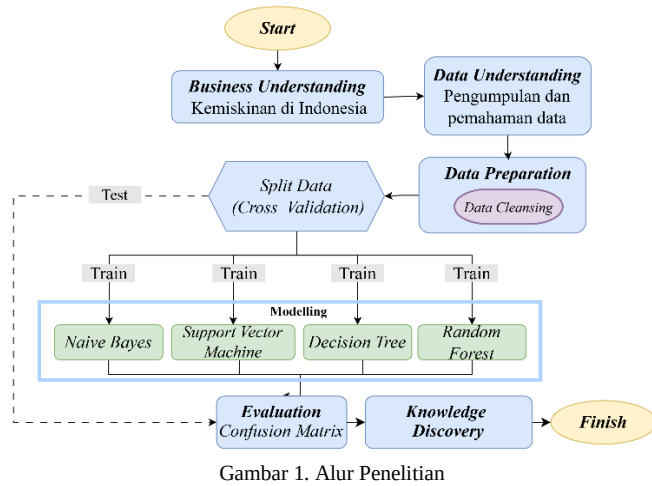
massa, jenis kelamin kepala rumah tangga, sumber air untuk diminum, tingkat pendidikan ibu, jenis fasilitas toilet, dan status pekerjaan ibu. Hasil menunjukkan bahwa pada penelitian tersebut, *Random Forest* memperoleh tingkat akurasi tertinggi, yaitu 96,38%, diikuti oleh *Logistic Regression* sebesar 74,95%, *Decision Tree* 73,73%, dan SVM 67,21%. *Random Forest* juga menunjukkan nilai *recall* dan *sensitivity* tertinggi, masing-masing sebesar 95,90%, yang menunjukkan bahwa algoritma ini mampu mengidentifikasi sebagian besar individu yang tergolong miskin secara tepat. Sebagai perbandingan, *Logistic Regression* memperoleh nilai *recall* sebesar 70,49%, *Decision Tree* dan SVM masing-masing memperoleh nilai 66,44% dan 62,49%. Dari sisi *specificity*, *Random Forest* kembali menunjukkan performa terbaik dengan nilai 96,80%, diikuti oleh *Decision Tree* 86,17%, *Logistic Regression* 80,09%, dan SVM 73,45%.

Berdasarkan berbagai penelitian sebelumnya, klasifikasi tingkat kemiskinan telah banyak dilakukan dengan menerapkan metode *machine learning* untuk mendukung analisis dan pengambilan keputusan dalam penanggulangan kemiskinan. Namun, penelitian-penelitian tersebut umumnya hanya membandingkan beberapa algoritma *machine learning*, seperti *Random Forest* dan LMT [2]. Selain itu, penelitian terdahulu lebih banyak menggunakan variabel yang berasal dari karakteristik rumah tangga atau individu, sehingga pemanfaatan indikator makro tingkat provinsi dalam proses klasifikasi tingkat kemiskinan masih relatif terbatas. Melalui pendekatan *knowledge discovery*, hasil analisis berbagai indikator dapat menghasilkan informasi mengenai keterkaitan dan kontribusi relatif indikator terhadap proses klasifikasi tingkat kemiskinan di wilayah tertentu. Proses tersebut dapat membantu menggambarkan karakteristik observasi berdasarkan kategori tingkat kemiskinan serta memberikan informasi mengenai indikator yang memiliki kontribusi relatif dalam membedakan kedua kategori tersebut.

Berdasarkan celah yang ditemukan pada penelitian sebelumnya, penelitian ini akan membandingkan kinerja empat algoritma *machine learning*, yaitu *Naïve Bayes*, *Support Vector Machine*, *Decision Tree*, dan *Random Forest* dalam mengklasifikasikan tingkat kemiskinan provinsi di Indonesia ke dalam kategori “Rendah” dan “Tinggi”, serta menganalisis *feature importance* pada model dengan kinerja terbaik. Penelitian ini menggunakan lima indikator sebagai variabel prediktor dan satu indikator yang digunakan sebagai dasar pembentukan label “Rendah” dan “Tinggi”. Selain itu, untuk menangani ketidakseimbangan kelas pada data, penelitian ini menggunakan teknik *Synthetic Minority Over-sampling Technique* (SMOTE). Hasil penelitian diharapkan dapat memberikan gambaran mengenai perbandingan kinerja algoritma dalam mengklasifikasikan tingkat kemiskinan provinsi di Indonesia.

II. METODE

Penelitian ini menggunakan metodologi *Cross-Industry Standard Process for Data Mining* (CRISP-DM) sebagai acuan penyusunan tahapan penelitian agar lebih terstruktur. Metodologi tersebut terdiri atas enam tahapan yang saling berkaitan, yaitu *Business Understanding*, *Data Understanding*, *Data Preparation*, *Modelling*, *Evaluation*, dan *Deployment* [11]. Dalam penelitian ini, proses penelitian dilakukan hingga tahap *Evaluation*, sedangkan tahapan *Deployment* tidak dilaksanakan karena penelitian difokuskan pada proses klasifikasi dan evaluasi model. Selanjutnya, hasil evaluasi dianalisis melalui tahapan *knowledge discovery* untuk memperoleh informasi dan pola yang dapat dimanfaatkan sebagai dasar pendukung pengambilan keputusan. Penelitian [12] juga menerapkan metodologi CRISP-DM dalam proses klasifikasi program bantuan sosial menggunakan algoritma *K-Nearest Neighbor* (K-NN). Gambar 1 menunjukkan alur yang akan dilakukan pada penelitian ini.



Gambar 1. Alur Penelitian

A. Business Understanding

Tahap ini diawali dengan mengidentifikasi permasalahan, memahami tujuan penelitian, serta menganalisis kondisi yang berkaitan dengan data kemiskinan provinsi di Indonesia. Hasil identifikasi tersebut kemudian digunakan sebagai dasar dalam menentukan solusi yang sesuai untuk proses klasifikasi [11].

B. Data Understanding

Tahap ini diawali dengan mengumpulkan data kemiskinan provinsi di Indonesia, kemudian dilanjutkan dengan identifikasi karakteristik dan pemahaman terhadap setiap variabel yang digunakan. Tahap ini dilakukan untuk memahami kondisi dan karakteristik data secara lebih mendalam sebagai dasar sebelum data memasuki tahap pengolahan. Terdapat lima variabel prediktor dan satu indikator yang digunakan sebagai dasar pembentukan label

pada penelitian ini. Rumus perhitungan oleh BPS dapat dilihat berikut ini:

$$RLS = \frac{1}{n} \times \sum_{i=1}^n x_i \quad (1)$$

Rata-rata Lama Sekolah (RLS) menggambarkan capaian pendidikan formal penduduk berusia 15 tahun ke atas. Variabel ini diperoleh dari rata-rata lama pendidikan yang telah ditempuh oleh seluruh penduduk pada kelompok usia tersebut.

$$IPM = \sqrt[3]{I_{\text{kesehatan}} \times I_{\text{pendidikan}} \times I_{\text{pengeluaran}}} \times 100 \quad (2)$$

IPM digunakan untuk menggambarkan kualitas pembangunan manusia berdasarkan tiga dimensi utama, yaitu kesehatan, pendidikan, dan standar hidup layak.

Persentase Akses Sanitasi Layak =

$$\frac{\text{Jumlah Rumah Tangga dengan Akses Sanitasi Layak}}{\text{Total Rumah Tangga}} \times 100\% \quad (3)$$

Persentase rumah tangga yang memiliki akses terhadap sanitasi layak menunjukkan jumlah rumah tangga yang menggunakan fasilitas sanitasi sesuai standar kelayakan, baik secara individu maupun secara bersama dengan rumah tangga lainnya.

$$P = \frac{R_{\text{layak}}}{R_{\text{total}}} \times 100\% \quad (4)$$

Persentase diatas diperoleh dengan membandingkan jumlah rumah tangga yang memiliki akses air minum layak dengan jumlah total rumah tangga, kemudian dikalikan 100%. Nilai yang dihasilkan menunjukkan persentase rumah tangga yang memiliki akses air minum layak dalam suatu wilayah.

$$P_1 = \frac{1}{N} \sum_{i=1}^q \left(\frac{z - y_i}{z} \right) \quad (5)$$

Indeks Kedalaman Kemiskinan (P1) digunakan untuk menggambarkan rata-rata selisih pengeluaran penduduk miskin terhadap garis kemiskinan. Indikator ini menunjukkan tingkat kedalaman kemiskinan yang dialami penduduk miskin, sehingga semakin besar nilai P1, semakin besar pula rata-rata kekurangan pengeluaran penduduk miskin dibandingkan dengan garis kemiskinan [13].

C. Data Preparation

Tahap selanjutnya merupakan proses penyiapan data yang bertujuan menghasilkan dataset dengan kondisi yang sesuai sebelum digunakan dalam pemodelan. Proses penyiapan data dilakukan menggunakan Google Colab, yang

meliputi pembersihan data dengan menangani data yang memiliki nilai kosong (*missing value*), memeriksa adanya duplikasi data, serta menyaring data yang tidak valid sehingga dataset yang digunakan menjadi lebih lengkap dan konsisten [14]. Dalam penelitian ini, variabel numerik tidak melalui proses normalisasi maupun standarisasi. Keputusan tersebut didasarkan pada karakteristik algoritma *Decision Tree*, *Random Forest*, dan *Naïve Bayes* yang tidak menjadikan skala fitur sebagai ketergantungan utama dalam pembentukan model. Sementara itu, SVM memiliki sensitivitas yang lebih tinggi terhadap perbedaan skala antarfitur. Meskipun demikian, skala asli data tetap dipertahankan secara konsisten pada seluruh model untuk menjaga keseragaman tahapan *preprocessing*, sehingga kondisi tersebut menjadi salah satu batasan khusus pada penggunaan SVM. Setelah pembersihan data selesai dilakukan, dataset yang telah melalui proses pemeriksaan kemudian digunakan untuk tahap selanjutnya.

D. Modelling

Pada tahap *modelling*, model klasifikasi dibangun menggunakan algoritma yang telah dipilih dengan bantuan perangkat lunak *data mining*, yaitu RapidMiner, untuk menghasilkan model klasifikasi yang kemudian dievaluasi berdasarkan kinerja yang diperoleh [15]. Penelitian ini menggunakan empat algoritma *machine learning* sebagai model klasifikasi, yaitu sebagai berikut:

- 1) *Naïve Bayes*: Algoritma ini menggunakan pendekatan berdasarkan Teorema Bayes untuk menentukan kelas suatu data [16]. Dalam prosesnya, algoritma ini menganggap setiap fitur tidak saling bergantung sehingga proses pembentukan model dapat dilakukan dengan lebih sederhana [17].
- 2) *Support Vector Machine*: SVM merupakan algoritma yang diterapkan pada data dengan karakteristik pemisahan secara linear maupun non-linear [18]. SVM bekerja dengan mencari batas pemisah (*hyperplane*) yang mampu memisahkan data berdasarkan kelasnya [19]. Kemampuan tersebut mendukung penerapan SVM dalam mengolah data dengan dimensi fitur teks yang kompleks [20].
- 3) *Decision Tree*: Algoritma ini melakukan proses klasifikasi dengan membentuk struktur seperti pohon yang merepresentasikan tahapan pengambilan keputusan. Algoritma ini dapat menggunakan *pre-pruning* dan *post-pruning* untuk mengurangi risiko *overfitting* [21].
- 4) *Random Forest*: Algoritma yang menggabungkan sejumlah *Decision Tree* untuk membentuk satu model klasifikasi [1]. Untuk menangani data dengan pola non-linier dan hubungan yang kompleks, pendekatan klasifikasi RF menggabungkan banyak pohon keputusan independen [22].

5) *Synthetic Minority Over-Sampling Technique (SMOTE)*: Merupakan teknik mengatasi ketidakseimbangan distribusi kelas dengan menghasilkan data sintetis pada kelas minoritas. Penambahan data tersebut bertujuan agar proses pelatihan model tidak didominasi oleh kelas mayoritas sehingga hasil klasifikasi menjadi lebih baik [23]. Sampel tersebut dihasilkan berdasarkan hubungan kedekatan antardata yang ditentukan menggunakan algoritma *k-nearest neighbor*, kemudian menghasilkan data sintetis baru melalui interpolasi antara sampel minoritas dan tetangganya, hal ini membantu mengurangi dominasi kelas mayoritas dalam proses pelatihan model.

6) *K-Fold Cross Validation*: Membagi dataset menjadi sejumlah bagian (*fold*) yang memiliki ukuran relatif seimbang. Salah satu bagian digunakan sebagai data pengujian dan bagian lainnya digunakan untuk melatih model, kemudian proses tersebut dilanjutkan secara bergantian hingga seluruh bagian memperoleh giliran sebagai data pengujian. Nilai *k* yang umum digunakan adalah 5 atau 10, terutama karena metode ini dapat memberikan estimasi kinerja model yang lebih stabil [11]. Metode ini banyak dipilih karena mampu menekan waktu komputasi tanpa mengorbankan keakuratan hasil estimasi.

7) *RapidMiner*: Aplikasi yang menyediakan berbagai operator untuk mendukung tahapan pengolahan data, mulai dari persiapan data, pemrosesan, analisis, hingga penyajian hasil dalam bentuk visualisasi yang digunakan untuk melakukan proses *data mining* [12].

E. Evaluation

Tahap *evaluation* dilakukan dengan tujuan mengetahui kinerja model dalam melakukan klasifikasi terhadap data. Evaluasi dilakukan dengan menerapkan metode *K-Fold Cross Validation*, dataset dibagi ke dalam sejumlah *fold* untuk proses pelatihan dan pengujian secara bergantian. Nilai *k* yang akan digunakan adalah 10, karena penggunaan nilai tersebut dapat memberikan estimasi kinerja model yang lebih stabil berdasarkan variasi data yang tersedia. Seluruh proses kemudian dijalankan sebanyak *k* iterasi agar setiap bagian data dapat digunakan sebagai pengujian data secara bergiliran. Pendekatan tersebut memungkinkan seluruh data berkontribusi dalam proses pelatihan maupun pengujian, sehingga hasil evaluasi dapat memberikan gambaran kinerja model yang lebih representatif. Setelah seluruh iterasi selesai, performa model ditinjau melalui *confusion matrix*.

Confusion matrix menyajikan hasil klasifikasi pada setiap kelas untuk memperlihatkan perbandingan hasil prediksi model dan kelas aktual, sehingga jumlah prediksi yang tepat maupun salah pada masing-masing kelas dapat diketahui [24]. Berdasarkan *confusion matrix*, kinerja model diukur menggunakan metrik *accuracy*, *precision*, *recall*, *F1-score*, dan *balanced accuracy* [23]. Perhitungan masing-masing metrik ditunjukkan pada persamaan berikut:

$$Accuracy = \frac{TP+TN}{TP+TN+FP+FN} \quad (6)$$

$$Precision = \frac{TP}{TP+FP} \quad (7)$$

$$Recall (Sensitivity) = \frac{TP}{TP+FN} \quad (8)$$

$$F1-Score = 2 \times \frac{Precision \times Recall}{Precision + Recall} \quad (9)$$

$$Balanced Accuracy = \frac{1}{2} \left(\frac{TP}{TP+FN} + \frac{TN}{TN+FP} \right) \quad (10)$$

F. Knowledge Discovery

Knowledge discovery merupakan tahapan yang digunakan untuk menemukan pola atau informasi tertentu dari data yang sebelumnya belum terlihat. Informasi yang diperoleh dapat digunakan untuk menghasilkan pengetahuan yang bermanfaat dari sekumpulan data dalam jumlah besar [9]. Melalui proses tersebut, data diolah menjadi pengetahuan yang lebih bermakna [25], sehingga menghasilkan wawasan baru yang dapat membantu penyusunan strategi dan pengambilan keputusan pada berbagai bidang [26], [27].

Dalam konteks pengukuran kemiskinan, setiap negara memiliki ukuran atau indikator kemiskinan yang berbeda. Menurut Badan Pusat Statistik (BPS), kemiskinan di Indonesia ditentukan berdasarkan kemampuan masyarakat dalam memenuhi kebutuhan dasar (*basic needs approach*) dinilai dari sisi pengeluaran untuk mengukur kemiskinan di Indonesia. Seseorang dikategorikan miskin apabila rata-rata pengeluaran per kapita per bulan berada di bawah garis kemiskinan (GK). Garis kemiskinan merupakan batas nilai pengeluaran yang digunakan untuk menentukan apakah seseorang dapat memenuhi kebutuhan dasarnya. Nilai tersebut terdiri atas garis kemiskinan makanan dan garis kemiskinan non-makanan. Garis kemiskinan makanan dihitung berdasarkan nilai minimum pengeluaran untuk memenuhi kebutuhan makanan, sedangkan garis kemiskinan non-makanan mencakup pengeluaran untuk memenuhi kebutuhan dasar lainnya, seperti kebutuhan perumahan, kebutuhan pendidikan untuk mendapatkan layanan pendidikan, kebutuhan kesehatan, kebutuhan pakaian sebagai sandang, transportasi untuk mendukung mobilitas, serta kebutuhan pokok lainnya [13].

Berbeda dengan pendekatan tersebut, World Bank mengembangkan *Multidimensional Poverty Measure* (MPM), yaitu pendekatan yang mengukur kemiskinan berdasarkan tiga dimensi utama, yaitu moneter, pendidikan, dan akses terhadap infrastruktur dasar [28]. Sementara itu, United Nations Development Programme (UNDP) menggunakan pendekatan *Multidimensional Poverty Index* (MPI) untuk mengukur kemiskinan berdasarkan tiga dimensi, yaitu kesehatan, pendidikan, dan standar hidup [29].

Dalam penerapannya, *knowledge discovery* dapat dilakukan melalui pemodelan *machine learning* untuk menganalisis data dan memperoleh pengetahuan berdasarkan hasil pemodelan [30]. Melalui proses *knowledge discovery*, data yang berupa kumpulan nilai statistik dari berbagai indikator diolah melalui beberapa tahapan, mulai dari tahap *business understanding* hingga tahap evaluasi untuk memperoleh pola atau informasi penting yang dapat mendukung proses klasifikasi tingkat kemiskinan pada setiap provinsi di Indonesia. *Knowledge discovery* juga dilakukan melalui interpretasi model terbaik menggunakan analisis *feature importance* untuk melihat tingkat kepentingan relatif setiap indikator prediktor dalam proses klasifikasi. Dengan demikian, pengetahuan yang dihasilkan tidak hanya berupa algoritma dengan kinerja terbaik, tetapi juga informasi mengenai indikator prediktor yang memiliki kepentingan relatif lebih besar dalam membedakan kategori tingkat kemiskinan pada dataset penelitian.

III. HASIL DAN PEMBAHASAN

Hasil penelitian ini mencakup perbandingan keempat model yang digunakan dalam menganalisis algoritma untuk mengklasifikasikan tingkat kemiskinan provinsi di Indonesia.

A. Business Understanding

Tahapan ini dilakukan untuk mengidentifikasi tujuan penelitian serta menyesuaikan kebutuhan analisis yang akan dilakukan. Hasil pada tahap ini akan digunakan sebagai dasar dalam menyusun perencanaan proses *data mining*, sehingga penelitian dapat dilakukan sesuai dengan alur yang telah ditetapkan [11]. Penelitian ini menggunakan algoritma klasifikasi, yaitu *Naïve Bayes*, *SVM*, *Decision Tree*, dan *Random Forest*, untuk mengolah data kemiskinan provinsi di Indonesia. Penerapan algoritma tersebut bertujuan untuk menghasilkan model yang mampu mengelompokkan provinsi berdasarkan kategori tingkat kemiskinan “Rendah” dan “Tinggi”. Selain itu, penelitian ini melakukan perbandingan terhadap kinerja keempat algoritma untuk memperoleh model dengan kemampuan klasifikasi terbaik.

B. Data Understanding

Pada tahapan ini, data yang digunakan bersumber dari data sekunder yang diperoleh melalui aplikasi Allstats BPS dan dikelola oleh Badan Pusat Statistik (BPS). Data tersebut mencakup tingkat kemiskinan pada seluruh provinsi di Indonesia selama periode 2018-2025. Penelitian ini menggunakan data tingkat provinsi dengan periode pengamatan tersebut untuk memperoleh gambaran kondisi kemiskinan antarprovinsi. Setiap data yang tersedia pada masing-masing tahun menunjukkan kondisi satu provinsi pada tahun tertentu. Data dari setiap tahun kemudian digabungkan untuk digunakan pada tahapan penelitian berikutnya. Hasil pengumpulan dan penggabungan data

tersebut menghasilkan sebanyak 304 observasi provinsi-tahun sebagai dataset awal penelitian.

Proses pengumpulan data dilakukan dengan mengambil data setiap indikator dari sumber resmi BPS, kemudian mengintegrasikan seluruh data ke dalam satu berkas Microsoft Excel dan menyimpannya dalam format CSV untuk digunakan pada tahap pengolahan dan analisis. Penelitian ini menggunakan lima variabel prediktor yang dipilih berdasarkan relevansinya dalam merepresentasikan kondisi sosial-ekonomi yang berkaitan dengan kemiskinan, yaitu rata-rata lama sekolah penduduk usia 15+ (tahun), Indeks Pembangunan Manusia (IPM), persentase rumah tangga yang memiliki akses terhadap sanitasi layak, persentase rumah tangga yang memiliki akses terhadap air minum layak, dan tingkat pengangguran terbuka (TPT). Kelima variabel tersebut merepresentasikan beberapa aspek sosial-ekonomi dan pemilihan variabel tersebut didukung oleh penelitian terdahulu yang menggunakan indikator sosial-ekonomi dalam klasifikasi tingkat kemiskinan [10], [31], serta pendekatan pengukuran kemiskinan multidimensi yang mempertimbangkan aspek pendidikan, kesehatan, dan akses terhadap layanan dasar [28], [29]. Selain kelima variabel prediktor tersebut, Indeks Kedalaman Kemiskinan (P1) digunakan secara khusus sebagai dasar dalam pembentukan label tingkat kemiskinan. P1 tidak digunakan sebagai variabel prediktor dalam proses klasifikasi karena penggunaannya sebagai dasar pembentukan label dapat menyebabkan kebocoran data (*data leakage*) apabila digunakan kembali sebagai fitur.

C. Data Preparation

Pada tahap ini, dataset dipersiapkan sebagai bagian dari proses pengolahan sebelum pemodelan menggunakan perangkat lunak *data mining*, yaitu RapidMiner [32]. Dataset awal yang akan digunakan pada penelitian berjumlah 304 sampel data. Tahap awal *data preparation* dilakukan menggunakan Google Colab untuk melakukan pemilihan fitur dan *preprocessing* dataset sebelum memasuki tahap pemodelan. Pemilihan fitur dilakukan dengan mempertimbangkan atribut yang sesuai untuk digunakan dalam proses pemodelan. Fitur provinsi dan tahun tidak digunakan sebagai variabel prediktor karena keduanya hanya berfungsi sebagai penanda identitas observasi, bukan sebagai atribut yang digunakan untuk proses klasifikasi. Selanjutnya, dilakukan proses *preprocessing* menggunakan Google Colab meliputi pembersihan data (*data cleansing*) untuk memastikan dataset yang digunakan berada dalam kondisi siap untuk tahap selanjutnya. Proses pembersihan data dilakukan untuk menangani data yang memiliki nilai kosong (*missing value*) [32]. Pada tahap pengolahan data, nilai yang kosong tersebut dapat ditangani melalui beberapa cara, yaitu mengisi nilai yang kosong menggunakan nilai rata-rata (*mean imputation*) atau menghapus data yang memiliki nilai kosong [24]. Dalam penelitian ini, peneliti memilih pendekatan kedua, yaitu menghapus data yang mengandung

nilai kosong. Dari total 304 data awal, ditemukan 24 data dengan nilai kosong sehingga dihapus dari dataset. Dengan demikian, hanya 280 data yang memiliki nilai lengkap pada seluruh fitur yang digunakan dalam tahap analisis selanjutnya. Setelah proses *preprocessing* selesai, dilakukan proses pemberian label menggunakan *threshold*.

Threshold berdasarkan rata-rata P1 ditetapkan sebagai aturan klasifikasi tetap untuk seluruh periode pengamatan 2018-2025. Penetapan *threshold* ini bertujuan untuk mempertahankan konsistensi dalam pembentukan kelas tingkat kemiskinan selama periode pengamatan. Adapun kriteria pembentukan label sebagai berikut:

1. Jika P1 (Indeks Kedalaman Kemiskinan) < rata-rata dari P1 maka diberikan label rendah.
2. Jika P1 (Indeks Kedalaman Kemiskinan) $\geq$ rata-rata dari P1 maka diberikan label tinggi.

Proses pelabelan dilakukan dengan menggunakan variabel Indeks Kedalaman Kemiskinan (P1) sebagai dasar pembentukan label berdasarkan nilai rata-rata (*threshold*). *Threshold* ditentukan berdasarkan nilai rata-rata P1 dari seluruh 280 data provinsi-tahun hasil proses *preprocessing* selama periode 2018-2025. Pemilihan rata-rata P1 sebagai *threshold* dilakukan untuk membentuk dua kategori relatif berdasarkan posisi observasi terhadap kondisi rata-rata selama periode penelitian. Kategori “Rendah” diberikan kepada data dengan nilai P1 di bawah *threshold*, sedangkan kategori “Tinggi” diberikan kepada data dengan nilai P1 sama dengan atau di atas *threshold*. *Threshold* ini tidak dimaksudkan sebagai batas resmi kemiskinan yang ditetapkan BPS, sehingga kategori “Rendah” dan “Tinggi” merupakan label yang dibentuk secara khusus dalam penelitian ini. Penelitian lainnya [33] menggunakan *threshold* sebagai dasar pembentukan label data dengan mengelompokkan tingkat kedalaman kemiskinan ke dalam kode tertentu, yaitu 0 (rendah) dan 1 (tinggi), sebagai dasar proses klasifikasi. Setelah proses pelabelan selesai, variabel P1 dihapus dari dataset untuk memastikan variabel tersebut tidak digunakan sebagai fitur selama proses klasifikasi sehingga dapat mencegah terjadinya kebocoran data (*data leakage*). Proses pengolahan data menggunakan Google Colab menghasilkan fitur baru, yaitu tingkat kemiskinan sebagai label. Dari pengolahan tersebut diperoleh 181 data pada tingkat kemiskinan “Rendah” dan 99 data pada tingkat kemiskinan “Tinggi”, seperti ditampilkan pada Gambar 2.

```
... Rata-rata P1 = 1.8619642857142853

Distribusi kelas:
Tingkat Kemiskinan
Rendah      181
Tinggi       99
Name: count, dtype: int64

Ukuran dataset akhir: (280, 6)
```

Gambar 2. Proses Data Labeling

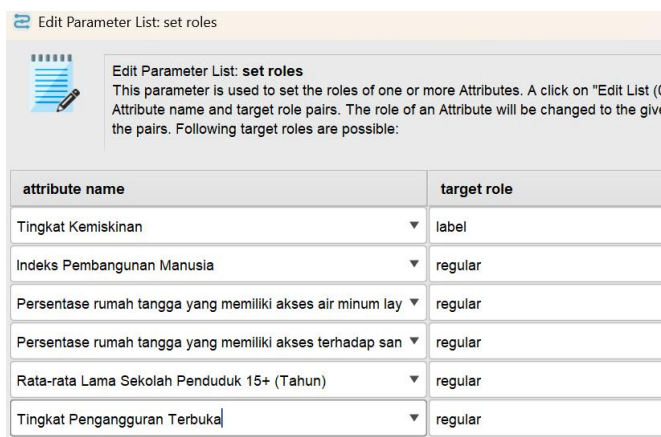
Data hasil *preprocessing* dan pelabelan menggunakan Google Colab selanjutnya disimpan dalam format CSV agar dapat diproses lebih lanjut di RapidMiner. Dataset tersebut selanjutnya dibaca melalui operator *Read CSV*, yang dapat dilihat pada Gambar 3.

Row No.	Rata-rata L...	Indeks Pem...	Persentase ...	Persentase ...	Tingkat Pen...	Tingkat Ke...
1	10.420	76.230	82.210	92.060	5.570	Tinggi
2	9.950	75.360	81.100	90.080	5.650	Tinggi
3	9.890	74.700	78.850	89.740	5.890	Tinggi
4	9.790	74.110	77.480	89.700	6.070	Tinggi
5	9.770	73.480	77.550	88.790	6.300	Tinggi
6	9.710	73.290	77.060	87.660	6	Tinggi
7	9.590	71.900	73.160	85.810	5.830	Tinggi
8	9.460	71.190	67.090	66.480	6.440	Tinggi
9	10.290	76.470	87.470	93.240	5.190	Rendah
10	10.180	75.760	85.730	92.940	5.350	Rendah
11	10.070	75.130	84.180	92.190	5.560	Rendah
12	9.990	74.510	82.300	92.130	5.820	Rendah
13	9.880	73.840	82.020	90.890	6.170	Rendah

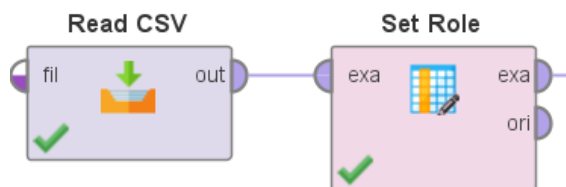
ExampleSet (280 examples, 0 special attributes, 6 regular attributes)

Gambar 3. Data Tingkat Kemiskinan

Berdasarkan hasil dari pembacaan dataset, diperoleh 280 sampel data. Selanjutnya, dilakukan penetapan peran masing-masing fitur menggunakan operator *Set Role*, seperti yang ditunjukkan pada Gambar 4, dengan menetapkan fitur tingkat kemiskinan sebagai label, sementara fitur lainnya tetap sebagai regular. Gambar 5 menunjukkan penggunaan operator pada proses pengolahan data di RapidMiner.



Gambar 4. Operator Set Role

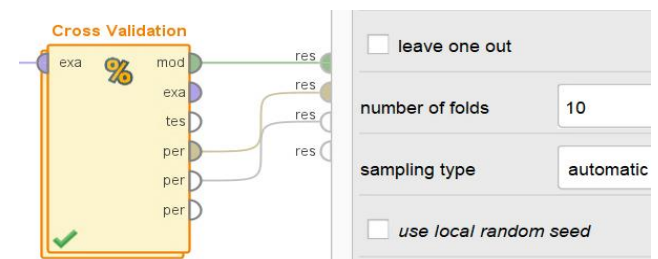


Gambar 5. Penggunaan Operator RapidMiner

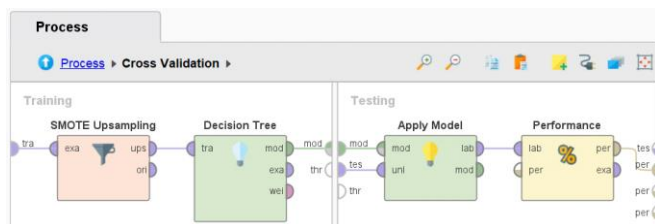
D. Modelling

Tahap *modelling* dilakukan dengan membangun model klasifikasi menggunakan empat algoritma *machine learning*. Berdasarkan hasil *data preparation*, terdapat 181 data dengan label “Rendah” dan 99 data dengan label “Tinggi”. Kondisi tersebut menunjukkan adanya ketidakseimbangan jumlah data antarkelas, sehingga diterapkan SMOTE untuk menyeimbangkan distribusi data dengan menghasilkan sampel sintesis pada kelas yang jumlahnya lebih sedikit.

Proses validasi dilakukan menggunakan *10-Fold Cross Validation*, yaitu dengan membagi dataset menjadi 10 bagian (*fold*). Pada setiap iterasi, satu *fold* digunakan sebagai data pengujian dan sembilan *fold* lainnya digunakan sebagai data pelatihan. Proses tersebut dilakukan hingga seluruh *fold* memperoleh giliran sebagai data pengujian. Pada setiap iterasi *10-Fold Cross Validation*, model dibangun menggunakan data pelatihan yang telah diseimbangkan dengan SMOTE [11]. SMOTE ditempatkan pada *training subprocess* di dalam operator *K-Fold Cross Validation* pada RapidMiner. Dengan demikian, pada setiap iterasi, data pelatihan terlebih dahulu diseimbangkan menggunakan SMOTE sebelum digunakan untuk membangun model, sedangkan data pengujian tidak melalui proses SMOTE dan tetap digunakan dalam kondisi aslinya [34]. Proses SMOTE pada penelitian ini menggunakan parameter $k=5$ yang menunjukkan bahwa pembentukan sampel baru dilakukan berdasarkan 5 data pada kelas minoritas yang memiliki karakteristik paling dekat. Selanjutnya data pelatihan hasil SMOTE digunakan untuk menjalankan algoritma *machine learning* yang telah dipilih. Model yang terbentuk kemudian digunakan untuk menghasilkan prediksi terhadap pengujian data. Setelah seluruh 10 iterasi selesai, hasil evaluasi dari setiap *fold* kemudian dihitung untuk memperoleh nilai kinerja model secara keseluruhan. Gambar 6 dan Gambar 7 menunjukkan penerapan metode SMOTE dan *10-Fold Cross Validation* dalam proses pemodelan.



Gambar 6. Penggunaan 10-Fold Cross Validation



Gambar 7. Penggunaan 10-Fold Cross Validation, SMOTE, dan Algoritma Terpilih

Pada bagian ini, pengaturan parameter untuk masing-masing algoritma disesuaikan dengan konfigurasi yang digunakan dalam penelitian. Pada algoritma *Random Forest*, digunakan 100 *trees* dengan *maximal depth* sebesar 10, dan kriteria pemisahan *gain ratio*. Pada *Decision Tree*, digunakan kriteria *gain ratio*, *maximal depth* sebesar 10, *pruning* dan *pre-pruning* diaktifkan, dengan *confidence* sebesar 0,1 dan *minimal gain* 0,01. Pada algoritma SVM, digunakan *dot kernel* dengan nilai C sebesar 1.0 dan *maximum iterations* 100.000. Sementara itu, algoritma *Naïve Bayes* menggunakan *Laplace correction* dalam proses estimasi probabilitas. Setelah proses pelatihan selesai, model digunakan untuk melakukan klasifikasi pada data pengujian guna memperoleh hasil prediksi untuk setiap *fold* pada proses validasi.

Dalam penelitian ini 10-Fold Cross Validation memperlakukan setiap observasi provinsi-tahun sebagai unit data dalam proses pembagian *fold*. Dengan demikian, observasi dari provinsi yang sama pada tahun yang berbeda dapat berada pada *fold* yang berbeda, sehingga karakteristik suatu provinsi pada tahun tertentu dapat muncul pada data pelatihan ketika tahun lainnya digunakan sebagai data pengujian. Kondisi tersebut menjadi keterbatasan dalam penelitian karena observasi antar tahun pada provinsi yang sama tidak sepenuhnya independen, sehingga hasil evaluasi model dalam penelitian ini perlu dipahami sebagai gambaran kemampuan klasifikasi pada data provinsi-tahun yang digunakan, bukan sebagai ukuran kemampuan model dalam memprediksi provinsi yang sama pada tahun berbeda. Hasil evaluasi tersebut juga belum sepenuhnya menunjukkan kemampuan model dalam mengklasifikasikan provinsi yang belum pernah digunakan dalam pelatihan atau memprediksi data pada tahun yang belum diamati. Pada penelitian selanjutnya, proses evaluasi dapat mempertimbangkan penggunaan *grouped cross-validation* berdasarkan provinsi atau *time-based validation* berdasarkan tahun untuk menguji kemampuan model generalisasi pada provinsi atau periode yang belum digunakan dalam pelatihan.

E. Evaluation

Tahap evaluasi dilakukan untuk melihat hasil kinerja sekaligus membandingkan kemampuan empat algoritma *machine learning* yang telah diterapkan. Pengujian dilakukan menggunakan 10-Fold Cross Validation. *Confusion matrix* digunakan untuk melihat hasil prediksi model pada masing-

masing kelas, yaitu “Rendah” dan “Tinggi”. Berdasarkan *confusion matrix*, kinerja setiap model kemudian dihitung menggunakan nilai *accuracy*, *precision*, *recall*, F1-score, dan *balanced accuracy*. Nilai dari setiap *fold* selanjutnya digabungkan untuk mendapatkan nilai akhir masing-masing algoritma. Hasil tersebut digunakan sebagai dasar dalam membandingkan keempat algoritma untuk menentukan model yang paling sesuai dalam mengklasifikasikan tingkat kemiskinan provinsi di Indonesia ke dalam kategori “Rendah” dan “Tinggi”. Algoritma dengan nilai evaluasi yang paling baik kemudian ditetapkan sebagai model terbaik berdasarkan hasil perbandingan tersebut.

1) Naïve Bayes

Gambar 8 dan Gambar 9 menunjukkan hasil laporan klasifikasi untuk model *Naïve Bayes*.

accuracy: 72.50% +/- 8.08% (micro average: 72.50%)

	true Tinggi	true Rendah	class precision
pred. Tinggi	42	20	67.74%
pred. Rendah	57	161	73.85%
class recall	42.42%	88.95%	

Gambar 8. Hasil Klasifikasi *Naïve Bayes* Sebelum SMOTE

accuracy: 72.50% +/- 8.26% (micro average: 72.50%)

	true Tinggi	true Rendah	class precision
pred. Tinggi	52	30	63.41%
pred. Rendah	47	151	76.26%
class recall	52.53%	83.43%	

Gambar 9. Hasil Klasifikasi *Naïve Bayes* Sesudah SMOTE

Berdasarkan hasil klasifikasi, model *Naïve Bayes* sebelum penerapan SMOTE memperoleh akurasi sebesar 72,50%, dengan *recall* kategori “Tinggi” sebesar 42,42% dan kategori “Rendah” sebesar 88,95%, sebagaimana ditunjukkan pada Gambar 8. Setelah penerapan SMOTE, akurasi model tetap sebesar 72,50%, sedangkan *recall* kategori “Tinggi” meningkat menjadi 52,53% dan *recall* kategori “Rendah” menurun menjadi 83,43%. Penerapan SMOTE menghasilkan nilai akurasi yang tetap sama, sementara kemampuan model dalam mengidentifikasi kategori “Tinggi” mengalami peningkatan.

Berdasarkan *confusion matrix* pada Gambar 9, sebanyak 52 data kategori “Tinggi” diklasifikasikan dengan benar sebagai “Tinggi” dan 151 data kategori “Rendah” diklasifikasikan dengan benar sebagai “Rendah”. Sementara itu, terdapat 47 data kategori “Tinggi” yang diklasifikasikan sebagai “Rendah” dan 30 data kategori “Rendah” yang diklasifikasikan sebagai “Tinggi”. Dari sisi *precision*, kategori “Tinggi” memperoleh nilai 63,41%, sedangkan kategori “Rendah” sebesar 76,26%. Selanjutnya, nilai F1-

score dan *balanced accuracy* dihitung berdasarkan nilai *precision* dan *recall* yang diperoleh dari *confusion matrix*.

$$F_1 = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}}$$

$$F_{1,Tinggi} = 2 \times \frac{0.6341 \times 0.5253}{0.6341 + 0.5253} = 0.5746 = 57.46\%$$

$$F_{1,Rendah} = 2 \times \frac{0.7626 \times 0.8343}{0.7626 + 0.8343} = 0.7968 = 79.68\%$$

$$F_{1-macro} = \frac{F_{1,Tinggi} + F_{1,Rendah}}{2}$$

$$F_{1-macro} = \frac{57.46\% + 79.68\%}{2} = 68.57\%$$

$$\text{Balanced Accuracy} = \frac{\text{Recall}_{Tinggi} + \text{Recall}_{Rendah}}{2}$$

$$\text{Balanced Accuracy} = \frac{0.5253 + 0.8343}{2} = 0.6798 = 67.98\%$$

Hasil perhitungan menunjukkan bahwa F1-score kategori “Tinggi” sebesar 57,46%, sedangkan kategori “Rendah” sebesar 79,68%, dengan nilai F1-score rata-rata (*macro average*) sebesar 68,57%. Sementara itu, nilai *balanced accuracy* yang diperoleh sebesar 67,98%.

2) Support Vector Machine

Gambar 10 dan Gambar 11 menunjukkan hasil laporan klasifikasi untuk model SVM.

accuracy: 77.14% +/- 7.18% (micro average: 77.14%)

	true Tinggi	true Rendah	class precision
pred. Tinggi	65	30	68.42%
pred. Rendah	34	151	81.62%
class recall	65.66%	83.43%	

Gambar 10. Hasil Klasifikasi SVM Sebelum SMOTE

accuracy: 76.07% +/- 9.23% (micro average: 76.07%)

	true Tinggi	true Rendah	class precision
pred. Tinggi	70	38	64.81%
pred. Rendah	29	143	83.14%
class recall	70.71%	79.01%	

Gambar 11. Hasil Klasifikasi SVM Sesudah SMOTE

Berdasarkan hasil klasifikasi, model SVM sebelum penerapan SMOTE memperoleh akurasi sebesar 77,14%, dengan *recall* kategori “Tinggi” sebesar 65,66% dan kategori “Rendah” sebesar 83,43%, sebagaimana ditunjukkan pada Gambar 10. Setelah penerapan SMOTE, akurasi model menjadi 76,07%, dengan *recall* kategori “Tinggi” sebesar 70,71% dan kategori “Rendah” sebesar 79,01%. Hasil tersebut menunjukkan bahwa penerapan SMOTE menyebabkan penurunan akurasi sebesar 1,07%, tetapi meningkatkan kemampuan model dalam mengenali kategori “Tinggi”.

Berdasarkan *confusion matrix* pada Gambar 11, sebanyak 70 data kategori “Tinggi” diklasifikasikan dengan benar sebagai “Tinggi” dan 143 data kategori “Rendah” diklasifikasikan dengan benar sebagai “Rendah”. Sementara itu, terdapat 29 data kategori “Tinggi” yang diklasifikasikan sebagai “Rendah” dan 38 data kategori “Rendah” yang diklasifikasikan sebagai “Tinggi”. Dari hasil tersebut, model memiliki nilai *recall* sebesar 79,01%, sedangkan *recall* kategori “Tinggi” sebesar 70,71%. Dari sisi *precision*, kategori “Tinggi” memperoleh nilai 64,81%, sedangkan kategori “Rendah” sebesar 83,14%. Selanjutnya, nilai F1-score dan *balanced accuracy* dihitung berdasarkan nilai *precision* dan *recall* yang diperoleh dari *confusion matrix*.

$$F_1 = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}}$$

$$F_{1,Tinggi} = 2 \times \frac{0.6481 \times 0.7071}{0.6481 + 0.7071} = 0.6763 = 67.63\%$$

$$F_{1,Rendah} = 2 \times \frac{0.8314 \times 0.7901}{0.8314 + 0.7901} = 0.8102 = 81.02\%$$

$$F_{1-macro} = \frac{F_{1,Tinggi} + F_{1,Rendah}}{2}$$

$$F_{1-macro} = \frac{67.63\% + 81.02\%}{2} = 74.33\%$$

$$\text{Balanced Accuracy} = \frac{\text{Recall}_{Tinggi} + \text{Recall}_{Rendah}}{2}$$

$$\text{Balanced Accuracy} = \frac{70.71\% + 79.01\%}{2} = 74.86\%$$

Hasil perhitungan menunjukkan bahwa F1-score kategori “Tinggi” sebesar 67,63%, sedangkan kategori “Rendah” sebesar 81,02%, dengan nilai F1-score rata-rata (*macro average*) sebesar 74,33%. Sementara itu, nilai *balanced accuracy* yang diperoleh sebesar 74,86%. Hasil tersebut perlu diinterpretasikan dengan mempertimbangkan bahwa pada penelitian ini tidak dilakukan proses normalisasi atau standarisasi data sebelum pemodelan. Kondisi tersebut dapat menjadi keterbatasan dalam penerapan SVM karena algoritma SVM sensitif terhadap perbedaan skala antarfitur, sehingga performa SVM yang dihasilkan pada penelitian ini perlu dipahami berdasarkan kondisi *preprocessing* yang digunakan.

3) Decision Tree

Gambar 12 dan Gambar 13 menunjukkan hasil laporan klasifikasi untuk model *Decision Tree*.

accuracy: 81.79% +/- 8.32% (micro average: 81.79%)

	true Tinggi	true Rendah	class precision
pred. Tinggi	70	22	76.09%
pred. Rendah	29	159	84.57%
class recall	70.71%	87.85%	

Gambar 12. Hasil Klasifikasi *Decision Tree* Sebelum SMOTE

accuracy: 80.00% +/- 7.75% (micro average: 80.00%)

	true Tinggi	true Rendah	class precision
pred. Tinggi	67	24	73.63%
pred. Rendah	32	157	83.07%
class recall	67.68%	86.74%	

Gambar 13. Hasil Klasifikasi *Decision Tree* Sesudah SMOTE

Berdasarkan hasil klasifikasi, model *Decision Tree* sebelum penerapan SMOTE memperoleh akurasi sebesar 81,79%, dengan *recall* kategori “Tinggi” sebesar 70,71% dan kategori “Rendah” sebesar 87,85%, sebagaimana ditunjukkan pada Gambar 12. Setelah penerapan SMOTE, akurasi model menjadi 80,00%, dengan *recall* kategori “Tinggi” sebesar 67,68% dan kategori “Rendah” sebesar 86,74%. Berdasarkan hasil tersebut, penerapan SMOTE pada model *Decision Tree* tidak meningkatkan kemampuan klasifikasi secara keseluruhan.

Berdasarkan *confusion matrix* pada Gambar 13, sebanyak 67 data kategori “Tinggi” diklasifikasikan dengan benar sebagai “Tinggi” dan 157 data kategori “Rendah” diklasifikasikan dengan benar sebagai “Rendah”. Sementara itu, terdapat 32 data kategori “Tinggi” yang diklasifikasikan sebagai “Rendah” dan 24 data kategori “Rendah” yang diklasifikasikan sebagai “Tinggi”. Dari sisi *precision*, kategori “Tinggi” memperoleh nilai 73,63%, sedangkan kategori “Rendah” sebesar 83,07%. Nilai *recall* kategori “Tinggi” sebesar 67,68%, sedangkan kategori “Rendah” sebesar 86,74%. Selanjutnya, nilai *F1-score* dan *balanced accuracy* dihitung berdasarkan nilai *precision* dan *recall* yang diperoleh dari *confusion matrix*.

$$F_1 = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}}$$

$$F_{1,\text{Tinggi}} = 2 \times \frac{0.7363 \times 0.6768}{0.7363 + 0.6768} = 0.7053 = 70.53\%$$

$$F_{1,\text{Rendah}} = 2 \times \frac{0.8307 \times 0.8674}{0.8307 + 0.8674} = 0.8487 = 84.87\%$$

$$F_{1-\text{macro}} = \frac{F_{1,\text{Tinggi}} + F_{1,\text{Rendah}}}{2}$$

$$F_{1-\text{macro}} = \frac{70.53\% + 84.87\%}{2} = 77.70\%$$

$$\text{Balanced Accuracy} = \frac{\text{Recall}_{\text{Tinggi}} + \text{Recall}_{\text{Rendah}}}{2}$$

$$\text{Balanced Accuracy} = \frac{67.68\% + 86.74\%}{2} = 77.21\%$$

Hasil perhitungan menunjukkan bahwa *F1-score* kategori “Tinggi” sebesar 70,53%, sedangkan kategori “Rendah” sebesar 84,87%, dengan nilai *F1-score* rata-rata (*macro average*) sebesar 77,70%. Sementara itu, nilai *balanced accuracy* yang diperoleh sebesar 77,21%.

4) Random Forest

Gambar 14 dan Gambar 15 menunjukkan hasil laporan klasifikasi untuk model *Random Forest*.

accuracy: 82.50% +/- 8.15% (micro average: 82.50%)

	true Tinggi	true Rendah	class precision
pred. Tinggi	68	18	79.07%
pred. Rendah	31	163	84.02%
class recall	68.69%	90.06%	

Gambar 14. Hasil Klasifikasi *Random Forest* Sebelum SMOTE

accuracy: 81.79% +/- 8.49% (micro average: 81.79%)

	true Tinggi	true Rendah	class precision
pred. Tinggi	78	30	72.22%
pred. Rendah	21	151	87.79%
class recall	78.79%	83.43%	

Gambar 15. Hasil Klasifikasi *Random Forest* Sesudah SMOTE

Berdasarkan hasil klasifikasi, model *Random Forest* sebelum penerapan SMOTE memperoleh akurasi sebesar 82,50%, dengan nilai *recall* kategori “Tinggi” sebesar 68,69% dan kategori “Rendah” sebesar 90,06%, sebagaimana ditunjukkan pada Gambar 14. Setelah penerapan SMOTE, model memperoleh akurasi sebesar 81,79%. SMOTE tidak selalu meningkatkan akurasi, tetapi cenderung meningkatkan kemampuan model dalam mengenali kelas minoritas. Meskipun terjadi penurunan akurasi sebesar 0,71%, kemampuan model dalam mengenali kategori “Tinggi” mengalami peningkatan, yang ditunjukkan oleh nilai *recall* sebesar 78,79%, sedangkan *recall* kategori “Rendah” sebesar 83,43%. Peningkatan *recall* pada kategori “Tinggi” menunjukkan bahwa penerapan SMOTE membantu model dalam mengenali lebih banyak observasi dari kelas minoritas.

Berdasarkan *confusion matrix* pada Gambar 15, sebanyak 78 data kategori “Tinggi” diklasifikasikan dengan benar sebagai “Tinggi” dan 151 data kategori “Rendah” diklasifikasikan dengan benar sebagai “Rendah”. Sementara itu, terdapat 21 data kategori “Tinggi” yang diklasifikasikan sebagai “Rendah” dan 30 data kategori “Rendah” yang diklasifikasikan sebagai “Tinggi”. Dari sisi *precision*, kategori “Tinggi” memperoleh nilai 72,22%, sedangkan kategori “Rendah” sebesar 87,79%. Selanjutnya, nilai *F1-score* dan *balanced accuracy* dihitung berdasarkan nilai *precision* dan *recall* yang diperoleh dari *confusion matrix*.

$$F_1 = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}}$$

$$F_{1,\text{Tinggi}} = 2 \times \frac{0.7222 \times 0.7879}{0.7222 + 0.7879} = 0.7536 = 75.36\%$$

$$F_{1,\text{Rendah}} = 2 \times \frac{0.8779 \times 0.8343}{0.8779 + 0.8343} = 0.8555 = 85.55\%$$

$$F_{1-\text{macro}} = \frac{F_{1,\text{Tinggi}} + F_{1,\text{Rendah}}}{2}$$

$$F_{1-\text{macro}} = \frac{75.36\% + 85.55\%}{2} = 80.46\%$$

$$\text{Balanced Accuracy} = \frac{\text{Recall}_{\text{Tinggi}} + \text{Recall}_{\text{Rendah}}}{2}$$

$$\text{Balanced Accuracy} = \frac{78.79\% + 83.43\%}{2} = 81.11\%$$

Hasil perhitungan menunjukkan bahwa *F1-score* kategori “Tinggi” sebesar 75,36%, sedangkan kategori “Rendah” sebesar 85,55%, dengan nilai *F1-score* rata-rata (*macro average*) sebesar 80,46%. Sementara itu, nilai *balanced accuracy* yang diperoleh sebesar 81,11%. Hasil tersebut menunjukkan bahwa meskipun penerapan SMOTE tidak meningkatkan akurasi keseluruhan model, penerapannya meningkatkan kemampuan model dalam mengenali kategori “Tinggi”. Kinerja model yang dievaluasi tidak hanya didasarkan pada akurasi, tetapi memerlukan pertimbangan melalui *F1-score* dan *balanced accuracy* untuk memberikan gambaran kinerja klasifikasi pada kedua kategori secara lebih seimbang. Berdasarkan hasil perbandingan kinerja seluruh model klasifikasi, algoritma *Random Forest* memperoleh kinerja terbaik dibandingkan algoritma lainnya berdasarkan perbandingan metrik evaluasi yang digunakan. Performa tersebut juga terlihat dari nilai *recall* yang relatif tinggi pada kedua kategori, yaitu 83,43% pada kategori “Rendah” dan 78,79% pada kategori “Tinggi”.

Perbedaan akurasi antara *Random Forest* dan *Decision Tree* sebesar 1,79% menunjukkan bahwa *Random Forest* memiliki kinerja yang lebih tinggi pada dataset penelitian. Namun, selisih tersebut relatif kecil sehingga keunggulan *Random Forest* perlu dipahami berdasarkan keseluruhan metrik evaluasi, terutama rata-rata makro *F1-score* dan akurasi seimbang. Oleh karena itu, dilakukan pengujian melalui *10-Fold Cross Validation* dengan menggunakan hasil dari setiap lipatan sebagai dasar perhitungan. Berdasarkan hasil tersebut, *Random Forest* dipilih sebagai model terbaik dalam penelitian ini karena memiliki nilai evaluasi yang lebih tinggi dibandingkan model lainnya. Untuk mengetahui kontribusi masing-masing indikator dalam proses klasifikasi, tahap berikutnya dilakukan analisis *feature importance*. Analisis ini digunakan untuk melihat indikator yang memiliki peran lebih besar dalam membedakan tingkat kemiskinan “Rendah” dan “Tinggi”.

attribute	weight ↓
Rata-rata Lama Sekolah Penduduk 15+ (Tahun)	0.297
Indeks Pembangunan Manusia	0.221
Persentase rumah tangga yang memiliki akses terhadap sanitasi Layak	0.193
Persentase rumah tangga yang memiliki akses air minum layak	0.150
Tingkat Pengangguran Terbuka	0.139

Gambar 16. *Feature Importance Random Forest*

Berdasarkan hasil *feature importance* pada Gambar 16, rata-rata lama sekolah menjadi indikator dengan nilai kepentingan paling besar, yaitu 0,297. Setelah itu, Indeks Pembangunan Manusia memperoleh nilai 0,221, akses sanitasi layak sebesar 0,193, akses air minum layak sebesar

0,150, dan tingkat pengangguran terbuka sebesar 0,139. Hasil tersebut menunjukkan bahwa rata-rata lama sekolah merupakan indikator yang paling banyak digunakan model dalam membedakan observasi ke dalam kategori tingkat kemiskinan “Rendah” dan “Tinggi” dalam dataset penelitian.

F. Knowledge Discovery

Tahap *knowledge discovery* dilakukan dengan menghubungkan hasil klasifikasi dengan analisis *feature importance* untuk melihat gambaran mengenai karakteristik indikator yang berperan dalam proses klasifikasi tingkat kemiskinan. Berdasarkan hasil perbandingan kinerja empat algoritma, *Random Forest* memperoleh akurasi sebesar 81,79%, lebih tinggi dibandingkan *Decision Tree* sebesar 80,00%, SVM sebesar 76,07%, dan *Naïve Bayes* sebesar 72,50%. Dengan hasil tersebut, *Random Forest* menjadi algoritma dengan kinerja klasifikasi paling baik pada dataset penelitian dibandingkan ketiga algoritma lainnya. Salah satu keunggulan *Random Forest* adalah kemampuannya menggunakan beberapa *Decision Tree* secara bersamaan, sehingga hasil prediksi yang diperoleh tidak hanya bergantung pada satu pohon keputusan. Hasil ini sejalan dengan penelitian [2], [10], dan [35] yang memperlihatkan bahwa *Random Forest* memiliki kinerja yang baik dalam mengklasifikasikan data sosial ekonomi.

Sebagai bagian dari proses *knowledge discovery*, analisis *feature importance* juga dilakukan pada *Random Forest* yang memiliki kinerja terbaik untuk mengetahui kepentingan relatif setiap variabel prediktor dalam proses klasifikasi. Hasil analisis menunjukkan bahwa rata-rata lama sekolah penduduk usia 15+ (tahun) memiliki nilai *importance* paling besar, yaitu sebesar 0,297.

Tingginya *feature importance* RLS menunjukkan bahwa indikator yang berkaitan dengan pendidikan memiliki kontribusi yang cukup besar dalam proses klasifikasi dua kategori pada penelitian ini. Temuan tersebut dapat dikaitkan dengan pendekatan multidimensional dalam pengukuran kemiskinan yang mempertimbangkan aspek pendidikan sebagai salah satu dimensinya [31]. Rata-rata lama sekolah dan IPM menempati dua peringkat tertinggi dengan nilai *importance* masing-masing sebesar 0,297 dan 0,221. Hasil ini menggambarkan bahwa pendidikan, terutama melalui rata-rata lama sekolah, menjadi salah satu indikator yang cukup berperan dalam membedakan tingkat kemiskinan. Nilai *feature importance* tersebut menunjukkan besarnya kontribusi relatif suatu indikator terhadap hasil klasifikasi model, tetapi tidak dapat diartikan sebagai hubungan sebab-akibat atau pengaruh langsung terhadap tingkat kemiskinan.

Temuan ini sejalan dengan pendekatan *Multidimensional Poverty Measure* (MPM) yang dikembangkan oleh World Bank dan *Multidimensional Poverty Index* (MPI) dari UNDP, yang menyatakan bahwa kemiskinan merupakan fenomena multidimensi sehingga tidak cukup dijelaskan

hanya melalui tingkat pendapatan atau pengeluaran, tetapi juga perlu mempertimbangkan dimensi pendidikan, kesehatan, serta akses terhadap layanan dasar dalam perumusan kebijakan pembangunan berkelanjutan [28], [29].

Dengan mempertimbangkan berbagai indikator tersebut secara bersamaan, proses *knowledge discovery* memberikan informasi mengenai kepentingan relatif setiap indikator dalam proses klasifikasi tingkat kemiskinan pada tingkat provinsi. Hasil *feature importance* menunjukkan bahwa rata-rata lama sekolah merupakan indikator dengan kepentingan relatif tertinggi dalam model *Random Forest*, sehingga memberikan informasi tambahan mengenai karakteristik indikator yang paling banyak digunakan model dalam membedakan kategori “Rendah” dan “Tinggi”. Namun, hasil *feature importance* tidak menunjukkan hubungan kausal dan penelitian ini tidak secara langsung mengevaluasi efektivitas maupun dampak suatu kebijakan pengentasan kemiskinan, sehingga pemanfaatannya dalam perumusan kebijakan memerlukan analisis lebih lanjut.

IV. KESIMPULAN

Hasil pada penelitian ini menunjukkan bahwa klasifikasi tingkat kemiskinan provinsi di Indonesia dapat dilakukan dengan membandingkan empat algoritma klasifikasi, yaitu *Naïve Bayes*, *SVM*, *Decision Tree*, dan *Random Forest*. Berdasarkan hasil evaluasi, *Random Forest* menunjukkan kinerja terbaik dengan akurasi sebesar 81,79%, *F1-score macro average* sebesar 80,46%, dan *balanced accuracy* sebesar 81,11%. Selanjutnya, *Decision Tree* memperoleh akurasi sebesar 80,00%, *F1-score macro average* sebesar 77,70%, dan *balanced accuracy* sebesar 77,21%. *SVM* memperoleh akurasi sebesar 76,07%, *F1-score macro average* sebesar 74,33%, dan *balanced accuracy* sebesar 74,86%, sedangkan *Naïve Bayes* memperoleh akurasi sebesar 72,50%, *F1-score macro average* sebesar 68,57%, dan *balanced accuracy* sebesar 67,98%. Evaluasi model dilakukan menggunakan *10-Fold Cross Validation*, sehingga hasil kinerja yang diperoleh menggambarkan kemampuan klasifikasi pada observasi provinsi-tahun dalam dataset penelitian. Hasil tersebut belum dapat secara langsung diinterpretasikan sebagai kemampuan model dalam memprediksi provinsi atau tahun yang sepenuhnya baru. Selanjutnya, analisis *feature importance* pada model *Random Forest* menunjukkan bahwa rata-rata lama sekolah penduduk usia 15+ (tahun) merupakan indikator dengan kepentingan relatif tertinggi dengan nilai *importance* sebesar 0,297. Hasil tersebut menunjukkan bahwa indikator pendidikan memiliki kepentingan relatif lebih besar dalam membedakan kategori tingkat kemiskinan “Rendah” dan “Tinggi”. Namun, *feature importance* tidak dapat diinterpretasikan sebagai hubungan sebab-akibat atau pengaruh langsung terhadap tingkat kemiskinan.

DAFTAR PUSTAKA

- [1] F. Ramayanti, D. Vionanda, D. Permana, and Z. Zilrahmi, “Application of Random Forest to Identify for Poor Households in West Sumatera Province,” *UNP J. Stat. Data Sci.*, vol. 1, no. 2, pp. 97–104, 2023.
- [2] S. Sukarna, K. A. Notodiputro, and B. Sartono, “Comparison of Logistic Model Tree and Random Forest on Classification for Poverty in Indonesia,” *Media Stat.*, vol. 16, no. 2, pp. 112–123, 2023.
- [3] M. D. Akhda and K. D. Tania, “Comparison of K-Means and K-Medoids Algorithms for Clustering Poverty Data in South Sumatra Using DBI Evaluation,” *Digit. Zo. J. Teknol. Inf. dan Komun.*, vol. 15, no. 2, pp. 233–245, 2024.
- [4] A. P. Priambodo and M. A. Djirimu, “Government intervention strategy in poverty reduction: Study on the district and city in Indonesia Across 2016-2023,” *J. Bina Praja*, vol. 16, no. 3, pp. 489–508, 2024.
- [5] M. Yanto, Y. Wiyandra, and S. Defit, “Deep learning approach analysis model prediction and classification poverty status,” *IAES Int. J. Artif. Intell.*, vol. 12, no. 1, p. 459, 2023.
- [6] R. H. Sachrrial and A. Iskandar, “Analisa Perbandingan Complete Linkage AHC dan K-Medoids Dalam Pengelompokan Data Kemiskinan di Indonesia,” *Build. Informatics, Technol. Sci.*, vol. 5, no. 2, 2023.
- [7] Z. Wahidah and D. T. Utari, “Comparison of K-means and Gaussian mixture model in profiling areas by poverty indicators,” *BAREKENG J. Ilmu Mat. dan Ter.*, vol. 17, no. 2, pp. 717–726, 2023.
- [8] M. Ravallion, “Poverty lines in theory and practice (English),” The World Bank, Washington, D.C., 1998. [Online]. Available: <http://documents.worldbank.org/curated/en/916871468766156239>
- [9] M. R. Maulizidan and K. D. Tania, “Knowledge discovery based on sentiment analysis of public perceptions about generative AI on X,” *IJIE (Indonesian J. Informatics Educ.)*, vol. 9, no. 2, pp. 92–107, 2025.
- [10] A. A. Hassan, A. H. Muse, and C. Chesneau, “Machine learning study using 2020 SDHS data to determine poverty determinants in Somalia,” *Sci. Rep.*, vol. 14, no. 1, p. 5956, 2024.
- [11] R. Nursyahfitri, C. Rozikin, and R. I. Adam, “Penerapan Metode SMOTE dalam Klasifikasi Daerah Rawan Banjir di Karawang Menggunakan Algoritma Naive Bayes,” *JUSTIN (Jurnal Sist. dan Teknol. Informasi)*, vol. 10, no. 4, pp. 339–346, 2022.
- [12] H. N. F. Fikrillah, “Klasifikasi Program Bantuan Sosial Menggunakan Algoritma K-Nearest Neighbor (K-NN)(Studi Kasus Kecamatan Malangbong Kabupaten Garut),” *J. Teknol. Inf. dan Ilmu Komput.*, vol. 12, no. 3, pp. 609–616, 2025.
- [13] B. P. Statistik, *Indikator Kesejahteraan Rakyat 2023*. Jakarta: Badan Pusat Statistik, 2023. [Online]. Available: <https://www.bps.go.id/id/publication/2023/11/06/7807339c2d4ae0ca8e0beaa/welfare-indicators-2023>.
- [14] L. A. Alfia, F. Nugroho, and Y. M. Arif, “Comparative Analysis of Artificial Neural Networks, Linear Regression, Random Forest, and Support Vector Machine for Predicting Poverty Levels in Indonesia,” *Int. J. Adv. Data Inf. Syst.*, vol. 6, no. 3, pp. 789–800, 2025.
- [15] D. Kurniadi, F. Nuraeni, and M. Firmansyah, “Klasifikasi Masyarakat Penerima Bantuan Langsung Tunai Dana Desa Menggunakan Naïve Bayes dan SMOTE,” *J. Teknol. Inf. dan Ilmu Komput.*, vol. 10, no. 2, 2023.
- [16] P. Paramita and A. Ibrahim, “Analisis Sentimen Terhadap Pengguna Qris (Quick Respond Code Indonesian Standart) Pada Twitter Menggunakan Metode Naïve Bayes Classifier,” *JOISIE (Jurnal Inf. Syst. Informatics Eng.)*, vol. 7, no. 1, pp. 1–6, 2023.
- [17] Z. A. Mukharyahya, Y. P. Astuti, and O. N. Cahyani, “Perbandingan Naive Bayes Dan Support Vector Machine Dalam Klasifikasi Tingkat Kemiskinan Di Indonesia,” *Edumatic J. Pendidik. Inf.*, vol. 9, no. 1, pp. 119–128, 2025.

- [18] A. F. Rizkyllah, A. Meiriza, and D. Y. Hardiyanti, "Comparative Study of KNN and SVM Methods for Analyzing College Major Consistency Based on High School Background," *J. Sisfokom (Sistem Inf. dan Komputer)*, vol. 14, no. 4, pp. 429–436, 2025.
- [19] N. F. Palisoa, L. J. Sinay, M. Y. Matdoan, Y. Yudistira, and L. Bakarbesy, "Penerapan Support Vector Machine (SVM) untuk Klasifikasi Kabupaten Tertinggal di Provinsi Maluku," *Param. J. Mat. Stat. Dan Ter.*, vol. 2, no. 2, pp. 79–86, 2023.
- [20] S. M. Putri and K. D. Tania, "User Review Analysis: Psychological Impact of Duolingo Gamification Using Machine Learning," *J. Appl. Informatics Comput.*, vol. 10, no. 2, pp. 1886–1896, 2026.
- [21] J. Yang and J. Guan, "A Heart Disease Prediction Model Based on Feature Optimization and Smote-Xgboost Algorithm. Information 2022, 13, 475," 2022, ed: s *Note: MDPI stays neutral with regard to jurisdictional claims in published.*
- [22] J. Prasetya and A. Abdurakhman, "Comparison of smote random forest and smote k-nearest neighbors classification analysis on imbalanced data," *Media Stat.*, vol. 15, no. 2, pp. 198–208, 2023.
- [23] Y. A. Singgalen, "Penerapan CRISP-DM dalam Klasifikasi Sentimen dan Analisis Perilaku Pembelian Layanan Akomodasi Hotel Berbasis Algoritma Decision Tree (DT)," *J. Sist. Komput. dan Inform. Hal*, vol. 237, p. 248, 2023.
- [24] T. Gori, A. Sunyoto, and H. Al Fatta, "Preprocessing Data dan Klasifikasi untuk Prediksi Kinerja Akademik Siswa," *J. Teknol. Inf. dan Ilmu Komput.*, vol. 11, no. 1, pp. 215–224, 2024.
- [25] D. K. Harahap, K. D. Tania, and P. E. Sevtiyuni, "Topic Mining-Based Knowledge Discovery of User Health Information Needs," *Indones. J. Electron. Electromed. Eng. Med. Informatics*, vol. 7, no. 4, pp. 641–653, 2025.
- [26] M. N. Ramadhani, K. D. Tania, and M. Afrina, "Knowledge Discovery in Sharia Mobile Banking Reviews Using Aspect-Based Sentiment Analysis and Machine Learning," *J. Appl. Informatics Comput.*, vol. 10, no. 1, pp. 640–650, 2026.
- [27] V. Novalia, K. D. Tania, A. Meiriza, and A. Wedhasmara, "Knowledge Discovery of Application Review Using Word Embedding's Comparison with CNN-LSTM Model on Sentiment Analysis," in *2024 International Conference on Electrical Engineering and Computer Science (ICECOS)*, IEEE, 2024, pp. 234–238.
- [28] G. Lara Ibarra, M. C. Nguyen, D. Salmeron Gomez, and C. Haddad, "April 2026 Update to the Multidimensional Poverty Measure: What's New".
- [29] N. Suppa and U. Kanagaratnam, "The global Multidimensional Poverty Index: Harmonised level estimates and their changes over time," *Sci. Data*, vol. 12, no. 1, p. 153, 2025.
- [30] M. R. Maulana, K. D. Tania, and P. Sari, "Comparative Analysis of Machine Learning Models for Daily Urban Waste Volume's Knowledge Discovery," in *2025 International Conference on Artificial Intelligence and Technological Solutions (ICAITEch)*, IEEE, 2025, pp. 131–136.
- [31] N. A. Afifah, M. A. Ejung, M. D. Berti, R. Handayani, R. F. Syam, and I. Iskandar, "Explainable Machine Learning Approach to Classifying Poverty Levels Across Indonesia," *Integr. J. Artif. Intell. Data Sci.*, vol. 1, no. 2, pp. 57–66, 2025.
- [32] F. Febriansyah and S. Muntari, "Penerapan Algoritma K-Means untuk Klasterisasi Penduduk Miskin pada Kota Pagar Alam," *JISKA (Jurnal Inform. Sunan Kalijaga)*, vol. 8, no. 1, pp. 66–77, 2023.
- [33] N. P. N. Hendayanti and M. Nurhidayati, "Regresi logistik biner dalam penentuan ketepatan klasifikasi tingkat kedalaman kemiskinan provinsi-provinsi di Indonesia," *Sainstek J. Sains dan Teknol.*, vol. 12, no. 2, pp. 63–70, 2020.
- [34] G. C. Kurniawan, I. Riadi, and A. Yudhana, "Optimalisasi Fitur Backward Elimination Pada Klasifikasi Diabetes Mellitus Menggunakan Algoritma Random Forest," *JIPI (Jurnal Ilm. Penelit. dan Pembelajaran Inform.)*, vol. 11, no. 2, pp. 1272–1284, 2026.
- [35] N. N. Sholihah and A. Hermawan, "Implementation of random forest and smote methods for economic status classification in Cirebon City," *J. Tek. Inform.*, vol. 4, no. 6, pp. 1387–1397, 2023.