TUGAS AKHIR

PERBANDINGAN METODE *MACHINE LEARNING* DALAM KLASIFIKASI POTENSI KELUARGA BERESIKO *STUNTING* PADA KECAMATAN MERBAU

SYARIFAH KUSUMA MAHARANI

193510512

PROGRAM STUDI TEKNIK INFORMATIKA

FAKULTAS TEKNIK

UNIVERSITAS ISLAM RIAU

PEKANBARU

2024

HALAMAN PENGESAHAN TUGAS AKHIR

Nama : Syarifah Kusuma Maharani

NPM : 193510512

Kelompok Keahlian : Artificial Intelligence

Program Studi : Teknik Informatika

Jenjang Pendidikan : Strata Satu (S1)

Judul TA : Perbandingan Metode Machine Learning Dalam Klasifikasi Potensi Keluarga Beresiko Stunting Pada Kecamatan Merbau

Format sistematika dan pembahasan materi pada masing-masing bab dan sub bab dalam tugas akhir ini telah dipelajari dan dinilai relatif telah memenuhi ketentuan-ketentuan dan kriteria- kriteria dalam metode penelitian ilmiah. Oleh karena itu tugas akhir ini dinilai layak dapat disetujui untuk disidangkan dalam ujian **Seminar Tugas Akhir.**

Pekanbaru,6 Maret 2024

Di sahkan oleh :

Penguji I

Nesi Syafitri, S.Kom,. M.Sc
NIDN 0009088102

Penguji II

Ir. Des Suryani, M.Sc
NIDN 1026126801

Ketua Program Studi
Teknik Informatika

Dr. Apri Siswanto, S.Kom., M.Kom
NIDN 1016048502

Dosen Pembimbing

Dr. Arbi Haza Nasution, B.IT(Hons), M.IT
NIDN 1023048901

HALAMAN PENGESAHAN

DEWAN PENGUJI TUGAS AKHIR

Nama : Syarifah Kusuma Maharani
NPM : 193510512
Kelompok Keahlian : Artificial Intelligence
Program Studi : Teknik Informatika
Jenjang Pendidikan : Strata Satu (S1)
Judul TA : Perbandingan Metode Machine Learning Dalam Klasifikasi Potensi Keluarga Beresiko Stunting Pada Kecamatan Merbau

Tugas Akhir ini secara keseluruhan dinilai telah memenuhi ketentuan-ketentuan dan kaidah-kaidah dalam penulisan penelitian ilmiah serta telah diuji dan dapat dipertahankan dihadapan dewan penguji. Oleh karena itu, Tim Penguji Ujian Tugas Akhir Fakultas Teknik Universitas Islam Riau menyatakan bahwa mahasiswa yang bersangkutan dinyatakan Telah Lulus Mengikuti Ujian Tugas Akhir Pada Tanggal **2 April 2024** dan disetujui serta diterima untuk memenuhi salah satu syarat guna memperoleh gelar Sarjana Strata Satu Bidang Ilmu Teknik Informatika.

Pekanbaru, 18 April 2024

Dewan Penguji

1. Pembimbing : Dr. Arbi Haza Nasution,B.IT.(Hons), M.IT ( )
2. Penguji 1 : Nesi Syafitri, S.Kom., M.Cs ( )
3. Penguji 2 : Ir. Des Suryani, M.Sc ( )

Disahkan Oleh :

Ketua Program Studi
Teknik Informatika

Dr. Apri Siswanto,S.Kom.,M.Kom
NIDN.1016048502

PERNYATAAN KEASLIAN TUGAS AKHIR

Dengan ini saya menyatakan bahwa tugas akhir ini merupakan karya saya sendiri dan semua sumber yang tercantum didalamnya baik yang dikutip maupun dirujuk telah saya nyatakan dengan benar sesuai ketentuan. Jika terdapat unsur penipuan atau pemalsuan data maka saya bersedia dicabut gelar yang telah saya peroleh.

Pekanbaru, 02 April 2024

SYARIFAH KUSUMA MAHARANI
NPM : 193510512

KATA PENGANTAR

Dengan menyebut nama Allah SWT yang Maha Pengasih lagi Maha Panyayang, penulis ucapkan puji syukur atas kehadirat-Nya, yang telah melimpahkan rahmat, hidayah, dan inayah-Nya kepada penulis, sehingga penulis dapat menyelesaikan Skripsi yang berjudul "Perbandingan Metode *Machine Learning* Dalam Klasifikasi Potensi Keluarga Beresiko Stunting Pada Kecamatan Merbau" Skripsi ini telah penulis susun dengan maksimal dan mendapatkan bantuan dari berbagai pihak. Untuk itu penulis menyampaikan banyak terima kasih kepada semua pihak yang telah berkontribusi dalam pembuatan skripsi ini :

1. Bapak Dr. Eng. Muslim, ST., MT selaku Dekan Fakultas Teknik Universitas Islam Riau.
2. Bapak Dr. Apri Siswanto, S.Kom., M.Kom selaku Ketua Program Studi Teknik Informatika.
3. Ibu Ana Yulianti, ST., M.Kom selaku Sekretaris Program Studi Teknik Informatika.
4. Bapak Yudhi Arta, S.T, M.Kom selaku dosen PA yang telah memberikan masukan dan bimbingan selama melaksanakan perkuliahan.
5. Bapak Dr Arbi Haza Nasution B.IT.(Hons), M.IT selaku dosen pembimbing yang sangat banyak membantu, membimbing dan memberikan arahan sehingga penulis dapat menyelesaikan laporan skripsi ini dengan baik dan benar.

6. Ibu Nesi Syafitri, S.Kom., M.Cs dan Ibu Ir. Des Suryani, M.Sc selaku dosen penguji yang telah memberikan masukan dan arahan dalam membuat laporan skripsi ini.
7. Seluruh dosen Program Studi Teknik Informatika Universitas Islam Riau yang telah mendidik dan memberi arahan selama dibangku kuliah.
8. Badan Kependudukan dan Keluarga Berencana Nasional Provinsi Riau, yang telah sudi memberikan data dan informasi untuk penelitian penulis.
9. Teristimewa kepada kedua orang tua, almarhum Bapak Said Zamri, S.E yang semasa hidup selalu memberikan dukungan serta pengorbanan kepada penulis, semoga beliau dihadiahkan surga nya Allah SWT. Dan Ibu Sunarni yang telah memberikan dukungan moril maupun materil serta telah mendidik dengan penuh kasih sayang, dan doa yang tidak pernah putus selalu mengiringi langkah penulis sehingga dapat menyelesaikan skripsi ini. Terima kasih juga kepada yang tersayang, adik Said Afif Al-Khalifi, bude Suratmini, juga yang terkasih seluruh keluarga besar Said Mustafa dan Ali Rajiman yang telah memberikan doa juga semangat kepada penulis.
10. Kepada sahabat-sahabat penulis, Siti Sri Wati, Widiya Juniarti, Nurul Syahira, Intan Bunaya Putri, Ira Sinta Azlina, Ragil Anugrah Febrianti, Nabila, Yovita Dwi Ananda, dan Diondy Okto Dwi Juandi yang telah memberikan banyak perhatian, dukungan dan motivasi kepada penulis.

11. Teman-teman seperjuangan penulis, Agus Harianty, Nurhusna Salsabila, Juwita Novi Maradika, Nardianas Silalahi, Rezky Byon, Rizky Hendriawan, Kholilurohman, Ferdi Alfarabi, dan Eet Mendra, yang telah memberikan dukungan, motivasi serta berbagai sumbangan pemikiran kepada penulis. Selanjutnya kepada rekan seperjuangan Teknik Informatika angkatan 2019 yang tidak bisa disebutkan satu persatu.
12. Terakhir terima kasih kepada diri sendiri, atas usaha dan kerja keras karena telah berhasil melewati salah satu proses di dalam kehidupan, jangan lupa bersyukur dan semoga terkabul doa baik yang sudah di langitkan, semoga hal-hal yang pernah membuat runtuh turut menjadi alasan untuk tumbuh, dan semoga terus ada senyum orang tua yang selalu di usahakan itu.

Terlepas dari semua itu, penulis menyadari sepenuhnya bahwa masih ada kekurangan dalam penulisan skripsi ini. Oleh karena itu dengan tangan terbuka penulis menerima segala saran dan kritik yang membangun dari pembaca untuk penyempurnaan laporan skripsi ini.

Akhir kata penulis berharap semoga skripsi ini dapat memberikan manfaat serta inspirasi.

Pekanbaru, 20 Maret 2024

Syarifah Kusuma Maharani

DAFTAR ISI

DAFTAR GAMBAR

DAFTAR TABEL

ABSTRAK

Stunting merupakan suatu kondisi dimana kurang gizi kronis yang disebabkan oleh asupan gizi yang kurang dalam jangka waktu yang cukup lama akibat pemberian makan yang tidak sesuai kebutuhan. Di Indonesia, menurut hasil Survei Gizi Bayi Indonesia (SSGBI) prevalensi stunting mencapai 21,6% di tahun 2022. Dalam hal ini, keluarga juga ikut dalam pemantauan, pemeriksaan status resiko pada keluarga beresiko stunting masih membutuhkan waktu cukup lama karena dilakukan secara manual juga rentan akan ketidakakuratan. Oleh karena itu, untuk mengklasifikasi keluarga dengan potensi stunting dapat dilakukan menggunakan algoritma machine learning yaitu *K-Nearest Neighbor, Decision Tree*, *Naïve Bayes*, dan *Random Forest*. Dataset yang digunakan diambil dari 3 jenis data yaitu data PK, KB dan KPD. Pengujian dilakukan dengan menerapakan metode *K-fold cross validation* untuk menghilangkan bias pada data. Hasil akurasi tertinggi pada K-fold cross validation yaitu algoritma *Decision Tree* mendapatkan nilai 99.76% dengan *precision* 99.73%, *recall* 99.58%, dan *f-1score* 99.65%. Pada algoritma *Random Forest* mendapatkan nilai accuracy 99.25%. Kemudian pada algoritma K-NN mendapatkan 92.42% dan akurasi terendah yaitu algoritma *Naïve Bayes* dengan nilai 41.69%. Dalam penelitian ini algoritma yang memiliki performa terbaik yaitu algoritma *Decision Tree* dalam melakukan klasifikasi keluarga beresiko stunting di Kecamatan Merbau.

Kata Kunci: *Decision Tree, K-Nearest Neighbor, Klasifikasi, Machine Learning, Naïve Bayes, Random Forest*

ABSTRACT

Stunting is a condition where chronic malnutrition is caused by insufficient nutritional intake over a long period of time due to administration eating food that doesn't meet your needs. In Indonesia, according to the results of the Infant Nutrition Survey Indonesia (SSGBI) stunting prevalence will reach 21.6% in 2022. In this case,The family also participates in monitoring, checking the risk status of the family The risk of stunting still takes quite a long time because it is done routinely Manuals are also prone to inaccuracies. Therefore, to classify families with the potential for stunting can be done using a machine algorithm learning, namely K-Nearest Neighbor, Decision Tree, Naïve Bayes, and Random Forest. The dataset used was taken from 3 types of data, namely PK, KB and KPD data. Testing was carried out by applying the K-fold cross validation method for eliminate bias in the data. The highest accuracy results are in K-fold cross validation namely the Decision Tree algorithm gets a score of 99.76% with a precision of 99.73%, recall 99.58%, and f-1score 99.65%. In the Random Forest algorithm we get accuracy value 99.25%. Then the K-NN algorithm gets 92.42% and The lowest accuracy is the Naïve Bayes algorithm with a value of 41.69%. In research This algorithm that has the best performance is the Deep Decision Tree algorithm classifying families at risk of stunting in Merbau District

Keywords: Decision Tree, K-Nearest Neighbor, Classification, Machine Learning, Naive Bayes, Random Forest

BAB I
PENDAHULUAN

1.1 Latar Belakang

Stunting atau kekurangan gizi adalah pertumbuhan fisik pada tidak sesuai dengan usia atau tinggi badan yang tidak normal (Yuliani et al., 2018). *Stunting* pada masa kanak-kanak mempunyai dampak jangka panjang terhadap sumber daya manusia, termasuk berkurangnya perkembangan fisik, dan pencapaian pendidikan, keterampilan kognitif, produktivitas tenaga kerja, dan upah yang lebih rendah (Akseer et al., 2022). Stunting adalah masalah kesehatan masyarakat yang utama di sebagian besar negara berkembang (Ponum et al., 2020). Meskipun stunting dapat dicegah dan diobati, namun tetap menjadi salah satu ancaman terbesar bagi kelangsungan hidup umat manusia dan pembangunan ekonomi secara global (Aheto & Dagne, 2021). *Stunting* juga diartikan dengan keadaan ketika tubuh sangat pendek atau tubuh pendek dengan berindikator dengan Tinggi Badan menurut Umur (TB/U) atau Panjang Badan menurut umur (PB/U) dengan ambang batas (*z-score*) antara -3 SD sampai dengan < -2 SD (Olsa et al., 2018).

Prevalensi dari *stunting* sangatlah tinggi, berdasarkan data dari Global Nutritional Report terdapat kurang lebih 22,3% atau 150,9 juta bayi usia lima tahun *stunting*. *World Health Organization* (WHO) memutuskan 5 tempat sub regional keseluruhan *stunting*, salah satunya yaitu Indonesia. Indonesia terletak di wilayah Asia Tenggara mencapai 36,4% angka *stunting* (Rita Kirana, Aprianti, 2022). Pemerintah terus mengupayakan agar angka *stunting* di Indonesia tidak meningkat. Menurut Feni Sulistyawati dan Ni Putu Widarini (Sulistyawati & Widarini, 2022). Angka *stunting* di Indonesia mencapai 36,4% pada tahun 2014, hal ini terus menurun pada tahun 2018 angka *stunting* di Indonesia menurun sebanyak 6,4% hingga tahun 2019 terus mengalami penurunan menjadi 27,67%. Berdasarkan data dari (Ditjen Bangda, 2021). T erdapat beberapa provinsi yang menunjukkan angka *stunting* yang tinggi di Indonesia, yaitu 22,6% di Nusa Tenggara Timur, 21,07% di Nusa Tenggara

Barat, 21,0% di Kalimantan Barat, 19,3% di wilayah Sulawesi Barat, 18,5% di wilayah Kalimantan Utara. Sedangkan berdasarkan data dari (Mediacenter Riau, 2023) angka stunting di Riau mencapai 22,3%. Dan untuk Kabupaten Kepulauan Meranti, 23.3%. Meskipun angka tersebut tidak terlalu tinggi dibanding provinsi lain, namun *stunting* salah satu isu yang menjadi perhatian serta penanganan yang serius karena dapat berdampak pada pertumbuhan serta penurunan kecerdasan anak.

Tingkat *stunting* dapat diminimalisir dengan cara peruntutan indikasi-indikasi yang menyebabkan seseorang mengalami *stunting* (Satriawan, 2018) . Beberapa indikasi yang dapat meminimalisir tingkat stunting yaitu, fasilitas sanitasi yang layak, ditambah dengan gizi seimbang, terutama di daerah miskin masih sangat dibutuhkan. Tidak mengherankan, jika keluarga dengan akses terhadap sumber air yang tidak memadai memiliki kemungkinan lebih tinggi untuk mengalami stunting dibandingkan dengan rumah tangga dengan akses ke sumber air yang layak. Pola yang sama diamati untuk rumah tangga dengan lantai yang buruk dan rumah tangga yang terletak di daerah pedesaan. Dengan perkembangan teknologi modern, hal ini dimungkinkan untuk dilakukannya dengan teknik yang ada pada bidang ilmu data mining (Uwiringiyimana et al., 2022). Data mining adalah proses yang menggunakan berbagai alat analisis data untuk mengidentifikasi pola dan hubungan tersembunyi dalam data (Asha Kiranmai & Jaya Laxmi, 2018). Penelitian ini menggunakan teknik *data mining* yaitu klasifikasi, untuk mengklasifikasi potensi keluarga beresiko *stunting* pada Kecamatan Merbau. Pada penelitian ini, peneliti menggunakan beberapa algoritma *machine learning, Machine learning* adalah pemrograman komputer yang menggunakan kriteria/performa tertentu dengan sekumpulan data pelatihan atau pengalaman sebelumnya untuk mencapai standar tertentu. Algoritma *machine learning* dapat digunakan untuk berbagai tujuan(Chazar & Widhiaputra, 2020). Algoritma yang dikomparasikan yaitu algoritma *K-Nearest Neighbor, Random Forest, Decision Tree,* dan juga algoritma *Naïve Bayes.*

Penelitian dengan menggunakan algoritma *K-Nearest Neighbor* pernah dilakukan oleh Mustafa Qays Hatem (Hatem, 2022). Menghasilkan hasil akurasi yang

baik yaitu sebesar 98%, pada klasifikasi untuk mengetahui lesi kulit. Juga berdasarkan penelitian Li dkk (L. Li et al., 2023), menghasilkan akurasi 93,43% dengan menggunakan algoritma *Random Forest* untuk pengklasifikasian desain skema dan pembobotan di lingkungan binaan. Penelitian *Decision Tree* telah digunakan pada penelitian Zhao dkk mendapatkan akurasi 99,78% (Zhao et al., 2019). Dan model algoritma *Naïve Bayes* pernah digunakan dalam penelitian klasifikasi keyakinan orang oleh Donnellan dkk (Donnellan et al., 2022), dan mendapatkan akurasi sebesar 87,13

Dari penelitian terdahulu tersebut, masing-masing algoritma memiliki kelebihan dan ketepatan apabila diterapkan dalam *dataset* tertentu. Berdasarkan permasalahan serta penelitian terkait, maka penelitian ini akan melakukan klasifikasi potensi keluarga beresiko *stunting* dengan menggunakan 4 algoritma seperti *K-Nearest Neighbor, Random Forest, Decision Tree* dan *Naïve Bayes,* sehingga dari penelitian ini dapat mengkomparasikan efektivitas setiap algoritma yang digunakan pada klasifikasi potensi keluarga beresiko *stunting* Kecamatan Merbau.

1.2 Identifikasi Masalah

Stunting menjadi salah satu masalah yang harus mendapatkan penanganan secara serius. Untuk meminimalisir terjadinya *stunting*, keluarga dengan resiko *stunting* harus dipantau dengan ekstra, dalam hal ini diperlukan pengkategorian keluarga dengan resiko (tidak, rendah, tinggi). Pengkategorian ini dilakukan secara manual, yang cukup memakan waktu dan rawan ketidakakuratan ketika mengklasifikasikan kategori keluarga yang tidak beresiko *stunting*, keluarga dengan resiko *stunting* rendah, dan keluarga dengan resiko *stunting* tinggi. Maka dari itu diperlukannya peruntutan indikasi-indikasi menggunakan teknik klasifikasi untuk meminimalisir angka *stunting* tersebut. Penggunaan algoritma klasifikasi yaitu algoritma *K-Nearest Neighbor*, *Random Forest*, *Decision Tree C4.5* dan *Naïve Bayes* pernah digunakan untuk me*mining knowledge* serta mengolah dari dataset banyaknya penyakit. Keempat algoritma *data mining* tersebut mempunyai berbagai kelebihan dan kekurangan yang berbeda-beda. Akan tetapi, dari keempat algoritma belum dapat

dipastikan algoritma mana yang lebih akurat dan cepat dalam melakukan klasifikasi. Hal ini karena *dataset* yang digunakan dalam penelitian sebelumnya mengenai topik penyakit yang berbeda, tidak sama dan data diperlakuan secara berbeda. Tentu saja semakin banyak data, dan juga *noise* dalam data, tentunya semakin besar pengaruhnya terhadap performa algoritma pengklasifikasinya. Untuk mendapatkan algoritma yang efektif, maka peneliti akan membandingkan algoritma *K-Nearest Neighbor, Random Forest, Decision Tree C4.5* dan *Naïve Bayes* untuk diimplementasikan pada *dataset* potensi keluarga beresiko *stunting* Kecamatan Merbau.

1.3 Rumusan Masalah

Penelitian ini dilakukan berdasarkan rumusan masalah yang diperoleh dari latar belakang penelitian ini, antara lain:

1. Bagaimana menemukan model terbaik dalam melakukan klasifikasi potensi keluarga beresiko *stunting* yang dapat meminimalisir peningkatan angka *stunting* di Kecamatan Merbau?
2. Bagaimana hasil performa dari algoritma *K-Nearest Neighbor, Random Forest, Decision Tree C4.5* dan *Naïve Bayes* dalam melakukan pengklasifikasian potensi keluarga beresiko *stunting* di Kecamatan Merbau?

1.4 Batasan Masalah

Terdapat aspek-aspek bermasalah dalam penelitian ini, dan ini digunakan sebagai batasan. Keterbatasan penelitian ini, antara lain:

1. Data yang diambil untuk penelitian ini adalah data dari keluarga beresiko *stunting* Kecamatan Merbau tahun 2021.
2. Dalam melakukan klasifikasi potensi keluarga beresiko *stunting* di Kecamatan Merbau, peneliti menggunakan komparasi algoritma *machine learning* seperti algoritma *K-Nearest Neighbor, Random Forest, Decision Tree C4.5* dan *Naïve Bayes*.

3. Parameter atau variabel yang dipakai dalam penentuan klasifikasi potensi keluarga beresiko stunting terdiri dari :
    a. Baduta (0-23 bulan)
    b. Balita (24-59)
    c. PUS (Pasangan Usia Subur)
    d. PUS hamil
    e. Keluarga tidak mempunyai sumber air minum utama yang layak
    f. Keluarga tidak mempunyai jamban yang layak
    g. Terlalu muda (umur istri < 20 tahun)
    h. Terlalu tua (umur istri 35-40 tahun)
    i. Terlalu dekat (jarak anak < 2 tahun)
    j. Terlalu banyak (> 3 anak)
4. Bentuk klasifikasi yang akan di hasilkan adalah tingkat potensi keluarga beresiko stunting, yang terdiri dari, tidak beresiko, beresiko rendah, dan beresiko tinggi.
5. Output dari penelitian ini berupa performa dari setiap algoritma yang digunakan dalam klasifikasi potensi keluarga beresiko *stunting* di Kecamatan Merbau.

1.5 Tujuan Penelitian

Tujuan penelitian ini yaitu:

1. Melakukan klasifikasi potensi keluarga beresiko *stunting* yang dapat meminimalisir peningkatan angka *stunting* di Kecamatan Merbau.
2. Menemukan model terbaik dari algoritma *K-Nearest Neighbor, Random Forest, Decision Tree C4.5* dan *Naïve Bayes* dalam klasifikasi potensi keluarga beresiko *stunting* di Kecamatan Merbau.
3. Menemukan model terbaik dari algoritma *K-Nearest Neighbor, Random Forest, Decision Tree C4.5* dan *Naïve Bayes* dalam melakukan pengklasifikasian potensi keluarga beresiko *stunting* di Kecamatan Merbau.

1.6 Manfaat Penelitian

Adapun manfaat dari penelitian ini adalah sebagai berikut:

1. Dapat mengetahui performa algoritma *K-Nearest Neighbor, Random Forest, Decision Tree C4.5* dan *Naïve Bayes* pada klasifikasi potensi keluarga beresiko *stunting* di Kecamatan Merbau.
2. Dapat mengetahui kategori keluarga yang beresiko terkena *stunting* berdasarkan indikasi-indikasi yang ada.

BAB II
LANDASAN TEORI

2.1 Tinjauan Pustaka

Terdapat penelitian terdahulu untuk menganalisis tingkat kesulitan soal yang dilakukan oleh (Prasetiya et al., 2020) yang meneliti kasus *stunting* bayi lima tahun menggunakan algoritma *K-Nearest Neighbor* di Desa Slangit. Penelitian tersebut memanfaatkan perkiraan jarak Euclidean, dimana metode tersebut untuk mengklasifikasikan *dataset* dari data *training* pada beberapa *neighbor* paling dekat dengan menggunakan rumus perhitungan jarak Euclidean. Penelitian tersebut masih menggunakan perhitungan manual. Terdapat kelemahan dengan menggunakan perhitungan manual yaitu jika data yang diklasifikasikan dalam jumlah banyak, hal tersebut akan memakan banyak waktu. Maka dari itu, penelitian ini menggunakan perhitungan otomatis dengan metode Algoritma *K-Nearest Neighbor, Random Forest, Decision Tree C4.5* dan *Naïve Bayes*. Namun ada juga penelitian yang membandingkan atau mengkomparasikan metode algoritma klasifikasi *data mining* dengan menggunakan *dataset* dan atribut yang berbeda. Studi terkait sebelumnya tercantum dalam Tabel 2.1.

Tabel 2. 1 Penelitian Sebelumnya yang Relevan

No.	Judul	Hasil	Peneliti
1.	Perbandingan Kinerja Algoritma *Naïve Bayes* serta *K-Nearest Neighbor* untuk Klasifikasi Artikel Bahasa	Dari hasil yang diperoleh bahwa algoritma *Naïve Bayes* menghasilkan kinerja bagus dengan tingkat akurasi 70%, sedangkan algoritma *K-Nearest Neighbor* menghasilkan akurasi yang kurang bagus yaitu 40%.	(Devita et al., 2018)

No.	Judul	Hasil	Peneliti
	Indonesia		
2.	Analisis Komparasi Algoritma Klasifikasi *Data Mining* untuk Prediksi Penderita Penyakit Jantung	Penelitian tersebut menunjukkan bahwa *accuracy* algoritma K-NN adalah yang kurang bagus diterapkan dalam *dataset*, algoritma *random forest* 80.38% serta *decision stump* menunjukkan performa yang paling baik dengan *accuracy* 78.95%, C4.5 dan *Naïve Bayes* juga tampil bagus.	(Annisa, 2019)
3.	Perbandingan Kinerja Algoritma Klasifikasi *Naïve Bayes, Support Vector Machine (SVM),* dan *Random Forest* untuk Prediksi Ketidakhadiran di *Workplace*	Penelitian ini menunjukkan hasil bahwa algoritma *Random Forest* memperoleh nilai akurasi yang paling tinggi dibandingan dengan algoritma *Naïve Bayes* dan SVM, yang menampilkan nilai *accuracy* sebesar 99.38%, *recall* 99.39%, dan *precision* 99.42%.	(Nalatissifa et al., 2021)
4.	Komparasi Algoritma KNN serta Algoritma *Naïve Bayes* pada Klasifikasi Diagnosis	Hasil penelitian ini menghasilkan algoritma *Naïve Bayes* cukup bagus dibandingkan algoritma KNN, dengan *accuracy* 80%	(Marito Putry & Nurina Sari, 2022)

		Penyakit Diabetes Melitus	
5.	Perbandingan Implementasi Algoritma *Naïve Bayes* serta Pada Klasifikasi Penyakit Hati	Penelitian tersebut menunjukkan hasil bahwa algoritma K-NN memberikan nilai *accuracy, recall,* serta *precision* yang lebih bagus dibandingkan dengan algoritma *Naïve Bayes*, dengan angka 100%.	(Desiani, 2022)

2.2 Dasar Teori

2.2.1 *Stunting*

Stunting merupakan suatu keadaan ketidakberhasilan pertumbuhan pada anak usia dibawah 5 tahun yang disebabkan oleh kekurangan gizi, yang mengakibatkan anak tumbuh terlalu pendek untuk usianya. Kekurangan gizi bisa saja terjadi sejak anak masih ada dalam kandungan dan pada hari-hari pertama setelah anak lahir, namun tidak tampak, akan terlihat jika anak sudah berusia dua tahun, dimana keadaan gizi anak serta ibu adalah hal terpenting untuk perkembangan anak (Rahayu et al., 2018). Pada tahun 2017, *stunting* terjadi pada lebih dari setengah bayi usia lima tahun di dunia yaitu 55% dari wilayah Asia, sedangkan 39% berasal dari Afrika. Dari 83,6 juta bayi usia lima tahun *stunting* di Asia, 58,7% merupakan jumlah terbanyak yang berasal dari kawasan Asia Selatan dan jumlah 0,9% paling sedikit berasal dari kawasan Asia Tengah (Atmarita & Zahrani, 2018).

Menurut (Boucot & Poinar Jr., 2010) faktor yang dapat menghambat *stunting* yaitu dilakukan dengan cara (1) ASI ekslusif hingga usia enam bulan dan juga setelah umur enam bulan diberikan makanan pendamping ASI (MPASI) yang memadai jumlahnya dan mutunya, (2) Memperhatikan perkembangan bayi usia lima tahun di POSYANDU, (3) Meningkatkan akses fasilitas sanitasi dan air bersih, serta menjaga kebersihan lingkungan, (4) Penyediaan kebutuhan zat gizi bagi ibu hamil yang cukup.

2.2.2 Keluarga Beresiko *Stunting*

Keluarga berisiko stunting adalah keluarga yang mempunyai satu atau lebih faktor risiko terjadinya stunting (BKKBN, 2021). Parameter keluarga beresiko stunting yang digunakan dalam penelitian ini adalah :

a. Baduta (0-23 bulan),
b. Balita (24-59 bulan)
c. PUS (Pasangan usia subur)

   Yang berusia 10-49 tahun
d. PUS hamil
e. Keluarga tidak mempunyai sumber air minum utama yang layak

   Kriteria air minum utama yang layak :

   - Air kemasan / isi ulang
   - Ledeng / PAM
   - Sumur bor
   - Sumur terlindungi

   Kriteria air minum utama yang tidak layak :

   - Sumur tidak terlindungi
   - Air permukaan (sungai, danau, dll)
   - Air hujan
   - Lainnya

f. Keluarga tidak mempunyai jamban yang layak

Kriteria jamban yang layak :

- Ya, dengan septic tank

Kriteria jamban yang tidak layak :

- Ya, tanpa septic tank
- Tidak, jamban umum / bersama
- Lainnya

g. Terlalu muda (umur istri < 20 tahun)

h. Terlalu tua (umur istri 35-40 tahun)

i. Terlalu dekat jarak anak (< 2 tahun)

j. Terlalu banyak (> 3 anak).

2.2.3 *Data Mining*

Data mining adalah proses mengidentifikasi pola dan wawasan data yang menarik dari kumpulan data yang sangat besar (Muslim et al., 2019). Menurut (Asha Kiranmai & Jaya Laxmi, 2018) Data mining adalah proses yang menggunakan berbagai alat analisis data untuk mengidentifikasi pola dan hubungan tersembunyi dalam data. Menurut (Duan & Gao, 2021) data mining adalah proses penggalian ekstrasi informasi yang berpotensi berguna dan pengetahuan dari sejumlah besar data yang tidak lengkap, samar, dan acak. Sedangkan menurut (Mardi Yuli, 2016) data mining merupakan sebuah metode untuk pencarian pola terhadap data yang terpilih dengan metode maupun cara khusus. Teknik *data mining* diolah untuk mengontrol *database* berukuran besar dengan cara menemukan suatu pola yang berbeda dan bermanfaat. Akan tetapi, tidak semua pencarian data atau bahan dapat diputuskan sebagai *data mining*. Metode pada data mining sangat beragam, penentuan metode tersebut berpusat dalam sasaran serta metode *Knowledge Discovery in Database*

(KDD) secara menyeluruh. Menurut (Muslim et al., 2019) *data mining* dikelompokkan menjadi:

1. *Classification*

   Classification suatu teknik untuk menemukan model agar menjelaskan atau membedakan konsep atau kelas dalam data, dengan tujuan untuk dapat menyimpulkan kelas objek yang namanya tidak diketahui.

2. *Clustering*

   Clustering adalah teknik pengelompokan data secara otomatis tanpa menentukan label kelasnya.

3. *Association Rule*

   Association Rule adalah teknik *data mining* yang berfokus pada pencarian aturan kesamaan dalam suatu peristiwa.

4. *Regression*

   Teknik *Regression* tidak jauh berbeda dengan teknik klasifikasi. Bedanya, teknik *regression* tidak dapat mencari pola yang dijelaskan pada tabel. Gunakan teknik *regression* untuk mencari pola dan menentukan angka.

Proses yang akan dilakukan dalam *data mining* :

1. *Data Selection*

   Pada tahap *data selection* dilakukan pemilihan data yang akan digunakan dalam proses *data mining*. Dataset yang telah disiapkan dilakukan proses pemilihan atribut dari masing-masing data

2. *Preprocessing/Cleaning*

Pada tahap ini dilakukan *preprocessing* dan *cleaning* agar data terhindar dari duplikasi dan tidak konsistennya data, serta memperbaiki kesalahan pada data atau menambahkan data untuk proses klasifikasi. Preprocessing yang digunakan yaitu label encoder. Label encoder berfungsi untuk mengubah data kategorikal menjadi numerikal (Latifah et al., 2019).

3. Transformation

Tahap *transformation* merupakan proses mengubah data dari bentuk asalnya menjadi data yang siap untuk proses *data mining*. Pada tahap ini dilakukan konversi bentuk data yang belum mempunyai entitas dengan jelas pada bentuk data yang valid atau siap untuk dilakukan proses *data mining*.

2.2.4 Klasifikasi

Klasifikasi merupakan teknik yang digunakan untuk menemukan model agar dapat menjelaskan atau membedakan konsep atau kelas data, dengan tujuan untuk dapat memperkirakan kelas dari suatu objek yang labelnya tidak diketahui (Muslim et al., 2019). Menurut (Widiastuti et al., 2017) Klasifikasi adalah proses pengelompokkan objek yang memiliki karakteristik atau ciri yang sama ke dalam beberapa kelas. Masing-masing cara klasifikasi yang memanfaatkan suatu algoritma untuk mengetahui suatu model yang melengkapi *requirement* antara atribut dan label *class* pada *input data*. *Input data* dari model *classification* merupakan sekumpulan *record.* Masing-masing *record* merupakan himpunan dari atribut yang merupakan *class.* Model atribut *class* yakni sebuah fungsi yang berasal dari nilai atribut lainnya. Suatu *data test* digunakan untuk menentukan akurasi model yang dibagi menjadi *data test* serta *data training, data training* digunakan untuk membangun model sedangkan *data test* digunakan untuk memvalidasi (Purnamawati et al., 2020).

Proses klasifikasi didasarkan pada empat komponen yaitu (1) kelas, variabel dependen yang berupa kategorikal yang merepresentasikan ‘label’ yang terdapat pada objek, (2) predictor merupakan variabel independent yang direpresentasikan oleh karakteristik (atribut) data, (3) *training dataset* merupakan satu set data yang berisi nilai dari kedua komponen di atas yang digunakan untuk menentukan kelas yang cocok berdasarkan *predictor*, (4) *testing dataset* yakni berisi data baru yang akan diklasifikasikan oleh model yang telah dibuat dan akurasi klasifikasi dievaluasi (A Damayanti, 2016). Terdapat beberapa algoritma klasifikasi yaitu sebagai berikut:

a. Algoritma *K-Nearest Neighbor*

Algoritma KNN dapat didefinisikan berdasarkan namanya dimana K melambangkan sejumlah nilai tetangga terdekat (Nearest Neighbors) yang diguna untuk mengidentifikasi kemiripan sebuah titik baru dengan titik tetangganya (Id, 2021). K-Nearest Neighbor (KNN) adalah algoritma yang memberikan hasil yang mudah diinterpretasikan dan serbaguna serta dapat digunakan untuk klasifikasi dan regresi (Hatem, 2022). *K-Nearest Neighbor* (KNN) adalah sebuah algoritma yang digunakan untuk mengatasi permasalahan pada klasifikasi. Algoritma KNN telah banyak digunakan dalam pengenalan pola karena kesederhanaannya dan keberhasilan empirisnya yang baik (Hu et al., 2018).

Algoritma KNN dapat digunakan untuk mengelompokkan atau mengklasifikasikan suatu data baru yang tidak diketahui kelasnya berdasarkan jarak data baru itu ke beberapa *neighbor* terdekat. Jumlah *neighbor* terdekat representasikan dengan k. Nilai k yang baik tergantung pada data. Pendekatan yang digunakan untuk menentukan nilai k yaitu : k = $\sqrt{n}$, dimana n adalah jumlah dari sampel data yang tersedia. Apabila terjadi dua atau lebih jumlah kelas yang muncul sama maka nilai k menjadi k – 1, jika masih ada yang sama lagi maka nilai k menjadi k – 2, begitu seterusnya hingga tidak ditemukan *class* yang sama banyak. Jauh dan dekatnya tetangga *(neighbor)* dapat dihitung berdasarkan jarak Euclidean (*Euclidean Distance*). Persamaan (2.1) ini merupakan rumus pencarian jarak menggunakan rumus Euclidean

$$d_i = \sqrt{\sum_{i=1}^{p}(x_{2i} - x_{1i})^2} \quad \textbf{(2.1)}$$

Keterangan:

x1 = sampel data

x2 = data uji

i = variabel data

dist = jarak

p = dimensi data

Contoh soal :

Diberikan data training berupa dua atribut Baik dan Buruk untuk mengklasifikasikan sebuah data apakah tergolong baik atau buruk, berikut ini adalah contoh datanya :

Tabel 2. 2 Contoh Soal *K-Nearest Neighbor*

X	Y	Kategori
7	6	Buruk
6	6	Buruk
6	5	Buruk
1	3	Baik
2	4	Baik
2	2	Baik
3	5	?

Lalu diberikan data baru yang akan kita klasifikasikan, yaitu X=3 dan Y=5. Jadi termasuk klasifikasi apa data baru? Baik atau Buruk?

Langkah penyelesaian :

Pertama, kita tentukan parameter K. Misalnya kita buat jumlah tertangga terdekat K=3.

Ke-dua, kita hitung jarak antara data baru dengan semua data training. Kita menggunakan Euclidean distance. Kita hitung seperti pada table berikut :

X	Y	Euclidean Distance (3,5)
7	6	$\sqrt{(7-3)^2+(6-5)^2} = \sqrt{(4)^2+(1)^2} = \sqrt{17} = 4.12$
6	6	$\sqrt{(6-3)^2+(6-5)^2} = \sqrt{(3)^2+(1)^2} = \sqrt{10} = 3.16$
6	5	$\sqrt{(6-3)^2+(5-5)^2} = \sqrt{(3)^2+(0)^2} = \sqrt{9} = 3$
1	3	$\sqrt{(1-3)^2+(3-5)^2} = \sqrt{(-2)^2+(-2)^2} = \sqrt{8} = 2.82$
2	4	$\sqrt{(2-3)^2+(4-5)^2} = \sqrt{(-1)^2+(-1)^2} = \sqrt{2} = 1.41$
2	2	$\sqrt{(2-3)^2+(2-5)^2} = \sqrt{(-1)^2+(-3)^2} = \sqrt{10} = 3.16$

X	Y	Euclidean Distance (3,5)	Ukuran jarak	Apakah termasuk 3-NN
7	6	4.12	5	Tidak (K>3)
6	6	3.16	4	Tidak (K>3)
6	5	3	3	Ya (K=3)
1	3	2.82	2	Ya (K<3)
2	4	1.41	1	Ya (K<3)
2	2	3.16	4	Tidak (K>3)

Dari kolom 4 (urutan jarak) kita mengurutkan dari yang terdekat ke terjauh antara jarak data baru dengan data training. Ada 2 jarak yang sama (yaitu 4) pada data baris 2 dan baris 6, sehingga memiliki urutan yang sama. Pada kolom 5 (Apakah termasuk 3-NN). Maksudnya adalah K-NN menjadi 3-NN, karena nilai K ditentukan sama dengan 3.

Ke-empat, tentukan kategori dari tetangga terdekat. Kita perhatikan baris 3, 4, dan 5 pada gambar sebelumnya (diatas). Kategori Ya diambil jika nilai K<=3. Jadi baris 3, 4, dan 5 termasuk kategori Ya dan sisanya Tidak.

X	Y	Euclidean Distance (3,5)	Ukuran jarak	Apakah termasuk 3-NN	Kategori Ya untuk KNN
7	6	4.12	5	Tidak (K>3)	-
6	6	3.16	4	Tidak (K>3)	-
6	5	3	3	Ya (K=3)	Buruk
1	3	2.82	2	Ya (K<3)	Baik
2	4	1.41	1	Ya (K<3)	Baik
2	2	3.16	4	Tidak (K>3)	-

Kategori ya untuk K-NN pada kolom 6, mencakup baris 3,4, dan 5. Kita berikan kategori berdasarkan tabel awal. baris 3 memiliki kategori buruk, dan 4,5 memiliki kategori baik.

Ke-lima, gunakan kategori mayoritas yang sederhana dari tetangga yang terdekat tersebut sebagai nilai prediksi data yang baru.

X	Y	Kategori
7	6	Buruk
6	6	Buruk
6	5	Buruk
1	3	Baik
2	4	Baik
2	2	Baik
3	5	Baik

Data yang kita miliki pada baris 3, 4 dan 5 kita punya 2 kategori baik dan 1 kategori buruk. Dari jumlah mayoritas (baik > buruk) tersebut kita simpulkan bahwa data baru (X=3 dan Y=5) termasuk dalam kategori baik.

b. Algoritma *Random Forest*

Random Forest membangun beberapa pohon keputusan dengan set pelatihan bootstrap dan menggabungkan prediksinya untuk membuat keputusan akhir (Schumacher & Jr, 2023). Random Forest adalah metode pengklasifikasi ansambel, semua pohon keputusan berpartisipasi dalam pemungutan suara, beberapa pohon keputusan berkualitas rendah akan mengurangi keakuratan hutan acak (Zhang et al., 2021). Metode *Random Forest* adalah perluasan dari metode CART, dengan mengimplementasikan metode *bootstrap aggregating (bagging)* serta *random feature selection*. *Random forest* yang didapatkan menampilkan *tree* yang mencapai ratusan, dan masing-masing *tree* ditanam dengan cara yang sama. Beberapa fungsi pembelajaran yang dihasilkan oleh *Random Forest* digunakan untuk strategi *ensemble bagging* dalam mengatasi permasalahan *overfitting* jika terdapat data *train* yang kecil (Samudra, 2019). Pada pohon keputusan *Random Forest* menggunakan metode CART, dimulai dengan cara menghitung nilai *entropy* untuk menentukan tingkat ketidakaslian atribut serta nilai *information gain*. Persamaan di bawah ini merupakan rumus untuk menghitung nilai *entropy*.

$$Entropy\ (Y) = -\sum_{i} p(c|Y) log2\ p(c|Y) \quad \textbf{(2.2)}$$

Keterangan:

Y = himpunan kasus

p(c|Y) = proporsi nilai Y terhadap kelas c

Untuk mengetahui *information gain* dalam mengukur efektivitas suatu atribut dalam mengklasifikasikan data dapat dihitung menggunakan rumus (2.3) sebagai berikut:

$$Information\ Gain\ (Y, a) = Entropy\ (Y) - \sum values(a) \frac{|Yv|}{|Ya|} Entropy\ (Yv) \quad \textbf{(2.3)}$$

Keterangan:

Value(a) = merupakan semua nilai yang mungkin dalam himpunan kelas a

Yv = subkelas dari Y dengan kelas v yang berhubungan dengan kelas a

Ya = semua nilai yang sesuai dengan a

Untuk mencari *split-point* yang paling baik maka data pada atribut terlebih dahulu diurutkan. Nilai tengah antara pasangan nilai yang berdekatan akan dianggap sebagai kemungkinan yang bisa dijadikan *split-point.*

Contoh Soal :

Cari atribut terbaik dari pohon 1

Tabel 2. 3 Contoh Soal *Random Forest*

id	Temperatur badan	Sesak nafas	Batuk	Diagnosis sakit covid-19
1	Tinggi	Ya	Tidak	Ya
2	Normal	Ya	Ya	Ya
3	Normal	Tidak	Tidak	Tidak
4	Tinggi	Tidak	Ya	Tidak
5	Normal	Ya	Tidak	Ya
6	Normal	Ya	Tidak	Ya
7	Tinggi	Ya	Ya	Ya
8	Normal	Tidak	Ya	Tidak

Langkah penyelesaian :

Iterasi pohon 1

Kita asumsikan bahwa pohon yang akan dibuat sebanyak 3 pohon. Karena dataset memiliki tupel yang sedikit, maka pohon pertama bisa kita buat dengan melibatkan seluruh atribut dan hilangkan baris nomor 5. Pada kasus ini, pencarian root atau atribut terbaik akan dilakukan dengan menggunakan gini indeks, tidak dengan information gain atau gain rasio. Tujuan dari contoh soal ini adalah mencari atribut terbaik dari berbagai pohon yang dibangun (best split).

id	Temperatur badan	Sesak nafas	Batuk	Diagnosis sakit covid-19
1	Tinggi	Ya	Tidak	Ya
2	Normal	Ya	Ya	Ya
3	Normal	Tidak	Tidak	Tidak
4	Tinggi	Tidak	Ya	Tidak
6	Normal	Ya	Tidak	Ya
7	Tinggi	Ya	Ya	Ya
8	Normal	Tidak	Ya	Tidak

Mengubah menjadi tabel kontingensi

Temperatur badan	Diagnosis		
	Ya	Tidak	Total
Tinggi	2	1	3
Normal	2	2	4
Total	4	3	7
Sesak Nafas	Diagnosis		
	Ya	Tidak	Total
Ya	4	0	4
Tidak	0	3	3
Total	4	3	7
Batuk	Diagnosis		
	Ya	Tidak	Total
Ya	2	2	4
Tidak	2	1	3
Total	4	3	7
	Diagnosis		
	Ya	Tidak	Total
	4	3	7

Selanjutnya, hitung gini indeks dari setiap kelas dari atribut yang tersedia dengan rumus:

$$Gini = 1 - \sum_{i=1}^{c} (\text{pi})^2$$

Maka untuk setiap atribut, kita bias mendapatkan gini indeks :

Temperatur badan | Tinggi = $1 - \left(\frac{2}{3}\right)^2 + \left(\frac{1}{3}\right)^2 = 1 - \frac{4}{9} + \frac{1}{9} = 1 - \frac{5}{9}$

$= \frac{9}{9} - \frac{5}{9} = 0{,}44$

Temperatur badan | Normal = $1 - \left(\frac{2}{4}\right)^2 + \left(\frac{2}{4}\right)^2 = 0{,}5$

Sesak nafas | Ya = $1 - \left(\frac{4}{4}\right)^2 + \left(\frac{0}{4}\right)^2 = 0$

Sesak nafas | Tidak = $1 - \left(\frac{0}{3}\right)^2 + \left(\frac{3}{3}\right)^2 = 0$

Batuk | Ya = $1 - \left(\frac{2}{4}\right)^2 + \left(\frac{2}{4}\right)^2 = 0{,}5$ 8

Batuk | Tidak = $1 - \left(\frac{2}{3}\right)^2 + \left(\frac{1}{3}\right)^2 = 0.44$

Setelah didapatkan semua gini indeks, selanjutnya dicari gini split sebagai berikut :

Temperatur badan = $\left(\frac{3}{7}\right) * 0.444444444 + \left(\frac{4}{7}\right) * 0.5 = 0.476190476$

Sesak nafas = $\left(\frac{5}{8}\right) * 0 + \left(\frac{3}{8}\right) * 0 = 0$

Batuk = $\left(\frac{4}{7}\right) * 0.5 + \left(\frac{4}{7}\right) * 0.444444444 = 0.476190476$

Terakhir adalah penentuan akar atau atribut terbaik. Berdasarkan perhitungan di atas, maka atribut sesak nafas merupakan atribut terbaik karena memiliki gini split indeks yang paling rendah.

c. Algoritma *Decision Tree* C4.5

Algoritma C4.5 diperkenalkan oleh Quinlan sebagai versi perbaikan dari ID3. Dalam ID3, induksi decision tree hanya bisa dilakukan pada fitur bertipe kategorikal (nominal/ordinal), sedangkan tipe numerik (internal/rasio) tidak dapat digunakan. Perbaikan yang membedakan algoritma C4.5 dari ID3 adalah dapat menangani fitur dengan tipe numerik, melakukan pemotongan (pruning) decision tree, dan penurunan (deriving) rule set (Muslim et al., 2019). Algoritma Decision Tree C4.5 merupakan algoritma yang dapat digunakan untuk membentuk pohon keputusan. Pohon keputusan (Decision Tree) adalah salah satu metode yang cukup mudah untuk di interpretasikan oleh manusia (Nasrullah, 2021). Salah satu teknik efisien yang banyak digunakan di berbagai bidang, seperti pembelajaran mesin, pemrosesan gambar, dan pengenalan pola adalah Decision Tree. Selain itu, algoritma ini juga dapat diterapkan pada klasifikasi secara langsung, yang secara efisien mengurangi kompleksitas algoritma (M. Li et al., 2019). Algoritma *decision tree* C4.5 lebih baik dibandingkan ID3 dan CART karena tidak dapat membatasi cabang format biner (Supangat et al., 2018). Dikutip dari (Enri, 2018) langkah-langkah pembentukan algoritma *decision tree* C4.5 yakni sebagai berikut:

1. Memilih atribut sebagai *root*, dengan cara mengkalkulasi nilai *gain* dari semua atribut, nilai *gain* tertinggi akan dijadikan *root*. Sebelum mengkalkulasi nilai *gain* harus mengklakulasi *entrophy*.
2. Membuat cabang pada setiap nilai.
3. Membagi kasus pada cabang.
4. Mengulangi proses pada setiap cabang sampai hingga masing-masing kasus cabang ada kelas yang sama.

Dalam menentukan atribut *root*, dilakukan pengkalkulasian nilai *gain* untuk setiap atribut rumus pada persamaan 2.4 berikut:

$$Gain\ (S, A) = Entropy\ (s) - \sum_{i=1}^{n} \frac{|Si|}{|S|} * Entropy\ (Si) \qquad \textbf{(2.4)}$$

Keterangan:

{S1, S2, S3, …, Si, …, Sn} = partisi S sejumlah nilai atribut A

n = jumlah atribut A

|Si| = jumlah kasus dalam partisi Si

|S| = total kasus S

Sementara untuk mendapatkan nilai *entropy* menggunakan rumus 2.5 sebagai berikut:

$$Entropy\ (s) = \sum_{i=1}^{n} - Pi * \log 2\ Pi \qquad \textbf{(2.5)}$$

Dimana:

S = jumlah kasus

N = jumlah kasus dalam partisi S

Pi = proporsi Si terhadap S

Contoh soal :

Dalam kasus berikut, pohon keputusan akan dibuat untuk memutuskan apakah akan bermain tenis tergantung pada kondisi cuaca, suhu, kelembapan, dan kondisi angin.

Tabel 2. 4 Contoh Soal *Decision Tree*

No	Kondisi	Suhu	Kelembaban	Berangin	Main
1	Cerah	Panas	Tinggi	Salah	Tidak
2	Cerah	Panas	Tinggi	True	Tidak
3	Berawan	Panas	Tinggi	Salah	Iya
4	Hujan	Sedang	Tinggi	Salah	Iya

5	Hujan	Dingin	Normal	Salah	Iya
6	Hujan	Dingin	Normal	True	Iya
7	Berawan	Dingin	Normal	True	Iya
8	Cerah	Sedang	Tinggi	Salah	Tidak
9	Cerah	Dingin	Normal	Salah	Iya
10	Hujan	Sedang	Normal	Salah	Iya
11	Cerah	Sedang	Normal	True	Iya
12	Berawan	Sedang	Tinggi	True	Iya
13	Berawan	Panas	Normal	Salah	Iya
14	Hujan	Sedang	Tinggi	True	Tidak

Secara umum, algoritma C4.5 untuk membangun pohon keputusan adalah sebagai berikut :

- Pilih atribut sebagai akar
- Buat cabang untuk masing-masing nilai
- Bagi kasus dalam cabang
- Ulangi proses untuk masing-masing cabang sampai semua kasus pada cabang memiliki kelas yang sama

Untuk memilih atrubut sebagai akar, didasarkan pada nilai gain tertinggi dari atribut-atribut yang ada. Untuk menghitung gain digunakan rumus seperti tertera dalam rumus 2.4, sedangkan nilai entropy dapat dilihat pada rumus 2.5.

Berikut ini adalah penjelasan lebih rinci mengenai masing-masing langkah dalam pembentukan pohon keputusan dengan menggunakan algoritma C4.5 untuk menyelesaikan permasalahan pada tabel 2.4 :

- Menghitung jumlah kasus, jumlah kasus untuk keputusan Iya, jumlah kasus untuk keputusan Tidak, dan Entropy dari semua kasus dan kasus yang dibagi berdasarkan atribut Kondisi, Temperatur, Kelembaban, dan Berangin. Setelah itu lakukan perhitungan Gain untuk masing-masing atribut. Hasil perhitungan ditunjukkan pada tabel 2.5

Tabel 2. 5 Perhitungan Node 1

Node			Jumlah kasus (S)	Tidak (S_1)	Ya (S_2)	Entropy	Gain
1	Total		14	4	10	0.863120569	
	Kondisi						0.258521037
		Berawan	4	0	4	0	
		Hujan	5	1	4	0.721928095	
		Cerah	5	3	2	0.970950594	
	Suhu						0.183850925
		Dingin	4	0	4	0	
		Panas	4	2	2	1	
		Sedang	6	2	4	0.918295834	
	Kelembaban						0.370506501
		Tinggi	7	4	3	0.985228136	
		Normal	7	0	7	0	
	Berangin						0.005977711
		Salah	8	2	6		
		True	6	4	2		

Baris Total kolom Entropy pada tabel 2.5, dihitung dengan rumus 2.5, sebagai berikut :

$$\textit{Entropy (Total)} = \left(-\frac{4}{14} * log2\ \left(\frac{4}{14}\right)\right) + \left(-\frac{10}{14} * log2\ \left(\frac{10}{14}\right)\right)$$

$$Entropy\ (Total) = 0.863120569$$

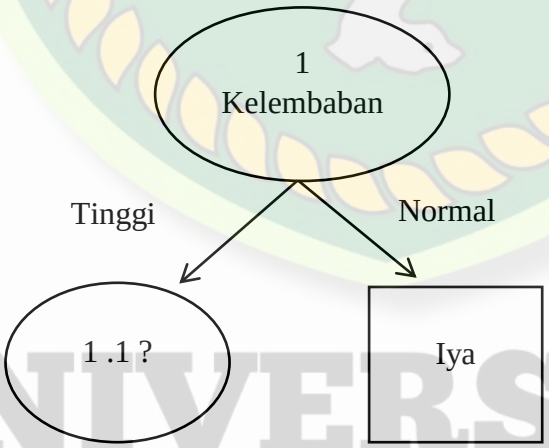
Sementara itu, nilai Gain pada baris Kondisi dihitung dengan menggunakan rumus 2.4, sebagai berikut :

$$Gain\ (Total, Outlook) = Entropy\ (Total) - \sum_{i=1}^{n} \frac{|\ Outlook_{\ i}|}{|Total|} * Entropy\ (Outlook_{\ i})$$

$$\textit{Gain (Total, Kondisi)} = 0.863120569\ - \left(\left(\frac{4}{14} * 0\right) + \left(\frac{5}{14} * 0.723\right)\left(\frac{5}{14}\ * 0.97\right)\right)$$

Gain (Total, Kondisi) = 0.23

Dari hasil pada tabel 2.5. terlihat bahwa atribut dengan gain tertinggi adalah Kelembaban yaitu sebesar 0.37. Oleh karena itu, kelembaban dapat menjadi node akar. Ada 2 nilai atribut dari Kelembaban yaitu Tinggi dan Normal. Dari dua nilai atribut tersebut, nilai atribut Normal sudah mengklasifikasikan kasus menjadi 1 yaitu, keputusan nya adalah Iya, sehingga tidak diperlukan perhitungan lebih lanjut, namun nilai atribut Tinggi masih perlu dilakukan perhitungan lagi. Dari hasil tersebut, kita dapat membuat pohon keputusan sementara seperti yang ditunjukkan pada gambar 2.1



Gambar 2. 1 Pohon Keputusan Hasil Perhitungan Node 1

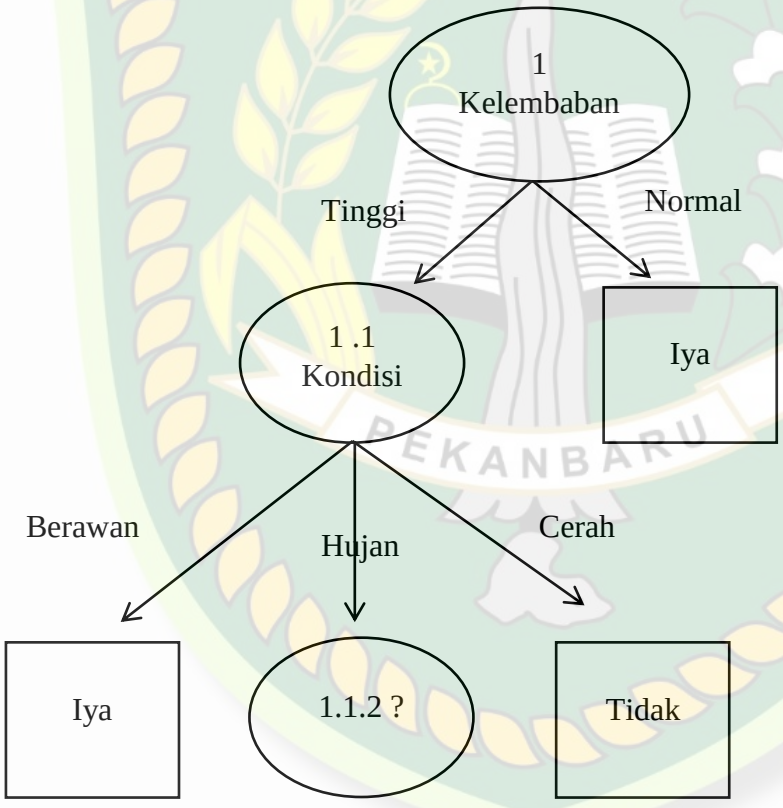
Menghitung jumlah kasus, jumlah kasus dengan keputusan Iya, jumlah kasus dengan keputusan Tidak, dan Entropy semua kasus dan pembagian kasus berdasarkan atribut Kondisi, Suhu dan Berangin yang dapat menjadi node akar dari nilai atribut Tinggi. Selanjutnya lakukan penghitungan Gain untuk masing-masing atribut. Hasil perhitungannya ditunjukkan pada Tabel 2.6.

Tabel 2. 6 Perhitungan Node 1.1

Node			Jumlah kasus (S)	Tidak (S_1)	Ya (S_2)	Entropy	Gain
1.1	Kelembaban – Tinggi		7	4	3	0.985228136	
	Kondisi						0.69951385
		Berawan	2	0	2	0	
		Hujan	2	1	1	1	
		Cerah	3	3	0	0	
	Suhu						0.020244207
		Dingin	0	0	0	0	
		Panas	3	2	1	0.918295843	
		Sedang	4	2	2	1	
	Berangin						0.020244207
		Salah	4	2	2	1	
		True	3	2	1	0.918295834	

Hasil pada tabel 2.6 menunjukkan bahwa atribut dengan Gain tertinggi adalah Kondisi yaitu sebesar 0.69. Hal ini memungkinkan Kondisi menjadi node cabang dari nilai atribut Tinggi Ada 3 nilai atribut dari Kondisi yaitu Cloud, Hujan dan Cerah. Di antara ketiga nilai atribut tersebut, nilai atribut Berawan sudah mengklasifikasikan kasus sebagai 1 yang merupakan penilaian Yes dan nilai atribut Cerah mengklasifikasikan kasus menjadi satu dengan keputusan No, sehingga perhitungan lebih lanjut tidak dilakukan, kecuali untuk nilai atribut Hujan masih diperlukan perhitungan lebih lanjut.

Pohon keputusan yang terbentuk sampai tahap ini ditunjukkan pada gambar 2.2 berikut:



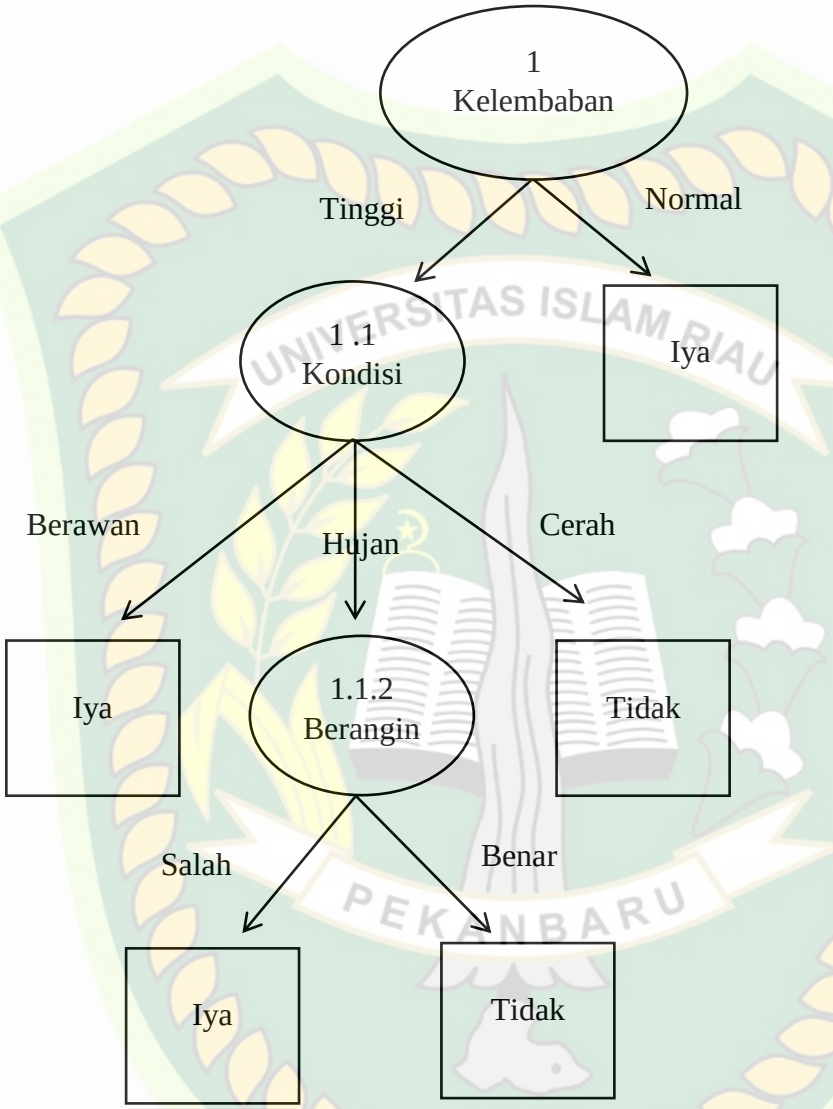
Gambar 2. 2 Pohon Keputusan Hasil Perhitungan Node 1.1

Hitung jumlah kasus, jumlah kasus dengan keputusan Yes, jumlah kasus dengan keputusan No, dan Entropy dari semua kasus dan kasus yang dibagi berdasarkan atribut Suhu dan Berangin yang dapat menjadi node cabang dari nilai atribut Hujan. Setelah menghitung Gain untuk setiap atribut. Perhitungannya ditunjukkan pada tabel 2.7

Dari hasil tabel 2.7 terlihat bahwa atribut dengan Gain tertinggi adalah Berangin yaitu sebesar 1. Oleh karena itu, Berangin dapat menjadi node cabang untuk nilai atribut Hujan. Berangin memiliki dua nilai atribut, yaitu Salah dan True. Dari kedua nilai atribut tersebut, nilai atribut Salah mengklasifikasikan kasus sebagai 1 yaitu keputusan Yes dan nilai atribut True mengklasifikasikan sebagai keputusan No, sehingga tidak diperlukan perhitungan lebih lanjut untuk nilai atribut ini.

Tabel 2. 7 Perhitungan Node 1.1.2

Node			Jumlah Kasus (S)	Tidak (S_1)	Ya (S_2)	Entropy	Gain
1.1.2	Kelembaban – Tinggi dan Kondisi - Hujan		2	1	1	1	
	Suhu						0
		Dingin	0	0	0	0	
		Panas	0	0	0	0	
		Sedang	2	1	1	1	
	Berangin						1
		Salah	1	0	1	0	
		True	1	1	0	0	



Gambar 2. 3 Pohon Keputusan Hasil Perhitungan Node 1.1.2

Dengan memperhatikan pohon keputusan pada gambar 2.3, diketahui bahwa semua kasus sudah masuk dalam kelas. Dengan demikian, pohon keputusan pada gambar 2.3 merupakan pohon keputusan terakhir yang terbentuk

d. Algoritma *Naïve Bayes*

Bayes merupakan teknik prediksi berbasis probalistik sederhana yang berdasar pada penerapan teorema bayes atau aturan bayes dengan asumsi independensi (ketidaktergantungan) yang kuat (*Naïve*) (Muslim et al., 2019). Menurut (Syarli & Muin, 2016) Algoritma *Naïve bayes* adalah algoritma pembelajaran paling efisien dan efektif untuk *machine learning* dan *data mining*. Model Naïve Bayes digunakan untuk klasifikasi karena sederhana, efisien, dan mudah untuk di mengerti (Ruan et al., 2020). Pengklasifikasi bergantung pada penerapan teorema Bayes, *Naïve bayes* menganggap setiap variabel atribut sebagai variabel independen (Kamel et al., 2019). Teorema keputusan *Bayes* merupakan sebuah pendekatan statistik yang mendasar dengan mengenali pola *(pattern recognition)*. *Naïve bayes* berpusat dalam pemikiran penyederhanaan dimana nilai atribut saling bebas apabila diberikan nilai *output* atau, kemungkinan mengamati secara bersama merupakan hasil dari probabilitas individu (Ridwan et al., 2013). $\left(\frac{5}{14} * 0.723\right)$

Secara garis besar algoritma *Naïve Bayes* dapat dijelaskan seperti persamaan 2.6 berikut:

$$P(S|R) = \frac{P(R)P(S|R)}{P(S)} \quad \textbf{(2.6)}$$

Keterangan:

R = Data yang belum diketahui kelasnya

S = Hipotesis pada data R yang merupakan *class* khusus

P(R|S) = Nilai probabilitas pada hipotesis R yang berdasarkan kondisi S

P(R) = Nilai probabilitas pada hipotesis R

P(S|R) = Nilai Probabilitas S yang berdasarkan dengan kondisi hipotesis R

P(S) = Nilai probabilitas S

Atau menggunakan rumus 2.7 berikut

$$P(vj) = \frac{N}{Jumlah} \quad \textbf{(2.7)}$$

Keterangan:

P(*vj)* = Probabilitas hipotesis vj *(prior)*

N = Jumlah data training dimana *v = vj*

Jumlah = Jumlah data training.

Contoh soal :

Data mobil tercuri pada tabel dibawah ini

Tabel 2. 8 Contoh Soal *Naïve Bayes*

Warna (a1)	Tipe (a2)	Asal (a4)	Tercuri ? (vj)
Merah	Sport	Domestik	Ya
Merah	Sport	Domestik	Tidak
Merah	Sport	Domestik	Ya
Kuning	Sport	Domestik	Tidak
Kuning	Sport	Import	Ya
Kuning	SUV	Import	Tidak
Kuning	SUV	Import	Ya
Kuning	SUV	Domestik	Tidak
Merah	SUV	Import	Tidak
Merah	Sport	Import	Ya

Dari tabel di atas, data mobil yang tercuri bisa dilihat dari atribut warna, tipe, dan asal. Misalkan kita ingin mengelompokkan mobil warna merah, tipe SUV, dan asal domestik. tentukan probabilitas tercuri dan probabilitas tidak tercuri, dan kemudian tentukan berapa persen mobil yang tercuri dan berapa persen mobil yang

tidak tercuri, serta tentukan mobil dengan warna merah, tipe SUV, dan asal domestik tersebut tercuri atau tidak?

Langkah penyelesaian :

Menghitung mobil tercuri Ya dan Tidak (P*(vj)*), menggunakan rumus 2.7

P (ya) $= \frac{5}{10} = 0.5$

P (tidak) $= \frac{5}{10} = 0.5$

Menghitung probabilitas Ya tercuri pada mobil warna merah, tipe SUV, dan asal domestik

P (merah | ya) $= \frac{3}{5} = 0.6$

P (SUV | ya) $= \frac{1}{5} = 0.2$

P (domestik | ya) $= \frac{2}{5} = 0.4$

Menghitung probabilitas tidak tercuri pada mobil warna merah, tipe SUV, dan asal domestik

P (merah | tidak) $= \frac{2}{5} = 0.4$

P (SUV | tidak) $= \frac{3}{5} = 0.6$

P (domestik | tidak) $= \frac{3}{5} = 0.6$

Menentukan berapa persen mobil tercuri dan tidak tercuri

Tercuri (ya) = P (ya) * P (merah|ya) * P (SUV|ya) * P(domestik|ya)

= 0.5 * 0.6 * 0.2 * 0.4

= 0.024 atau 2,4 %

Tercuri (tidak) = P (tidak) * P (merah|tidak) * P (SUV|tidak) *P (domestik|tidak)

= 0.5 * 0.4 * 0.6 * 0.6

= 0.072 atau 7,2%

Jadi, berdasarkan hasil perhitungan tercuri di atas dengan hasil tercuri (tidak) > tercuri (ya) yaitu 7.2% > 2.4% maka dapat disimpulkan mobil dengan warna merah, tipe SUV, dan asal domestik tidak tercuri.

2.2.5 Evaluasi Model

2.2.5.1 *K-Fold Cross Validation*

K-Fold Cross Validation adalah sebuah teknik validasi silang yang digunakan untuk menguji kinerja model (Hafid, 2023). K-fold cross-validation dimulai secara acak untuk memecah data partisi menjadi kelompok K, bagi masing-masing kelompok (Lyu et al., 2022). Data partisi tersebut diolah sejumlah K kali eksperimen dengan masing-masing eksperimen menggunakan data partisi ke-K sebagai data testing dan menggunakan sisa partisi lainnya sebagai data *training* (Arisandi et al., 2022).

Tujuan penggunaan K-fold cross validation adalah untuk mengurangi bias karena sebagian besar data digunakan untuk melatih jaringan untuk iterasi k. Selain itu, bobot jaringan dari lapisan konvolusional diperbarui terus-menerus untuk setiap iterasi, sehingga meningkatkan efisiensi training (Barile et al., 2022). Validasi silang *k-fold* membagi data sehingga setiap lipatan digunakan sebagai set pengujian pada beberapa titik validasi silang. Skenario yang dapat digunakan adalah 5 kali lipat cross validation (K=5). Kumpulan data dibagi menjadi 5 *fold*. Pada iterasi pertama, lipatan pertama digunakan untuk menguji model dan sisanya digunakan untuk melatih model. Pada iterasi kedua, lipatan kedua digunakan sebagai set pengujian, dan sisanya digunakan sebagai set pelatihan. Proses ini diulangi hingga masing-masing lipatan digunakan sebagai set pengujian (Peryanto et al., 2020).

Dalam penelitian ini menggunakan *k-fold cross validation* sebagai proses untuk memisahkan data *training* dan data *testing* sebanyak 5 fold untuk keempat algoritma yang ada, untuk *k-fold cross validation* menggunakan algoritma *K-Nearest Neighbors* dan *Naïve Bayes* yang digunakan dalam penelitian (Syahril Dwi Prasetyo et al., 2023) dengan KNN sebagai algoritma terbaik, dan mendapatkan nilai accuracy 88.12%. Juga *k-fold cross validation* menggunakan *Decission Tree* C4.5 pernah digunakan dalam klasifikasi tunjangan kinerja pegawai yang dilakukan oleh (Dwita & Tarigan, 2023) dan mendapat *accuracy* sebesar 97%. Dan *k-fold cross validation* menggunakan algoritma *Random Forest* pernah digunakan dalam penelitian (Adriansyah et al., 2022) untuk klasifikasi kemampuan beradaptasi pembelajaran jarak jauh siswa dengan hasil accuracy sebesar 91.5%

2.2.6 Metrik Evaluasi

Metrik evaluasi digunakan untuk mengukur kinerja suatu model dalam menyelesaikan tugas tertentu, seperti klasifikasi, regresi, atau tugas lainnya. Metrik yang digunakan dalam penelitian ini untuk evaluasi adalah accuracy, presision, recall, dan F1-score. Rumus yang digunakan untuk menghitung hasil evaluasi dari setiap metrik adalah sebagai berikut :

1. Akurasi: Akurasi mengukur seberapa baik model dalam mengklasifikasikan semua kelas dengan benar.

$$akurasi = \frac{TP + TN}{TP + TN + FP + FN} \quad \textbf{(2.8)}$$

2. Presisis: Presisi mengukur seberapa banyak dari yang diklasifikasikan sebagai positif oleh model yang benar-benar positif.

$$precision = \frac{TP}{TP + FP} \quad \textbf{(2.9)}$$

3. Recall: Recall mengukur seberapa banyak dari keseluruhan positif yang telah diidentifikasi dengan benar oleh model.

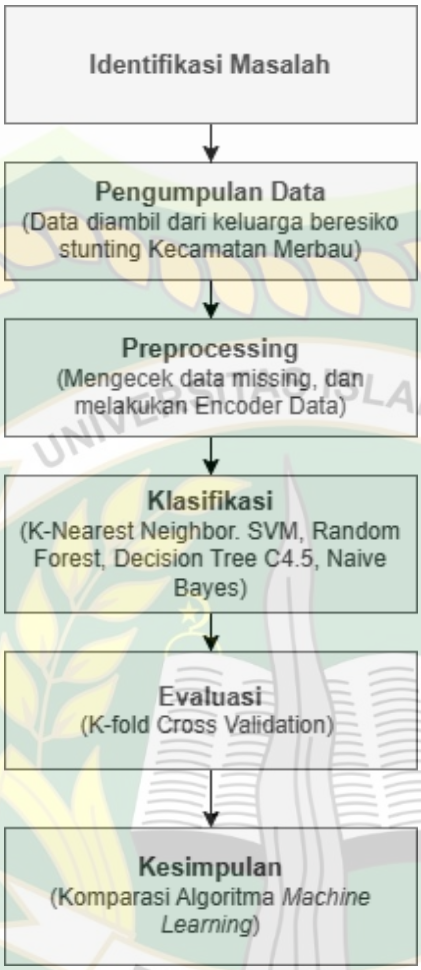
$$recall = \frac{TP}{TP + FN} \quad \textbf{(2.10)}$$

4. F1-Score: F1-Score adalah ukuran gabungan antara presisi dan recall, memberikan gambaran yang lebih lengkap tentang kinerja model.

$$\text{F1} = 2\text{X}\ \frac{\text{Presisi x } Recall}{\text{Presisi x } Recall} \quad \textbf{(2.11)}$$

2.3 Kerangka Pemikiran

Kerangka pemikiran yang diterapkan pada "Perbandingan metode *machine learning* dalam klasifikasi potensi keluarga beresiko stunting pada kecamatan Merbau" dapat dilihat pada Gambar 2.4 tersebut terdapat beberapa tahapan yaitu tahap identifikasi masalah, tahap pengumpulan data, tahap *preprocessing*, tahap klasifikasi menggunakan algoritma KNN, *Random Forest*, *Decission Tree* C4.5 dan *Naïve Bayes* dan tahap selanjutnya adalah evaluasi menggunakan *K-fold Cross Validation* serta tahap terakhir yaitu penarikan kesimpulan dari komparasi algoritma *Machine Learning* yang digunakan dalam penelitian ini.



Gambar 2. 4 Alur Kerangka Berpikir

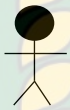
2.4 Alat Bantu Perancangan Sistem

Alat bantu perancangan sistem digunakan untuk menggambarkan proses sistem secara visual. Alat perancangan sistem yang digunakan dalam penelitian ini adalah UML (*Unified Modeling Language*), *Unified Modelling Language* adalah suatu alat untuk memvisualisasikan dan mendokumentasikan hasil analisa dan desain yang berisi sintak dalam memodelkan sistem secara visual (Haviluddin, 2011). Adapun diagram *Unified Modelling Language* (UML) yang digunakan dalam penelitian ini adalah *Use Case Diagram*, dan *Context Diagram*.

2.4.1 *Use Case Diagram*

Use Case Diagram adalah visualisasi dari kemampuan yang diinginkan dari suatu sistem, mencerminkan interaksi antara aktor dan sistem. Dalam diagram ini, aktor mencitrakan entitas manusia atau sistem yang menjalankan tugas dalam sistem tersebut (Prihandoyo M Teguh, 2018).

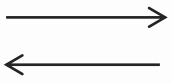
Tabel 2. 9 Simbol dan Fungsi *Use Case Diagram*

No	Simbol	Keterangan	Fungsi
1.		*Use Case*	Menjelaskan bagian utama dari kegunaan sistem
2.		Aktor / *Actor*	Mendefinisikan individu atau objek yang menggunakan atau berinteraksi dengan sistem
3.		Asosiasi	Sebagai penghubung antara aktor dengan *use case* yang saling berinteraksi
4.		*System*	Menspesifikasikan paket yang menampilkan system secara terbatas

2.4.2 *Context Diagram*

Context diagram adalah representasi visual dari keseluruhan sistem yang menunjukkan hubungannya dengan proses tertentu dan entitas eksternal. *Context diagram* menunjukkan sistem yang dirancang secara keseluruhan, semua entitas eksternal harus mewakili aliran data melalui input, proses, dan output dengan cara yang masuk akal, sehingga terlihat data yang mengalir pada input-proses-output. *Context diagram* menggunakan tiga symbol, yaitu simbol untuk mewakili entitas eksternal, simbol untuk mewakili aliran data, dan simbol untuk mewakili proses. *Context diagram* hanya dapat terdiri dari satu proses, bukan beberapa proses (Soufitri, 2019).

Tabel 2. 10 Simbol dan Fungsi *Context Diagram*

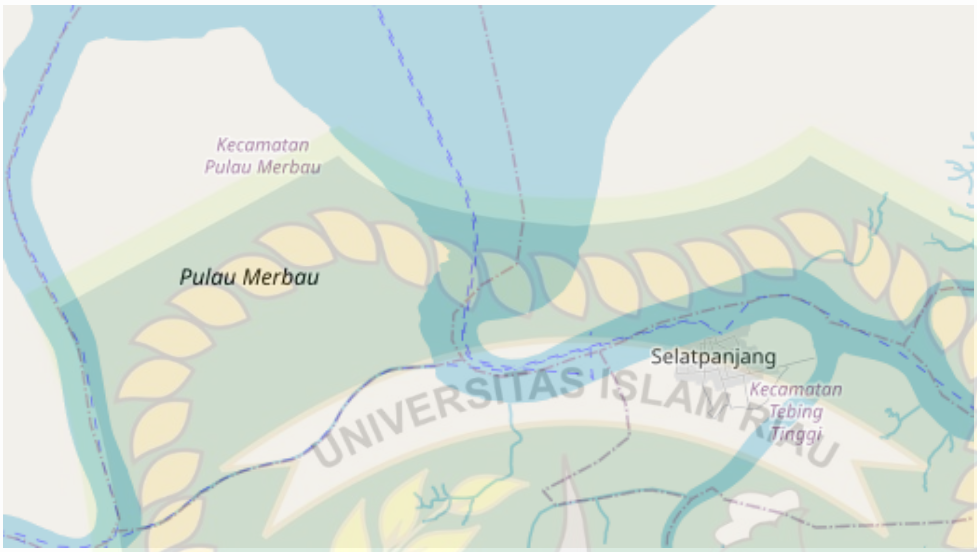
No	Simbol	Nama	Fungsi
1.		Entitas Eksternal / Terminator	Menggambarkan asal atau tujuan data di luar sistem
2.		Proses	Menggambarkan proses dimana aliran data masuk ditransformasikan ke aliran data keluar
3.		Aliran Data	Menggambarkan aliran / perpindahan data
4.		*Data Store*	Digunakan untuk menyimpan data baru

BAB III
METODOLOGI PENELITIAN

3.1 Tinjauan Umum Objek Penelitian

Kecamatan Pulau Merbau adalah salah satu kecamatan dari 11 (sebelas) Kecamatan yang ada di Kabupaten KepulauanMeranti. Kecamatan Pulau Merbau merupakan pemekaran dari Kecamatan Merbau, saat ini, Ibu Kota Kecamatan Pulau Merbau adalah Desa Renak Dungun. Kecamatan Pulau Merbau disahkan sebagai sebuah Kecamatan di Wilayah Kabupaten Kepulauan Meranti pada tanggal 26 Januari 2011. Dalam struktur pemerintahan Kecamatan Pulau Merbau secara administrasi terdiri dari 11 (sebelas) desa. Adapun sebelas desa tersebut adalah Desa Semukut, Desa Centai, Desa Batang Meranti, Desa Tanjung Bunga, Desa Kuala Merbau, Desa Pangkalan Balai, Desa Renak Dungun, Desa Baran Melintang, Desa Ketapang Permai, Desa Teluk Ketapang dan Desa Padang Kamal.

Kecamatan Pulau Merbau merupakan pulau yang berada ditengah-tengah diantara Pulau Ransang, Pulau Tebing Tinggi, Pulau Padang dan berhadapan dengan Selat Malaka. Terletak pada Koordinat **010 00^0 783 LU 102^0 350 07^0 BT** dengan Ibukota Kecamatan Semukut, jarak Ibukota Kecamatan Pulau Merbau, Semukut ke Ibu Kota Kabupaten, Selatpanjang ± 25 Km, dan ke Ibu Kota Propinsi Riau,Pekanbaru ±250 Km. Luas wilayah Kecamatan Pulau Merbau seluruhnya ± 455,00 Km^2 . Secara Administrasi batas-batas wilayah Kecamatan Pulau Merbau adalah, sebelah utara; Selat Malaka (Malaysia), sebelah selatan; Selat Rengit, sebelah timur; Selat Air Hitam, sebelah barat; Selat Asam. Pulau Merbau memiliki penduduk berjumlah 15.309 (Kantor camat Merbau, 2023)



Gambar 3. 1 Peta Kecamatan Merbau

3.2 Metode Penelitian

Desain penelitian yang akan diterapkan pada penelitian yang berjudul "Klasifikasi Potensi Keluarga Beresiko *Stunting* Pada Kecamatan Merbau Menggunakan *Machine Learning"* terdiri dari enam tahapan penelitian. Enam tahapan tersebut yaitu (1) Identifikasi Masalah, (2) Pengumpulan Data, (3) *Preprocessing*, (4) Klasifikasi, (5) Evaluasi, (6) Kesimpulan. Pada tahap pertama yaitu terdapat identifikasi masalah, tahap ini dilakukan identifikasi masalah terlebih dahulu untuk mengetahui permasalahan yang ada. Tahap kedua yaitu pengumpulan data, data yang diambil dari data keluarga beresiko *stunting* di Kecamatan Merbau. Tahap ketiga yaitu *preprocessing* yang akan dilakukan, mengecek data yang *missing* dan melakukan *encording* data. Tahap keempat ialah klasifikasi data yang akan dilakukan klasifikasi dengan keempat algoritma *K-Nearest Neighbo, Random Forest, Decision Tree C4.5* dan *Naïve Bayes*. Tahap kelima yaitu evaluasi dengan menggunakan *k-fold cross validation.* Tahap terakhir yaitu tahap kesimpulan dengan mengetahui hasil perbandingan kinerja algoritma *K-Nearest Neighbor, Random Forest, Decision Tree C4.5* dan *Naïve Bayes* dengan *recall, precision,* dan *accuracy* sebagai pembandingnya. Melalui nilai-nilai tersebut dapat diidentifikasi hasil klasifikasi metode yang terbentuk dari keempat metode tersebut pada penelitian ini.

3.2.1 Identifikasi Masalah

Penelitian ini ditujukan untuk mencari metode yang paling efektif untuk mengklasifikasikan potensi keluarga beresiko *stunting* Kecamatan Merbau. Data tersebut memiliki indikasi-indikasi yang nantinya dapat dijadikan variabel dalam penelitian ini, sehingga dapat dianalisis mengenai keluarga yang berpotensi beresiko *stunting*.

3.2.2 Pengumpulan Data

Data yang digunakan pada penelitian ini merupakan data-data dari 3.349 keluarga hasil *survey* di Kecamatan Merbau untuk mengetahui apakah memiliki resiko *stunting*. Data keluarga beresiko *stunting* ini terdiri dari 10 variabel. Variabel prediktor merupakan variabel yang mempengaruhi, atau bisa disebut dengan variabel independen (disimbolkan X), sedangkan variabel respon yaitu variabel yang dipengaruhi, atau bisa disebut dengan variabel dependen (disimbolkan Y). Berikut merupakan variabel prediktor dan variabel respon dari dataset *stunting* Kecamatan Merbau pada Tabel 3.1

Tabel 3. 1 Tabel Atribut Dataset Keluarga Beresiko *Stunting* Kecamatan Merbau

No	Variabel Prediktor
1.	Baduta (0-23 bulan)
2.	Balita (24-59 bulan)
3.	PUS
4.	PUS Hamil
5.	Keluarga Tidak Mempunyai Sumber Air Minum Utama yang Layak
6.	Keluarga Tidak Mempunyai Jamban yang Layak
7.	Terlalu Muda (Umur Istri < 20 Tahun)
8.	Terlalu Tua (Umur Istri 35-40 Tahun)
9.	Terlalu Dekat (Jarak Anak < 2 Tahun)
10.	Terlalu Banyak (> 3 Anak)
	Variabel Respon
1.	Kategori keluarga beresiko *stunting*

Sumber : Pedoman Pelaksanaan Pendataan Keluarga Tahun 2021 (BKKBN, 2020).

3.2.3 *Preprocessing*

Pengolahan awal data merupakan langkah penting. Proses pengolahan awal data dalam data mining dapat mempengaruhi hasil dari data tersebut. Pada tahap ini dilakukan pengecekan data missing dan melakukan encoder data.. Label Encoder merubah data kategorikal menjadi numerikal (Latifah et al., 2019).

3.2.4 Klasifikasi

Klasifikasi data yang akan dilakukan yaitu dengan keempat algoritma *K-Nearest Neighbor, Random Forest, Decision Tree C4.5* dan *Naïve Bayes*. Dapat ditunjukkan seperti Gambar 2.1 merupakan *flowchart* tahapan klasifikasi. Pada tahap pertama yaitu dilakukan *input dataset*, kemudian dilakukan *encoder*, selanjutnya dilakukan *split data* dengan menggunakan proses keempat algoritma klasifikasi diatas, setelah dilakukan pengklasifikasian maka akan tampil prediksi *data test*, setelah itu dilakukan evaluasi menggunakan *k-fold cross validation*, dan yang terakhir adalah keluaran terkait performa pada algoritma *machine learning* yang digunakan pada *dataset* keluarga beresiko *stunting* Kecamatan Merbau.

3.2.5 Evaluasi

Pada tahap evaluasi, akan dibandingkan hasil proses klasifikasi yang telah dilakukan menggunakan lima algoritma klasifikasi yang telah ditentukan. Proses evaluasi yang akan dilakukan untuk menguji hasil klasifikasi tersebut menggunakan metode *K-fold Cross Validation*. *K-fold cross validation* bertujuan untuk menghilangkan bias pada data, dan juga digunakan untuk menilai kinerja algoritma dengan membagi data sebanyak k.

3.2.6 Kesimpulan

Dari hasil yang telah didapatkan melalui berbagai proses yang telah dilakukan, tahapan akhir yang dilakukan adalah tahap evaluasi. Pada tahap evaluasi, akan dibandingkan hasil proses klasifikasi yang telah dilakukan menggunakan lima algoritma klasifikasi yang telah ditentukan, yaitu algoritma *K-Nearest Neighbor, Suport Vector Machine, Random Forest, Decision Tree C4.5, Naïve Bayes*. Algoritma yang memiliki hasil klasifikasi tertinggi, menandakan bahwa algoritma tersebut merupakan algoritma yang lebih baik dalam proses klasifikasi potensi keluarga beresiko *stunting* Kecamatan Merbau.

3.3 Alat dan Bahan Penelitian

3.3.1 Spesifikasi Perangkat Keras *(Hardware)*

Berikut adalah spesifikasi perangkat keras *(Hardware)* komputer yang digunakan dalam proses klasifikasi.

NO	Spesifikasi	Keterangan
1.	Sistem Operasi	Windows 10 Pro 64-bit (10.0, Build 16299)
2.	Processor	Intel® Core ™i3-7020U CPU @ 2.30CHz (4 CPUs), ~2.3GHz
3.	RAM	4.00 GB
4.	System Type	X64-based PC
5.	Graphic	Intel® HD Graphics 620

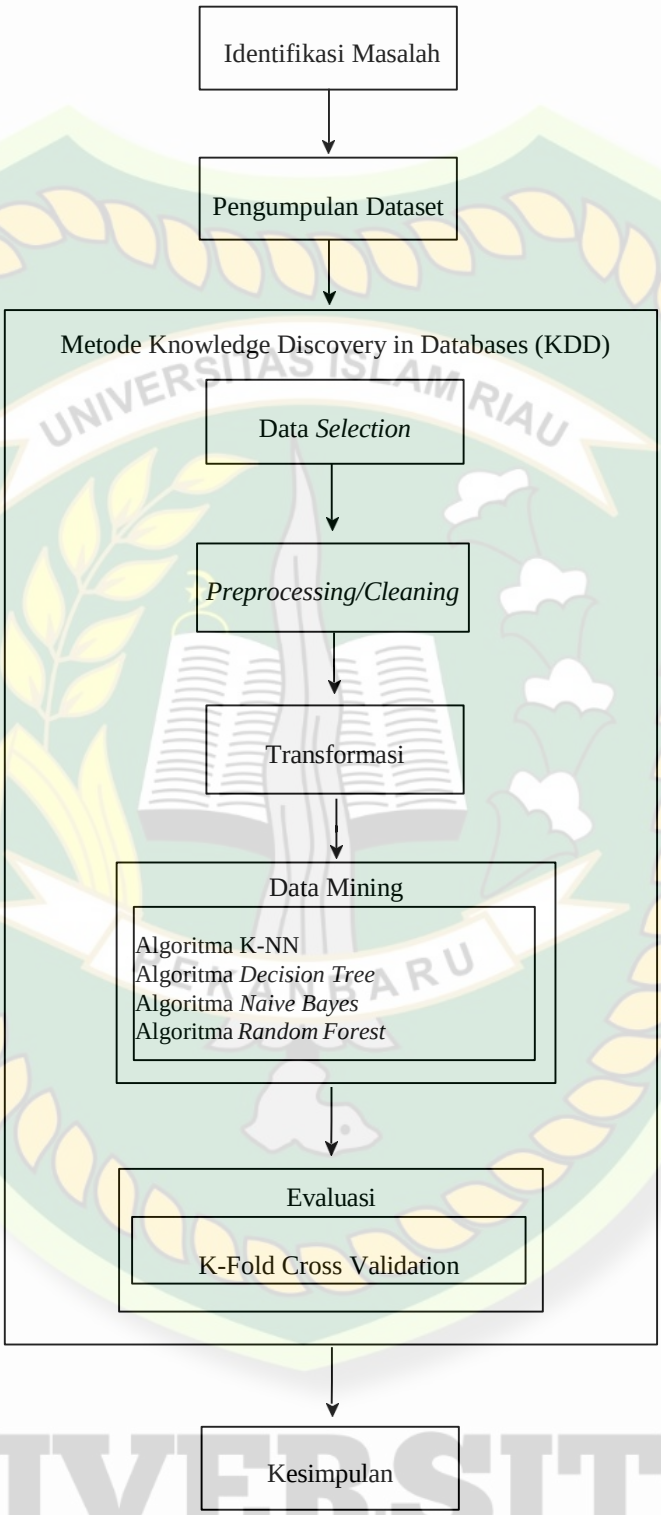
3.3.2 Spesifikasi Perangkat Lunak (*Software*)

Berikut adalah spesifikasi perangkat lunak *(Software)* komputer yang digunakan dalam proses klasifikasi.

NO	Spesifikasi	Keterangan
1.	Sistem Operasi	Windows 10 Pro 64-bit (10.0, Build 16299)
2.	Text Editor	Jupyter Notebook
3.	Bahasa Pemograman	Python

3.4 Perancangan Penelitian

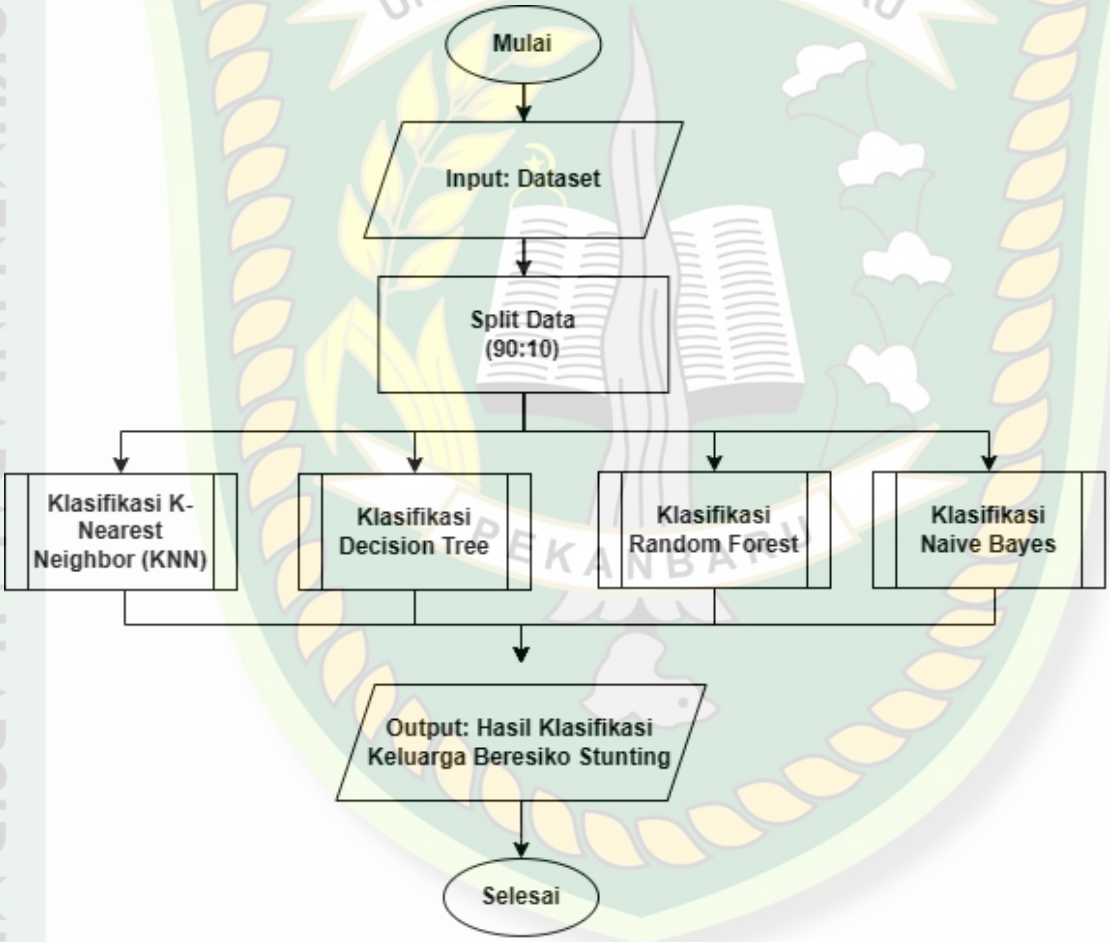
Perancangan proses penelitian yang dilakukan melalui beberapa tahapan yaitu identifikasi masalah, pengumpulan data, tahap *preprocessing* menggunakan metode KDD, tahap klasifikasi menggunakan algoritma *K-Nearest Neighbor*, *Random Forest*, *Decision Tree C4.5* dan *Naïve Bayes*, evaluasi menggunakan *K-fold Cross Validation*, dan tahap terakhir yaitu penarikan kesimpulan dari komparasi algoritma *Machine Learning*. Alur penelitian dapat dilihat pada gambar 3.2



Gambar 3. 2 Alur Proses Penelitian

3.5 Pembentukan Model

Dalam penelitian ini melakukan perbandingan algoritma *machine learning* diantaranya yaitu *K-Nearest Neighbor, Decision Tree, Naïve Bayes,* dan *Random Forest,* untuk mengetahui metode terbaik dalam melakukan klasifikasi potensi keluarga beresiko stunting pada data keluarga di Kecamatan Merbau. Adapun tahapan tahapan yang dilakukan saat melakukan klasifikasi pada data keluarga di Kecamatan Merbau dapat dilihat pada Gambar 3.3



Gambar 3. 3 *Flowchart* Klasifikasi Keseluruhan

3.5.1 *Input Dataset*

Dataset yang digunakan dalam penelitian ini terdiri dari 3 jenis data yaitu data pendataan keluarga (PK), data keluarga berencana (KB), dan data kependudukan(KPD), di Kecamatan Merbau yang diambil pada tahun 2021. Dataset terdiri dari 15 variabel diantaranya No KK, umur_kepala_keluarga, umur_istri, jumlah_baduta, jumlah_balita, pus_kb, pus_hamil_kb3, ber_kb_kb4, tidak_punya_sumber_air_minum_layak, tidak_punya_jamban_layak, istri_terlalu_ muda, istri_terlalu_tua, usia_anak_terlalu_dekat, terlalu_banyak_anak, label. Pada perhitungan manual data yang digunakan sebanyak 100 data dengan teridiri dari 3 label yaitu Tinggi (3), Rendah (2), Tidak (1).

1) Dataset Keterangan Label

Tabel 3. 2 Contoh Dataset Keterangan Label

No	No. KK	umur kepala keluarga	umur istri	jumlah baduta	jumlah balita	pus kb_	pus hamil_ kb3	ber kb_ kb4	tidak punya sumber air minum layak	tidak punya jamban layak	istri terlalu muda	istri terlalu tua	usia anak terlalu dekat	terlalu banyak anak	label
1	1410052026030000100101	46	39	0	3	1	2	2	7	2	0	1	3	2	Tinggi
2	1410052026020000100101	58	0	0	3	3	0	0	7	1	3	3	3	0	Rendah
.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.
99	1410051001000400303603636	56	0	0	3	3	0	0	1	1	3	3	3	0	Tidak
100	1410051001000200400402	79	0	0	3	3	0	0	1	1	3	3	3	0	Tidak

2) Dataset Keluarga Berisiko Stunting

Data dibawah ini yaitu mengubah data label menjadi tipe data numerik yang sebelumnya dari tipe data kategorik.

Tabel 3. 3 Contoh Dataset Keluarga Berisiko Stunting

No	No. KK	umur kepala keluarga	umur istri	jumlah baduta	jumlah balita	pus kb_	pus hamil_ kb3	ber kb_ kb4	tidak punya sumber air minum layak	tidak punya jamban layak	istri terlalu muda	istri terlalu tua	usia anak terlalu dekat	terlalu banyak anak	label
1	1410052026030000100101001	46	39	0	3	1	2	2	7	2	0	1	3	2	3
2	1410052026020000100101001	58	0	0	3	3	0	0	7	1	3	3	3	0	2
.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.
99	1410051001000400303603636	56	0	0	3	3	0	0	1	1	3	3	3	0	1
100	1410051001000200400400202	79	0	0	3	3	0	0	1	1	3	3	3	0	1

Keterangan variabel sebagai berikut:

Tabel 3. 4 Nama keterangan Variabel

Jenis Variabel	Inisiasi	Variabel
Variabel Bebas	Var 1	umur_kepala_keluarga
	Var 2	umur_istri
	Var 3	jumlah_baduta
	Var 4	jumlah_balita
	Var 5	pus_kb
	Var 6	pus_hamil_kb3
	Var 7	ber_kb_kb4
	Var 8	tidak_punya_sumber_air_minum_layak
	Var 9	tidak_punya_jamban_layak
	Var 10	istri_terlalu_muda
	Var 11	istri_terlalu_tua
	Var 12	usia_anak_terlalu_dekat
	Var 13	terlalu_banyak_anak
Variabel Terikat	Var 14	label

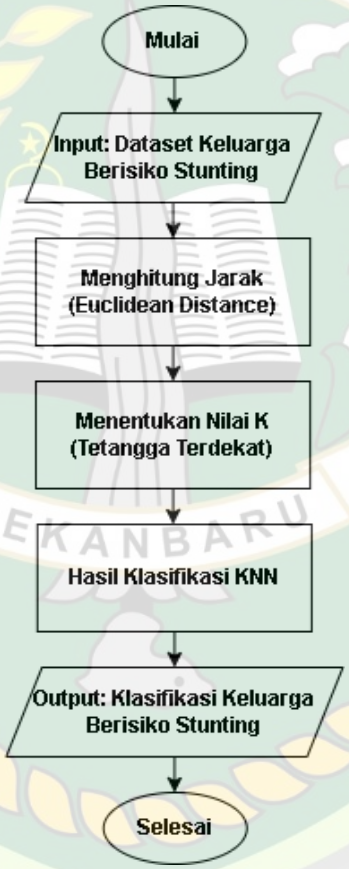
3.5.2 *Split Data*

Split Data atau konsep pembagian data merupakan metode yang digunakan untuk membagi data menjadi dua bagian atau lebih, dimana pada penelitian ini data dibagi menjadi data training dan data testing dengan pembagian 90% sebagai data *training* (data ke 1-90) dan 10% sebagai data *testing* (data ke 91-100).

3.5.3 Klasifikasi

3.5.3.1 Klasifikasi *K-Nearest Neighbor*

Algoritma K-Nearest Neighbor (KNN) merupakan algoritma supervised learning dimana hasil klasifikasi dipengaruhi oleh mayoritas tetangga terdekat ke-k. Algoritma KNN memiliki prinsip kerja yaitu menghitung jarak terdekat dari data baru dengan tetangga terdekatnya, dan hanya mengandalkan memori juga menggunakan mayoritas tetangga untuk klasifikasi. Adapun flowchart dari algoritma KNN dapat dilihat pada Gambar 3.4



Gambar 3. 4 Flowchart Algoritma *K-Nearest Neighbor*

Dari gambar 3.4 yaitu proses klasifikasi keluarga berisiko stunting menggunakan algoritma *k-nearest neighbor* :

1) Input Dataset Keluarga Berisiko Stunting

   Proses input data pada klasifikasi merujuk pada tabel 3.3

2) Menghitung Jarak (*Euclidean Distance*)

   Misalnya Menghitung Jarak Data ke-1:

$$d(x,y) = \sqrt{\begin{matrix}(46-40)^2 + (39-34)^2 + (0-0)^2 + (3-1)^2 + (1-1)^2 + \\ (2-2)^2 + (2-1)^2 + (7-7)^2 + (2-2)^2 + (0-0)^2\end{matrix}}$$

$$d(x,y) = 8{,}185352772$$

Contoh perhitungan *euclidean distance* :

Tabel 3. 5 Contoh perhitungan *Euclidean Distance*

No	No. KK	Var 1	Var 2	Var 3	Var 4	Var 5	Var 6	Var 7	Var 8	Var 9	Var 10	Var 11	Var 12	Var 13	Jarak (Euclidean Distance)	Rank
1	14100520260300001001001	46	39	0	3	1	2	2	7	2	0	1	3	2	8,185352772	73
2	14100520260200001001001	58	0	0	3	3	0	0	7	1	3	3	3	0	38,93584467	31
3	14100520260300001002002	66	0	0	3	3	0	0	7	3	3	3	3	0	43,22036557	14
.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.
88	14100520030100002016018	62	30	0	3	1	2	2	7	2	0	0	3	2	22,47220505	57
89	14100520260100001006008	43	38	0	3	1	2	1	7	1	0	1	3	2	5,567764363	82
90	14100520210300001017020	40	34	0	1	1	2	1	7	2	0	0	3	2	0	90

3) Menentukan Nilai K (Tetangga Terdekat)

Contoh *Data Testing*:

Tabel 3. 6 Contoh *Data Testing*

No	No. KK	Var 1	Var 2	Var 3	Var 4	Var 5	Var 6	Var 7	Var 8	Var 9	Var 10	Var 11	Var 12	Var 13
91	14100520030400 001003004	29	27	0	1	1	2	1	7	3	0	0	3	2
92	14100520180200 001003004	27	21	0	1	1	2	1	7	2	0	0	3	2
93	14100520030200 002008011	42	0	0	3	3	0	0	1	1	3	3	3	0

Contoh Klasifikasi *Data Testing*:

Tabel 3. 7 Contoh Klasifikasi *Data Testing*

No	Jarak (Euclidean Distance) Data 91	Rank Data 91	Jarak (Euclidean Distance) Data 92	Rank Data 92	Jarak (Euclidean Distance) Data 93	Rank Data 93
1	20,97617696	66	26,28687886	61	40,03748244	12
2	40,11234224	29	37,92097045	30	17,08800749	65
3	46,18441296	15	44,69899328	14	24,81934729	48
.	.	.	.	.	.	.
88	33,22649545	50	36,20773398	40	37,02701716	21
89	18,05547009	69	23,47338919	67	38,82009789	15
90	13,07669683	75	18,38477631	74	35,09985755	25

Pengurutan dari hasil perhitungan. Jarak diurutkan dari yang paling dekat jaraknya ke yang paling jauh. Setelah diurutkan diperoleh sebagai berikut, menentukan kelompok data hasil uji berdasarkan label mayoritas dari K tetangga terdekat. Karena nilai K = 3, maka diambil 3 jarak terkecil yaitu:

Tabel 3. 8 Contoh Klasifikasi K=3

Data Testing	Nilai K=3			Hasil Klasifikasi
	Jarak (Euclidean Distance)	Dokumen	Label Aktual Dokumen	
91	70,66116331	68	2	2
	60,69596362	32	2	
	58,05170109	23	2	
92	70,46985171	68	2	2
	60,10823571	32	2	
	57,33236433	23	2	
93	52,34500931	68	2	2
	48,35286961	12	2	
	47,08502947	50	2	

4) Klasifikasi KNN

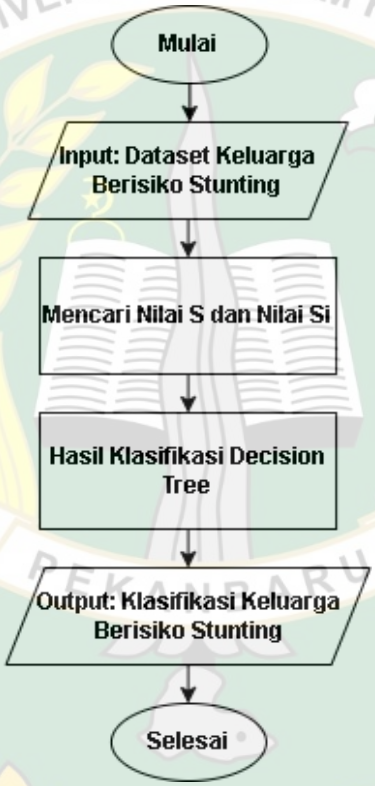
Contoh klasifikasi KNN menggunakan nilai K=3 pada data testing data 91 sampai data 93 menghasilkan klasifikasi yaitu 2 (Rendah) atau keluarga dapat dikatakan berisiko Rendah Stunting.

Tabel 3. 9 Contoh Klasifikasi KNN

No	Klasifikasi	Aktual Label
91	2	2
92	2	2
93	2	1

3.5.3.2 Klasifikasi *Decision Tree*

Algoritma *Decision Tree* merupakan algoritma algoritma *machine* learning yang menggunakan aturan untuk membuat keputusan dengan struktur seperti pohon (*tree*). Pohon keputusan juga berguna untuk mengeksplorasi data, menemukan hubungan tersembunyi antara sejumlah calon variabel input dengan sebuat variabel target. Adapun flowchart dari algoritma *Decision Tree* dapat dilihat pada Gambar 3.5



Gambar 3. 5 *Flowchart* Algoritma *Decision Tree*

Dari gambar 3.5 yaitu proses klasifikasi keluarga berisiko stuning menggunakan algoritma *decision tree*:

1) Input Dataset Keluarga Berisiko Stunting

Proses input data pada klasifikasi yaitu merujuk pada tabel 3.3.

2) Mencari Nilai S dan Nilai Si

Perhitungan Gain dan Entrophy Data Training:

Diketahui:

S = 90

Si Tinggi (3) = 4

Si Rendah (2) = 80

Si Tidak (1) = 6

Misalnya Menghitung Nilai Entrophy Total::

$$Entrophy\ Total = (\left(-\frac{4}{90}\right) * IMLOG2\left(\frac{4}{90}\right) + \left(-\frac{80}{90}\right) * IMLOG2\left(\frac{80}{90}\right) + \left(-\frac{6}{90}\right) * IMLOG2\left(\frac{6}{90}\right))$$

$$Entrophy\ Total = 0{,}611141734$$

Misalnya Menghitung Nilai Entrophy Var 1 (umur_kepala_keluarga):

$$Entrophy(Var\ 1) = (\left(-\frac{0}{1}\right) * IMLOG2\left(\frac{0}{1}\right) + \left(-\frac{1}{1}\right) * IMLOG2\left(\frac{1}{1}\right) + \left(-\frac{0}{1}\right) * IMLOG2\left(\frac{0}{1}\right))$$

$$Entrophy(Var\ 1) = 0$$

Misalnya Menghitung Nilai Gain Var 1 (umur_kepala_keluarga):

$$Gain = (0{,}611141734) - \left(\left(\frac{1}{90}\right) * 0\right) + \left(\left(\frac{1}{90}\right) * 0\right), \ldots + \left(\left(\frac{1}{90}\right) * 0\right)$$

$$Gain = \ 0{,}611141734$$

Contoh Perhitungan *Entrophy* dan *Gain Data Training* :

Tabel 3. 10 Contoh Perhitungan *Entrophy* dan *Gain Data Training*

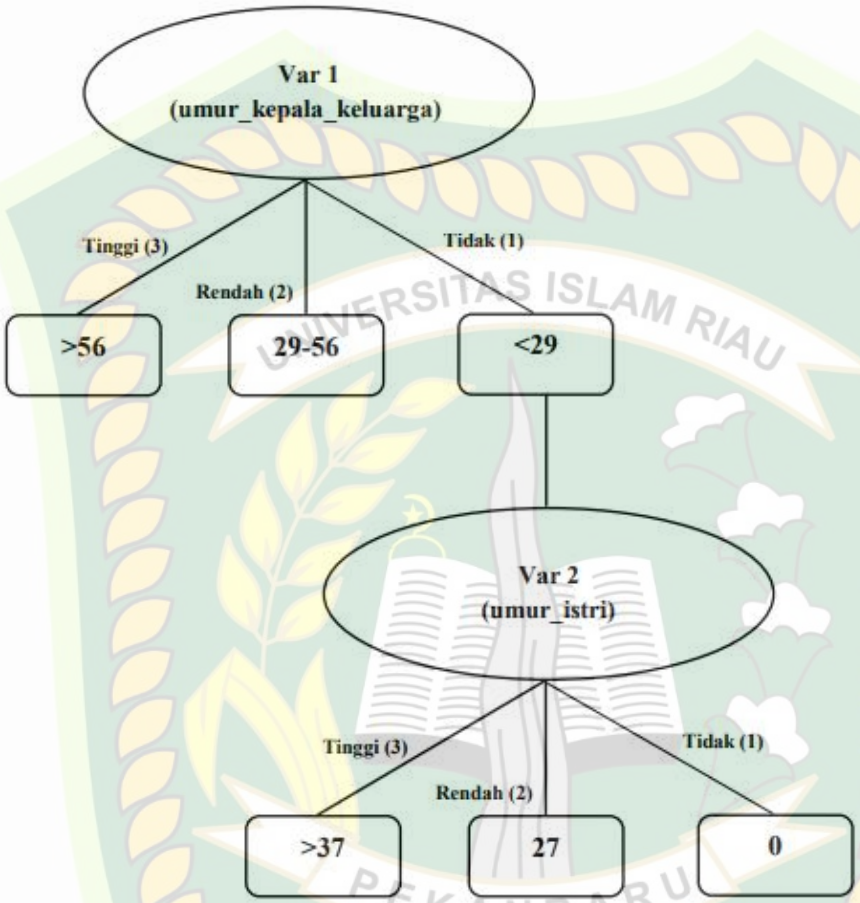
		Jumlah (S)	Si Tinggi (3)	Si Rendah (2)	Si Tidak (1)	Entrophy	Gain
Total		90	4	80	6	0,611141734	
Var 1							0,611141734
	21	1	0	1	0	0	
	22	1	0	1	0	0	
	30	1	0	1	0	0	

Contoh Perhitungan *Gain* dan *Entrophy Data Testing*:

Tabel 3. 11 Contoh Perhitungan *Gain* dan *Entrophy Data Testing*

		Jumlah (S)	Si Tinggi (3)	Si Rendah (2)	Si Tidak (1)	Entrophy	Gain
Total		10	1	5	4	1,360964047	
Var 1							0,611141734
	27	1	0	1	0	0	
	28	1	0	1	0	0	
	29	1	0	1	0	0	

Contoh pohon keputusan *data testing* sebagai berikut:



Gambar 3. 6 Contoh Pohon Keputusan Decision Tree

Jadi dari contoh keputusan (*tree*) yang menjadi akar adalah Var 1 (umur_kepala_keluarga), dengan nilai 1 sebagai label Tidak (1), nilai 2 sebagai label Rendah (2), nilai >3 sebagai label Tinggi (3). Kemudian variabel yang mempengaruhi yaitu Var 2 (umur_istri).

Misalnya pengecekan keputusan pada data ke 91 yaitu diperoleh bahwa dicek pada var 1 dengan nilai data 1 maka masuk dalam label Tidak (1). Kemudian dicek kembali pada var 2 dengan nilai data 0 maka masuk dalam label Tidak (1).

3) Klasifikasi *Decision Tree*

Klasifikasi *Decision Tree* pada data testing data 91 sampai data 94 menghasilkan klasifikasi yaitu 1 (Stunting) sesuai dengan class aktualnya.

Tabel 3. 12 Contoh Klasifikasi *Decision Tree*

No	No. KK		Klasifikasi *Decision Tree*	Aktual Label
91	14100520030400	001003004	2	2
92	14100520180200	001003004	2	2
93	14100520030200	002008011	2	1
94	14100520030400	001010012	2	3

3.5.3.3 Klasifikasi *Naïve Bayes*

Algoritma *naïve bayes* termasuk dalam pengklasifikasian dengan metode probabilitas dan statistic yang ditemukan oleh ilmuan Inggris Thomas *Bayes*, yaitu memprediksi peluang dimasa yang akan dating berdasarkan pengalaman dimasa sebelumnya, sehingga dikenal sebagai Teorema *Bayes*. Teorema tersebut dikombinasikan dengan *naïve* dimana asumsi kondisi antar variabel yang saling bebas. Klasifikasi *naïve bayes* mengasumsikan bahwa ada atau tidak ciri tertentu dari sebuah kelas tidak ada kaitannya dengan ciri dari kelas lainnya. Adapun *flowchart* dari algoritma *naïve bayes* dapat dilihat pada Gambar 3.7.

Gambar 3. 7 *Flowchart* Algoritma *Naive Bayes*

Dari Gambar 3.7 yaitu proses klasifikasi keluarga berisiko stuning menggunakan algoritma naïve bayes:

1) Input Dataset Keluarga Berisiko Stunting

   Proses input data pada klasifikasi merujuk pada tabel 3.3

2) Menghitung Probabilitas Class

   Diketahui jumlah dokumen yang digunakan untuk data training yaitu 90 data dan data testing yaitu 10 data.

   Tabel 3. 13 Diketahui *N Class Data Training* dan *Data Testing*

Nama Label	N Class
Class Tinggi	4
Class Rendah	80
Class Tidak	6
∑Doc Training	**90**
Class Tinggi	1
Class Rendah	5
Class Tidak	4
∑Doc Testing	**10**

   Misalnya menghitung Probabilitas Class 3 (Tinggi):

$$P(Tinggi) = \frac{4}{90}$$

$$P(Tinggi) = 0{,}044444444$$

   Misalnya menghitung Probabilitas Class 2 (Rendah):

$$P(Rendah) = \frac{80}{90}$$

$$P(Rendah) = 0{,}888888889$$

   Misalnya menghitung Probabilitas Class 1 (Tidak):

$$P(Tidak) = \frac{6}{90}$$

$$P(Tidak) = 0{,}066666667$$

3) Menghitung Teorema Bayes

Misalnya Menghitung Teorema Bayes untuk Fitur Pertama Field 21 pada Class Tinggi:

$$P(w|3) = \frac{0 * 0}{4}$$

$$P(w|3) = \frac{0}{4}$$

$$P(w|3) = 0$$

Misalnya Menghitung Teorema Bayes untuk Fitur Pertama Field 21 pada Class Rendah:

$$P(w|2) = \frac{1 * 1}{80}$$

$$P(w|2) = \frac{1}{80}$$

$$P(w|2) = 0{,}0125$$

Misalnya Menghitung Teorema Bayes untuk Fitur Pertama Field 21 pada Class Tidak:

$$P(w|1) = \frac{0 * 0}{6}$$

$$P(w|1) = \frac{0}{6}$$

$$P(w|1) = 0$$

Contoh Perhitungan Conditional Probabilitas *Feature* atau Variabel:

Tabel 3. 14 Contoh Perhitungan Conditional Probabilitas *Feature* atau Variabel

Variabel	Field	Conditional Probabilitas		
		P(W|3)	P(W|2)	P(W|1)
Var 1	21	0	0,0125	0
	22	0	0,0125	0
	30	0	0,0125	0
	32	0,25	0,025	0
	34	0	0,0375	0
	35	0	0,0375	0
	37	0	0,0125	0
	38	0	0,05	0
	39	0	0,05	0
	40	0	0,025	0
	41	0,25	0,0125	0
	42	0,25	0	0
	43	0	0,025	0
	44	0	0,0125	0
	46	0,25	0,0125	0
	44	0	0,025	0
	45	0	0,025	0
	46	0	0,0125	0
	47	0	0,0125	0

Untuk fitur bertipe numerik (kontinu) yaitu contohnya pada var 1 dengan nilai field 40,41,42,3, dst data field cenderung kontinu atau berkelanjutan. Distribusi Gaussian biasanya dipilih untuk merepresentasikan probabilitas bersyarat dari fitur kontinu pada sebuah kelas P(xi|Y), sedangkan distribusi Gaussian dikarakteristikkan dengan dua parameter: mean, m , dan variansi, 2 s . Untuk setiap kelas yi, probabilitas bersyarat kelas yi untuk fitur xi adalah.

4) Menghitung Mean dan Standard Deviasi

Langkah pertama memisahkan data Field masing-masing label:

a) Data Dengan Label Tinggi

Tabel 3. 15 Contoh Data Dengan Label Tinggi

No	No. KK	Var 1	Var 2	Var 3	Var 4	Var 5	Var 6	Var 7	Var 8	Var 9	Var 10	Var 11	Var 12	Var 13
1	14100520260300 001001001	46	39	0	3	1	2	2	7	2	0	1	3	2
38	14100520210300 001008009	41	39	0	3	1	2	1	7	2	0	1	3	2

b) Data Dengan Label Rendah

Tabel 3. 16 Contoh Data Dengan Label Rendah

No	No. KK	Var 1	Var 2	Var 3	Var 4	Var 5	Var 6	Var 7	Var 8	Var 9	Var 10	Var 11	Var 12	Var 13
2	14100520260200 001001001	58	0	0	3	3	0	0	7	1	3	3	3	0
3	14100520260300 001002002	66	0	0	3	3	0	0	7	3	3	3	3	0

Data Dengan Label Tidak

Tabel 3. 17 Contoh Data Dengan Label Rendah

No	No. KK	Var 1	Var 2	Var 3	Var 4	Var 5	Var 6	Var 7	Var 8	Var 9	Var 10	Var 11	Var 12	Var 13
15	14100510010005 001017028	50	0	0	3	3	0	0	4	1	3	3	3	0
16	14100510010005 001025039	68	0	0	3	3	0	0	4	1	3	3	3	0

Langkah kedua menghitung nilai Mean terhadap Field masing-masing label:

Sehingga dari hasil memisahkan data Field terhadap masing-masing label, berikut hasil nilai Mean dan Standar Deviasi masing-masing label:

a) Nilai Mean

Tabel 3. 18 Contoh Nilai Mean

No	Label	Var 1	Var 2	Var 3	Var 4	Var 5	Var 6	Var 7	Var 8	Var 9	Var 10	Var 11	Var 12	Var 13
1	Tinggi (3)	40	34	0	2,5	1	2	1,25	6,5	2,5	0,25	0,75	3	2
2	Rendah (2)	58	13	0,038	2,7	2,2	0,75	0,52	6,73	1,66	1,84	2,01	3	0,75
3	Tidak (1)	59	0	0	3	3	0	0	2	1	3	3	3	0

b) Nilai Standar Deviasi

Tabel 3. 19 Contoh Nilai Standar Deviasi

No	Label	Var 1	Var 2	Var 3	Var 4	Var 5	Var 6	Var 7	Var 8	Var 9	Var 10	Var 11	Var 12	Var 13
1	Tinggi (3)	5,90	10,17	0	1	0	0	0,5	1	0,58	0,5	0,5	0	0
2	Rendah (2)	14,74	17,55	0,19	0,70	0,98	0,96	0,76	1,17	0,81	1,47	1,29	0	0,96
3	Tidak (1)	6,74	0	0	0	0	0	0	1,55	0	0	0	0	0

5) Menghitung Distribusi Gaussian Naïve Bayes

Misalnya menghitung distribusi gaussian pada data ke 91 pada label Tinggi terhadap Var 1:

$$P(X_i) = x_i|Y = y_i = \frac{1}{\sqrt{2 * 3{,}14 * 5{,}909032634}} \exp^{-\frac{(29-40{,}25)^2}{2*5{,}909032634^2}}$$

$$P(X_i) = x_i|Y = y_i = 0{,}02680205$$

Perhitungan distribusi gaussian:

Tabel 3. 20 Contoh Perhitungan Distribusi Gaussian

No	Label	Var 1	Var 2	Var 3	Var 4	Var 5	Var 6	Var 7	Var 8	Var 9	Var 10	Var 11	Var 12	Var 13
91	Tinggi (3)	0,03	0,097	0	0,13	0	0	0,50	0,35	0,36	0,50	0,18	0	0
	Rendah (2)	0,03	0,07	0,89	0,02	0,18	0,17	0,38	0,36	0,11	0,15	0,10	0	0,17
	Tidak (1)	9,62	0	0	0	0	0	0	0,0018	0	0	0	0	0
92	Tinggi (3)	0,01	0,05	0	0,13	0	0	0,50	0,35	0,36	0,50	0,18	0	0
	Rendah (2)	0,02	0,09	0,89	0,023	0,18	0,17	0,38	0,36	0,40	0,15	0,10	0	0,17
	Tidak (1)	2,50	0	0	0	0	0	0	0,0017	0	0	0	0	0

6) Klasifikasi Naïve Bayes

Misalnya menghitung posterior data ke 91 pada label Tinggi:

$$P(d|c) = (0{,}02680205 * 0{,}097052818 * 0{,}129550438 * 0{,}498021814 * 0{,}352154602 * 0{,}360944195 * 0{,}498021814 * 0{,}183211987) * (0{,}044444444)$$

$$P(d|c) = 8{,}65081E-08 \; atau \; 0{,}0000000865081$$

Perhitungan posterior:

Tabel 3. 21 Contoh Perhitungan Klasifikasi Naive Bayes

No	No. KK	Nilai Posterior	Nilai Max	Hasil Klasifikasi	Aktual Label
91	Tinggi (3)	8,65081E-08	8,65081E-08	3	2
	Rendah (2)	4,98169E-11			
	Tidak (1)	1,12497E-09			
93	Tinggi (3)	1,76679E-28	0,000125701	1	1
	Rendah (2)	7,77394E-13			
	Tidak (1)	0,000125701			
94	Tinggi (3)	1,23255E-06	1,23255E-06	3	3
	Rendah (2)	8,26448E-10			
	Tidak (1)	1,06439E-06			

3.5.3.4 Klasifikasi *Random Forest*

Random Forest merupakan salah satu metode CART (Classification and Regression Tree) dalam data mining dan tidak memerlukan asumsi apapun. Metode ini menggunakan konsep pohon keputusan (decision tree). Model ini dibentuk dari banyak pohon sehingga membentuk sebuah kumpulan pohon seperti hutan (*forest*) dengan menerapkan metode bootstrap *aggregating* (*bagging*) dan *random feature selection*. Adapun *flowchart* dari algoritma *Random Forest* dapat dilihat pada Gambar 3.8

Gambar 3. 8 Flowchart Algoritma *Random Forest*

Dari gambar 3.8 yaitu proses klasifikasi keluarga berisiko stuning menggunakan algoritma *random forest*:

1) Input Dataset Keluarga Berisiko Stunting

   Proses input data pada klasifikasi merujuk pada tabel 3.3.

2) Membuat Model Decision Tree

   Inisialisasi model decision tree dibuat secara acak dalam pengambilan data. Model terdiri dari 3 model sebagai berikut:

   a) Model 1 Decision Tree

Tabel 3. 22 Contoh Model 1 *Decision Tree*

No	No. KK	Var 1	Var 2	Var 3	Var 4	Var 5	Var 6	Var 7	Var 8	Var 9	Var 10	Var 11	Var 12	Var 13	label
1	14100520260300001001001	46	39	0	3	1	2	2	7	2	0	1	3	2	3
2	14100520260200001001001	58	0	0	3	3	0	0	7	1	3	3	3	0	2
3	14100520260300001002002	66	0	0	3	3	0	0	7	3	3	3	3	0	2
4	14100520260200001002002	53	0	0	3	3	0	0	7	3	3	3	3	0	2
5	14100520260200001003003	43	34	0	3	1	2	2	7	1	0	0	3	2	2

b) Model 2 Decision Tree

Tabel 3. 23 Contoh Model 2 *Decision Tree*

No	No. KK	Var 1	Var 2	Var 3	Var 4	Var 5	Var 6	Var 7	Var 8	Var 9	Var 10	Var 11	Var 12	Var 13	label
6	14100520260200001004004	52	0	0	3	3	0	0	7	1	3	3	3	0	2
7	14100520260300001003004	39	33	0	1	1	2	2	1	3	0	0	3	2	2
8	14100520260300001004005	58	0	0	3	3	0	0	7	3	3	3	3	0	2
31	14100520260300001011014	65	0	0	3	3	0	0	7	3	3	3	3	0	2
32	14100520260100001005007	83	0	0	3	3	0	0	7	1	3	3	3	0	2

c) Model 3 Decision Tree

Tabel 3. 24 Contoh Model 3 *Decisioin Tree*

No	No. KK	Var 1	Var 2	Var 3	Var 4	Var 5	Var 6	Var 7	Var 8	Var 9	Var 10	Var 11	Var 12	Var 13	label
9	14100520260300001006007	71	0	0	3	3	0	0	7	1	3	3	3	0	2
10	14100520260300001007009	32	28	0	1	1	2	1	7	1	0	0	3	2	2
13	14100520260300001009012	63	0	0	3	3	0	0	7	1	3	3	3	0	2
14	14100520260300001002003	35	0	0	3	3	0	0	7	3	3	3	3	0	2
20	14100520260200001005006	39	0	0	3	3	0	0	7	1	3	3	3	0	2

3) Menghitung Gain dan Entrophy Model

a) Gain dan Entrophy Model 1

Misalnya Menghitung Nilai Entrophy Total:

$$Entrophy\ Total = (\left(-\frac{1}{30}\right) * IMLOG2\left(\frac{1}{30}\right) + \left(-\frac{26}{30}\right) * IMLOG2\left(\frac{26}{30}\right) + \left(-\frac{3}{30}\right) * IMLOG2\left(\frac{3}{30}\right))$$

$$Entrophy\ Total = 0{,}674679923$$

Misalnya Menghitung Nilai Entrophy Variabel Degree pada Atribut 2:

$$Entrophy(Var\ 1)$$
$$= (\left(-\frac{0}{3}\right) * IMLOG2\left(\frac{0}{3}\right) + \left(-\frac{0}{3}\right) * IMLOG2\left(\frac{0}{3}\right) + \left(-\frac{3}{3}\right) * IMLOG2\left(\frac{3}{3}\right))$$

$$Entrophy(Var\ 1) = 0$$

Misalnya Menghitung Nilai Gain Variabel Degree:

$$Gain = (0{,}674679923) - \left(\left(\frac{3}{30}\right) * 0\right) + \left(\left(\frac{2}{30}\right) * 0\right), \ldots + \left(\left(\frac{2}{30}\right) * 0\right)$$

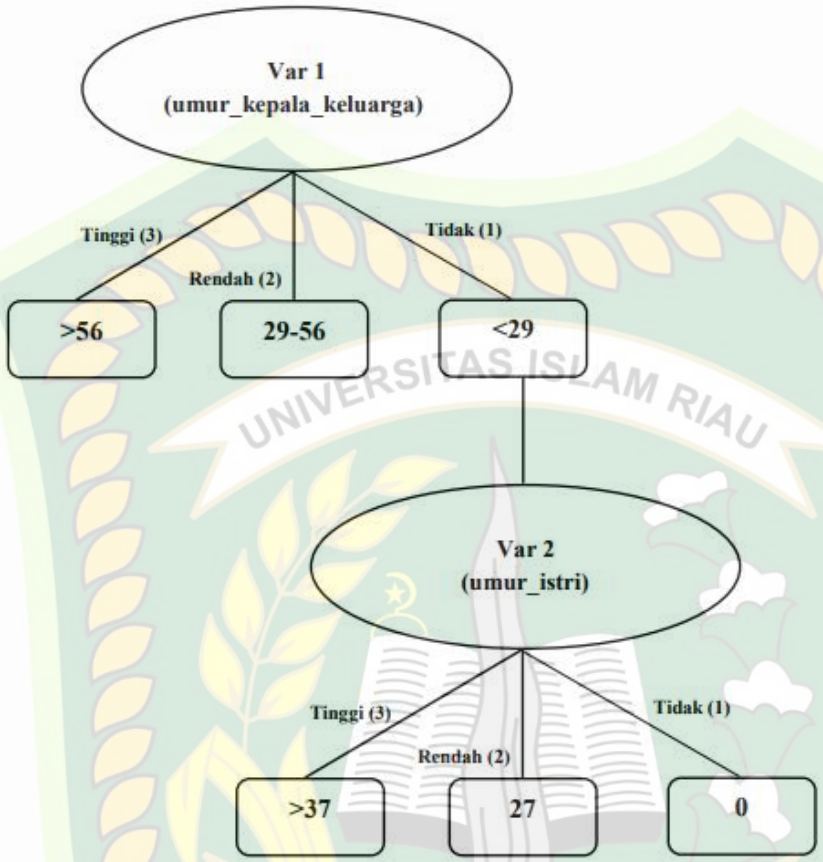
$$Gain = 0{,}674679923$$

Perhitungan Entrophy dan Gain Model 1:

Tabel 3. 25 Contoh Perhitungan Entrophy dan Gain Model 1

		Jumlah (S)	**Si Tinggi (3)**	**Si Rendah (2)**	**Si Tidak (1)**	**Entrophy**	**Gain**
Total		30	1	26	3	0,674679923	
Var 1							0,674679923
	34	3	0	3	0	0	
	35	2	0	2	0	0	
	38	1	0	1	0	0	
Max Gain Model 1							**0,674679923**

Nilai max gain pada model 1 menjadi node atau akar yaitu berapa pada variabel Var 1 (umur_kepala_keluarga).



Gambar 3. 9 Contoh Pohon Keputusan *Random Forest* Model 1

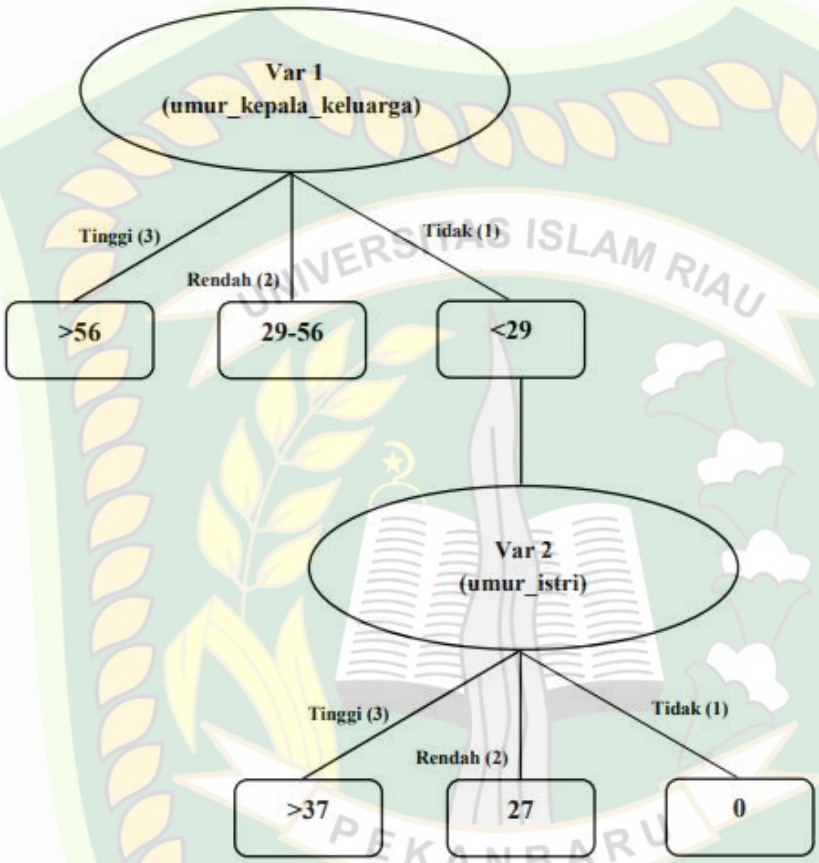
b) Gain dan Entrophy Model 2

Perhitungan Entrophy dan Gain Model 2:

Tabel 3. 26 Contoh Perhitungan Entrophy dan Gain Model 2

		Jumlah (S)	Si Tinggi (3)	Si Rendah (2)	Si Tidak (1)	Entrophy	Gain
Total		30	1	27	2	0,560825177	
Var 1							0,560825177
	21	1	0	1	0	0	
	32	1	0	1	0	0	
	37	1	0	1	0	0	
Max Gain Model 2							**0,560825177**

Nilai max gain pada model 2 menjadi node atau akar yaitu berapa pada variabel Var 1 (umur_kepala_keluarga).



Gambar 3. 10 Contoh Pohon Keputusan *Random Forest* Model 2

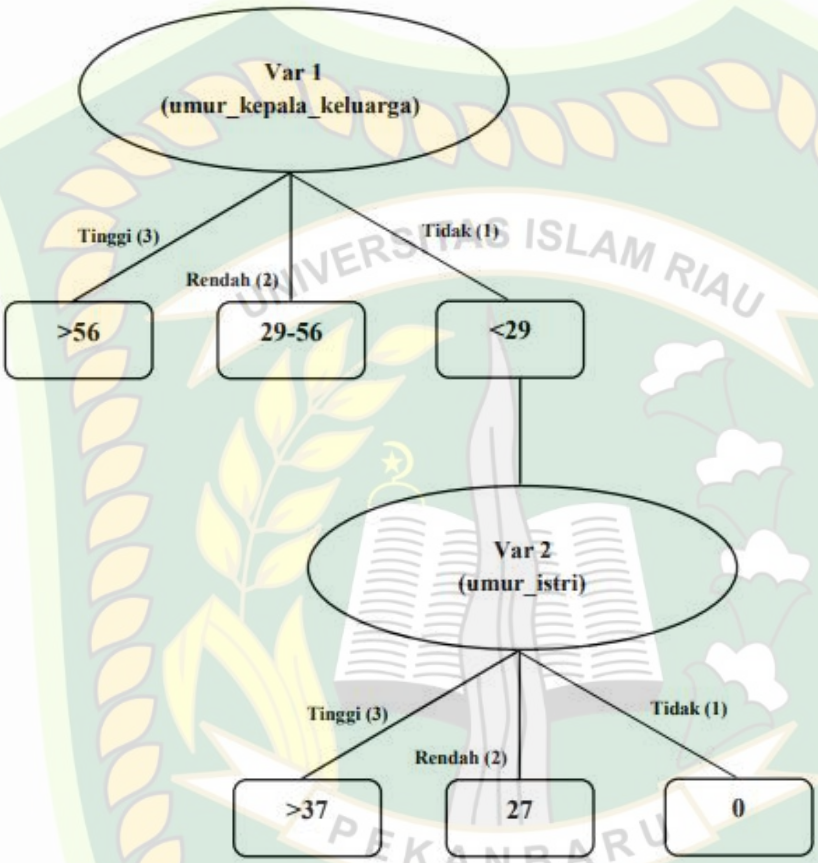
c) Gain dan Entrophy Model 3

Perhitungan Entrophy dan Gain Model 3:

Tabel 3. 27 Contoh Perhitungan Entrophy dan Gain Model 3

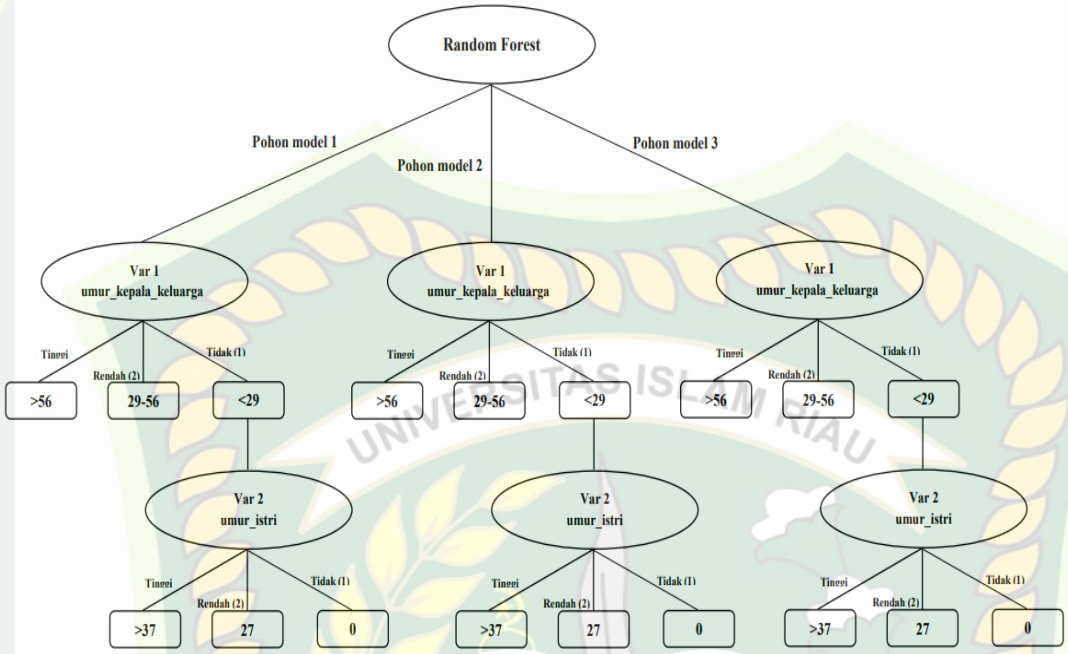
		Jumlah (S)	Si Tinggi (3)	Si Rendah (2)	Si Tidak (1)	Entrophy	Gain
Total		30	2	27	1	0,560825177	
Var 1							0,560825177
	22	1	0	1	0	0	
	30	1	0	1	0	0	
	32	2	1	1	0	0	
Max Gain Model 3							**0,560825177**

Nilai max gain pada model 3 menjadi node atau akar yaitu berapa pada variabel Var 4 (jumlah_balita).



Gambar 3. 11 Pohon Keputusan *Random Forest* Model 3

Dari 3 model Decision Tree yang telah dibuat, maka *Random Forest* akan menggabungkan 3 model tersebut. Sehingga jika divisualisasikan adalah sebagai berikut:



Gambar 3. 12 Contoh Pohon Keputusan *Random Forest*

4) Klasifikasi *Random Forest*

Pada contoh hasil *Random Forest* diatas, didapat bahwa setiap model tree hanya memiliki akar dan daun (leaf). Sehingga Setiap data test akan ditentukan berdasarkan model yang dibuat dengan melihat akar dan leaf nya.

Pada model 1, 2, 3, fitur yang akan dilihat adalah Var 1 (umur_kepala_keluarga). Sehingga setiap satu data test akan selalu memiliki 3 klasifikasi dari 3 model. Untuk menentukan kelas yang akan diklasifikasi akan menggunakan sistem majorit vactory yang dimana jika 2 model mengklasifikasi yes dan 1 model mengklasifikasi no. Maka data tersebut akan diklasifikasi yes.

Tabel 3. 28 Contoh Klasifikasi *Random Forest*

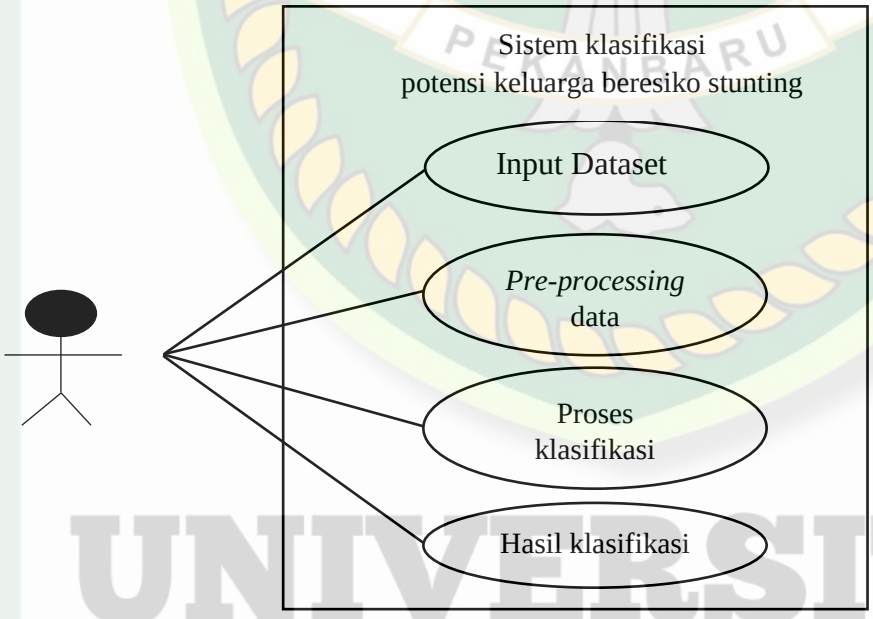
No	Klasifikasi Model 1	Klasifikasi Model 2	Klasifikasi Model 3	Hasil Klasifikasi *Random Forest*	Aktual Label
91	3	3	3	3	2
92	3	3	3	3	2
93	1	1	1	1	1

3.6 Analisa Kebutuhan Sistem

Dalam tahap ini, dibangun sistem yang bertujuan untuk mengelompokkan keluarga berdasarkan kategori tertentu. Aplikasi ini dirancang untuk memudahkan staff BKKBN Provinsi Riau dalam mengkategorikan keluarga dengan potensi beresiko stunting yang tepat

3.6.1 *Use Case Diagram*

Use case diagram menggambarkan ruang lingkup yang akan dibangun agar mendapatkan pemahaman yang lebih baik tentang sistem yang akan dibuat. Berikut proses yang terdapat pada sebuah *use case diagram* :



Gambar 3. 13 Use Case Diagram

Penjelasan dari gambar 3.3 adalah sebagai berikut :

a) Input dataset, staff menginput dataset yang akan di olah

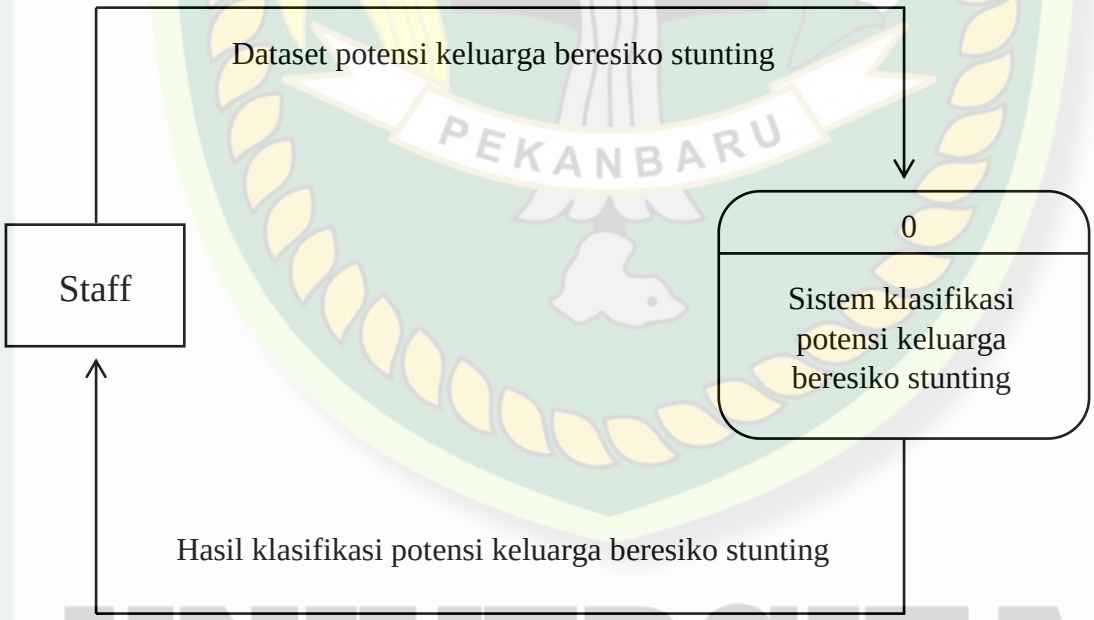
b) Pre-processing data, melakukan pemilahan data dan hanya mengambil data yang digunakan, untuk dapat diterima algoritma

c) Proses klasifikasi, pengklasifikasian dataset berdasarkan kategori yang cocok

d) Hasil klasifikasi, menampilkan hasil klasifikasi potensi keluarga beresiko stunting yang terdiri dari tidak, rendah, dan tinggi

3.6.2 *Context Diagram*

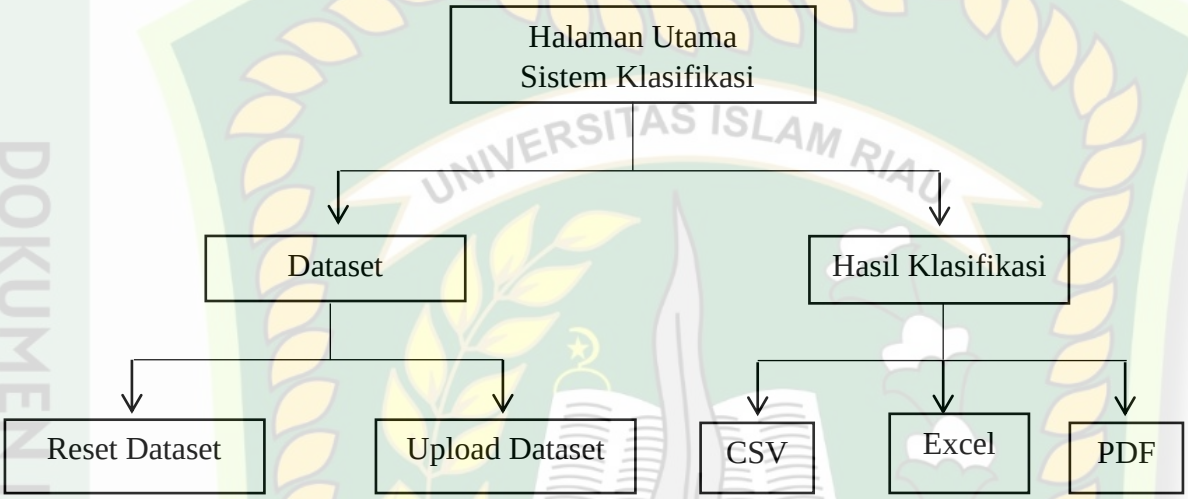
Context Diagram digunakan untuk menggambarkan sistem secara umum, serta menampilkan aliran-aliran proses *input* dan *output* dari entitas eksternal terhadap sistem klasifikasi yang dibangun.



Gambar 3. 14 *Context Diagram*

3.7 Desain Antarmuka

Pada desain antarmuka akan ditampilkan perancangan dari sistem yang akan digunakan pada saat input dan output klasifikasi. Berikut merupakan desain antarmuka dari sistem klasifikasi keluarga beresiko stunting pada kecamatan Merbau.



Gambar 3. 15 Desain Antarmuka

Penjelasan dari gambar 3.5 adalah sebagai berikut :

a) Halaman utama menampilkan tampilan awal pada sistem klasifikasi yang di dalam nya terdapat 2 menu, yaitu dataset dan hasil klasifikasi.

b) Dataset merupakan menu untuk menginput data yang akan di klasifikasi dalam bentuk excel.

c) Reset dataset merupakan tombol untuk menghapus dataset yang telah di unggah.

d) Upload dataset merupakan tombol untuk mengunggah dataset baru.

e) Hasil klasifikasi merupakan menu yang menampilkan hasil dari klasifikasi dataset yang telah di unggah.

f) Csv, excel dan pdf adalah jenis file yang bisa di pilih untuk di unduh

BAB IV
HASIL DAN PEMBAHASAN

4.1 Hasil Penelitian

Dalam penelitian ini melakukan perbandingan algoritma *machine learning* diantaranya yaitu *K-Nearest Neighbor, Decision Tree, Naïve Bayes,* dan *Random Forest* untuk mengetahui metode terbaik dalam melakukan klasifikasi potensi stunting keluarga pada Kecamatan Merbau.

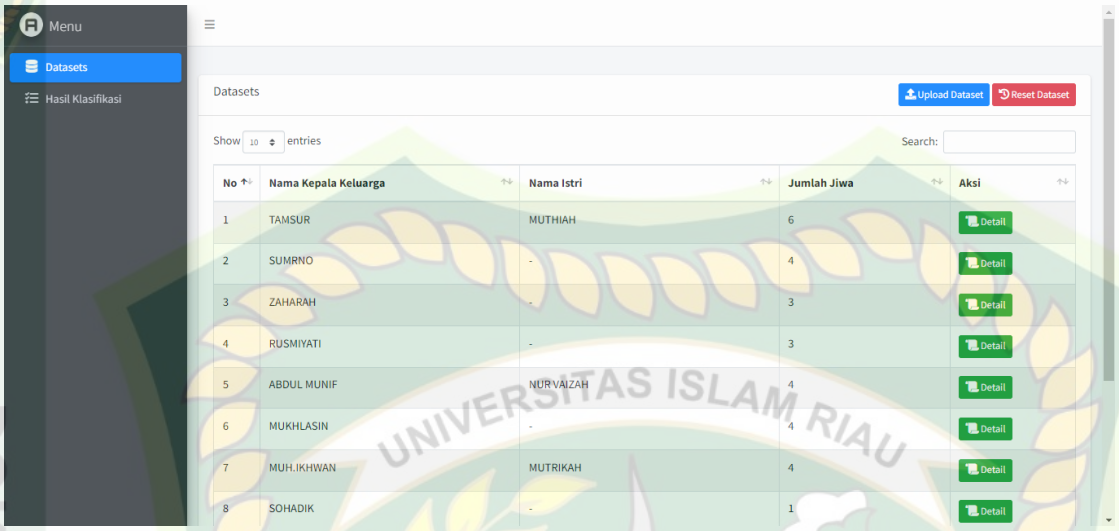
4.2 Pengujian Black Box

Pengujian *black box (black-box testing)* adalah metode pengujian perangkat lunak di mana sistem diuji tanpa pengetahuan internal tentang kode atau struktur internalnya. Dalam pengujian *black box,* pengujian dilakukan berdasarkan spesifikasi fungsional dan input-output yang diharapkan dari sistem, tanpa memperhatikan bagaimana sistem mencapai hasil tersebut.

Pengujian *black box* sering kali dilakukan dengan menggunakan berbagai teknik seperti pengujian fungsional, pengujian kebutuhan, pengujian perilaku, dan pengujian skenario. Fokus utama dari pengujian *black box* adalah memverifikasi bahwa sistem berfungsi sesuai dengan kebutuhan dan spesifikasi yang telah ditetapkan, tanpa perlu memahami implementasi internal dari sistem tersebut.

4.2.1 Halaman *Datasets*

Halaman *datasets* merupakan halaman utama yang pertama kali tampil ketika mengakses sistem klasifikasi, pada halaman ini, *staff* dapat melihat isi dari data lebih lanjut. *Staff* juga bisa menghapus dataset, atau menambahkan dataset baru yang ingin di klasifikasikan, file yang bisa diterima hanya dalam bentuk *xlsx,* dan tidak bisa menggunakan format lain.



Gambar 4. 1 Halaman *Datasets*

Tabel 4. 1 Skenario Pengujian pada Halaman *Datasets*

Bagian yang Diuji	Skenario Pengujian	Hasil yang Diinginkan	Hasil Pengujian
Button upload dokumen	Menambahkan dokumen bertipe selain xlsx	Sistem menolak dan menampilkan notifikasi "tipe file tidak sesuai, harap upload .xlsx"	[✓] Sesuai sasaran [] Tidak sesuai sasaran
	Menambahkan dokumen bertipe xlsx	Sistem menyimpan data dan menampilkannya pada tabel dataset	[✓] Sesuai sasaran [] Tidak sesuai sasaran
Button reset dataset	Menghapus semua dataset pada tabel dataset	Sistem menghapus semua data yang ada pada tabel dataset	[✓] Sesuai sasaran [] Tidak sesuai sasaran
Button detail	Tampil data sesuai baris	Sistem menampilkan semua data sesuai dari masing masing baris	[✓] Sesuai sasaran [] Tidak sesuai sasaran

4.2.2 Halaman Hasil Klasifikasi

Pada halaman hasil klasifikasi, *staff* bisa melihat hasil klasifikasi potensi keluarga beresiko stunting, juga berdasarkan kelas yang di inginkan, untuk mengunduh hasil klasifikasi dalam bentuk *pdf*, bisa dengan menekan tombol *pdf*, berlaku hal yang sama jika ingin mengunduh file bertipe *excel* dan *csv*.

Gambar 4. 2 Halaman Hasil Klasifikasi

Tabel 4. 2 Skenario Pengujian pada Halaman Hasil Klasifikasi

Bagian yang Diuji	Skenario Pengujian	Hasil yang Diinginkan	Hasil Pengujian
Filter Label	Menampilkan Dataset sesuai dengan pilihan pada filter label	Sistem menampilkan data sesuai urutan dari filter label	[✓] Sesuai sasaran [] Tidak sesuai sasaran
Button CSV	Mencetak dataset dalam bentuk CSV	Sistem berhasil mencetak dataset dalam bentuk CSV	[✓] Sesuai sasaran [] Tidak sesuai sasaran
Button Excel	Mencetak dataset dalam bentuk excel	Sistem berhasil mencetak dataset dalam bentuk excel	[✓] Sesuai sasaran [] Tidak sesuai Sasaran
Button PDF	Mencetak dataset dalam bentuk PDF	Sistem berhasil mencetak dataset dalam bentuk PDF	[✓] Sesuai Sasaran [] Tidak Sesuai sasaran

4.3 Pengujian Algoritma

Dalam penelitian ini pengujian *algoritma K-Nearest Neighbor, Decision Tree, Naïve Bayes,* dan *Random Forest,* dibutuhkan untuk mengetahui metode terbaik dalam melakukan klasifikasi potensi keluarga beresiko stunting pada data keluarga di Kecamatan Merbau.

4.3.1 Pengujian Klasifikasi *K-Nearest Neighbor*

Pada proses pengujian menggunakan algoritma KNN, data akan dibagi menjadi 2 data, yaitu *data training* dan *data testing*. Sebagian *Data training dan data testing* dapat dilihat pada tabel dibawah ini :

Tabel 4. 3 *Data Training*

No	No. KK	Var 1	Var 2	Var 3	Var 4	Var 5	Var 6	Var 7	Var 8	Var 9	Var 10	Var 11	Var 12	Var 13
1	14100520260300 001001001	46	39	0	3	1	2	2	7	2	0	1	3	2
2	14100520260200 001001001	58	0	0	3	3	0	0	7	1	3	3	3	0
3	14100520260300 001002002	66	0	0	3	3	0	0	7	3	3	3	3	0
4	14100520260200 001002002	53	0	0	3	3	0	0	7	3	3	3	3	0
5	14100520260200 001003003	43	34	0	3	1	2	2	7	1	0	0	3	2
.	.	.	.	.	.	.	.	.	.	.	.	.	.	.
86	14100520030100 002015017	53	0	0	3	3	0	0	7	2	3	3	3	0
87	14100520180200 001007008	57	0	0	3	3	0	0	7	2	3	3	3	0
88	14100520030100 002016018	62	30	0	3	1	2	2	7	2	0	0	3	2
89	14100520260100 001006008	43	38	0	3	1	2	1	7	1	0	1	3	2
90	14100520210300 001017020	40	34	0	1	1	2	1	7	2	0	0	3	2

Tabel 4. 4 *Data Testing*

No	No. KK	Var 1	Var 2	Var 3	Var 4	Var 5	Var 6	Var 7	Var 8	Var 9	Var 10	Var 11	Var 12	Var 13
91	14100520030400 001003004	29	27	0	1	1	2	1	7	3	0	0	3	2
92	14100520180200 001003004	27	21	0	1	1	2	1	7	2	0	0	3	2
93	14100520030200 002008011	42	0	0	3	3	0	0	1	1	3	3	3	0
94	14100520030400 001010012	35	36	0	3	1	2	1	5	3	0	1	3	2
95	14100510010005 001015025	67	0	0	3	3	0	0	4	1	3	3	3	0
96	14100520260300 001005006	39	40	1	1	1	2	1	1	2	0	1	3	2
97	14100520260100 001001001	60	0	0	3	3	0	0	7	1	3	3	3	0
98	14100520260200 001004005	28	27	0	3	1	2	2	7	1	0	0	3	2
99	14100510010004 003036036	56	0	0	3	3	0	0	1	1	3	3	3	0
100	14100510010002 004004002	79	0	0	3	3	0	0	1	1	3	3	3	0

Keterangan dari variabel :

Inisiasi	Variabel
Var 1	umur_kepala_keluarga
Var 2	umur_istri
Var 3	jumlah_baduta
Var 4	jumlah_balita
Var 5	pus_kb
Var 6	pus_hamil_kb3
Var 7	ber_kb_kb4
Var 8	tidak_punya_sumber_air_minum_layak
Var 9	tidak_punya_jamban_layak
Var 10	istri_terlalu_muda
Var 11	istri_terlalu_tua
Var 12	usia_anak_terlalu_dekat
Var 13	terlalu_banyak_anak

Setelah mendapatkan data training dan data testing, langkah selanjutnya melakukan tahapan yang digunakan dalam KNN :

1) Menghitung Jarak (*Euclidean Distance*)

Misalnya Menghitung Jarak Data ke-1:

$$d(x,y) = \sqrt{\begin{matrix}(46-40)^2 + (39-34)^2 + (0-0)^2 + (3-1)^2 + (1-1)^2 + \\ (2-2)^2 + (2-1)^2 + (7-7)^2 + (2-2)^2 + (0-0)^2\end{matrix}}$$

$$d(x,y) = 8{,}185352772$$

Hasil Perhitungan Euclidean Distance:

Tabel 4. 5 Hasil Perhitungan *Euclidean Distance*

No	No. KK	Var 1	Var 2	Var 3	Var 4	Var 5	Var 6	Var 7	Var 8	Var 9	Var 10	Var 11	Var 12	Var 13	Jarak (Euclidean Distance)	Rank
1	14100520260300001001001	46	39	0	3	1	2	2	7	2	0	1	3	2	8,185352772	73
2	14100520260200001001001	58	0	0	3	3	0	0	7	1	3	3	3	0	38,93584467	31
3	14100520260300001002002	66	0	0	3	3	0	0	7	3	3	3	3	0	43,22036557	14
4	14100520260200001002002	53	0	0	3	3	0	0	7	3	3	3	3	0	36,89173349	43
5	14100520260200001003003	43	34	0	3	1	2	2	7	1	0	0	3	2	3,872983346	86
6	14100520260200001004004	52	0	0	3	3	0	0	7	1	3	3	3	0	36,55133376	46
7	14100520260300001003004	39	33	0	1	1	2	2	1	3	0	0	3	2	6,32455532	78
.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.
84	14100520030100002007008	61	0	0	3	3	0	0	7	1	3	3	3	0	40,4103947	25
85	14100520030100002014016	47	37	0	3	1	2	2	1	2	0	1	3	2	10	71
86	14100520030100002015017	53	0	0	3	3	0	0	7	2	3	3	3	0	36,87817783	45
87	14100520180200001007008	57	0	0	3	3	0	0	7	2	3	3	3	0	38,47076812	38
88	14100520030100002016018	62	30	0	3	1	2	2	7	2	0	0	3	2	22,47220505	57
89	14100520260100001006008	43	38	0	3	1	2	1	7	1	0	1	3	2	5,567764363	82
90	14100520210300001017020	40	34	0	1	1	2	1	7	2	0	0	3	2	0	90

2) Menentukan Nilai K (Tetangga Terdekat)

Menghitung jarak *data training* terhadap *data testing*

Tabel 4. 6 Hasil Klasifikasi Data Testing

No	Jarak (Euclidean Distance) Data 91	Rank Data 91	Jarak (Euclidean Distance) Data 92	Rank Data 92	Jarak (Euclidean Distance) Data 93	Rank Data 93	Jarak (Euclidean Distance) Data 94	Rank Data 94	Jarak (Euclidean Distance) Data 95	Rank Data 95
1	20,97617696	66	26,28687886	61	40,03748244	12	11,66190379	70	44,73253849	18
2	40,11234224	29	37,92097045	30	17,08800749	65	43,11612227	29	9,486832981	65
3	46,18441296	15	44,69899328	14	24,81934729	48	47,8225888	15	3,741657387	81
4	36,60601044	45	33,95585369	46	12,68857754	77	40,62019202	45	14,45683229	48
5	15,93737745	72	20,76053949	72	35,02855978	26	8,831760866	75	42,13074887	30
6	36,01388621	46	33,19638535	49	11,66190379	81	40,23679908	46	15,29705854	47
7	13,15294644	74	18,05547009	75	33,76388603	29	6,8556546	82	43,8634244	25
.	.	.	.	.	.	.	.	.	.	.
85	21,61018278	65	26,41968963	60	37,73592453	20	12,76714533	69	42,52058325	29
86	36,61966685	44	33,9411255	48	12,56980509	79	40,63249931	44	14,35270009	50
87	39,35733731	37	37,09447398	38	16,18641406	70	42,55584566	37	10,48808848	59
88	33,22649545	50	36,20773398	40	37,02701716	21	27,78488798	56	31,12876483	37
89	18,05547009	69	23,47338919	67	38,82009789	15	8,717797887	76	45,33210783	16
90	13,07669683	75	18,38477631	74	35,09985755	25	6,244997998	84	43,93176527	24

Menentukan kelompok data hasil uji berdasarkan label mayoritas dari K tetangga terdekat. Karena nilai K = 3, maka diambil 3 jarak terkecil, pengurutan dari hasil perhitungan. Jarak diurutkan dari yang paling dekat jaraknya ke yang paling jauh. Setelah diurutkan diperoleh sebagai berikut:

Tabel 4. 7 Hasil Klasifikasi K=3

Data Testing	Nilai K=3			Hasil Klasifikasi
	Jarak (Euclidean Distance)	Dokumen	Label Aktual Dokumen	
91	70,66116331	68	2	2
	60,69596362	32	2	
	58,05170109	23	2	
92	70,46985171	68	2	2
	60,10823571	32	2	
	57,33236433	23	2	
93	52,34500931	68	2	2
	48,35286961	12	2	
	47,08502947	50	2	
94	69,36137254	68	2	2
	60,28266749	32	2	
	57,92236183	18	2	
95	50,59644256	83	2	2
	50,35871325	12	2	
	50,14977567	50	2	
96	68,50547423	68	2	2
	60,03332408	32	2	
	57,87054518	18	2	
97	14,59451952	12	2	2
	12,64911064	50	2	
	10,58300524	34	2	
98	71,54718723	68	2	2
	61,54673021	32	2	
	58,88972746	23	2	
99	48,06245936	12	2	2
	47,38143096	50	2	
	46,80811895	34	2	
100	61,92737682	83	2	2
	61,24540799	52	2	
	58,81326381	24	2	

3) Hasil Klasifikasi KNN

Hasil klasifikasi KNN menggunakan nilai K=3 pada data testing data 91 sampai data 100 menghasilkan klasifikasi yaitu 2 (Rendah) atau keluarga dapat dikatakan berisiko Rendah Stunting.

Tabel 4. 8 Hasil Klasifikasi KNN

No	Klasifikasi	Aktual Label
91	2	2
92	2	2
93	2	1
94	2	3
95	2	1
96	2	2
97	2	2
98	2	2
99	2	1
100	2	1

Pada tabel di atas dapat dilihat bahwa dari 10 sampel yang diuji, terdapat 5 data yang diprediksi belum benar. Seharusnya, algoritma tersebut memprediksi sesuai dengan aktual label, akan tetapi secara kenyataan belum semuanya terpenuhi. Pada tabel tersebut juga dapat disimpulkan bahwa algoritma condong memprediksi label 2, hal tersebut dipengaruhi dari dataset yang didapatkan karena lebih banyak data berlabel 2 dibandingkan dengan label yang lain.

4) Output Confusion Matrix

Berdasarkan pengujian yang dilakukan, dapat dihitung nilai accuracy menggunakan persamaan 2.8, nilai precision menggunakan persamaan 2.9, nilai recall menggunakan persamaan 2.10, dan nilai F1-Score menggunakan persamaan 2.11.

Tabel 4. 9 *Confusion Matrix* Algoritma KNN

Aktual \ *Predicted*	Tinggi	Rendah	Tidak
Tinggi	6	0	0
Rendah	4	85	1
Tidak	0	0	4

$$Accuracy = \frac{6 + 85 + 4}{6 + 4 + 85 + 0 + 4 + 1}$$

$$= \frac{6 + 85 + 4}{6 + 4 + 85 + 0 + 4 + 1}$$

$$= \frac{95}{100}$$

$$= 0{,}95$$

$$Precision = \frac{\left(\frac{6}{6+0} + \frac{85}{85+5} + \frac{4}{4+0}\right)}{3}$$

$$= \frac{\left(\frac{6}{6} + \frac{85}{90} + \frac{4}{4}\right)}{3}$$

$$= \left(\frac{53}{18}\right) / 3$$

$$= 0{,}981$$

$$Recall = \frac{\left(\frac{6}{10} + \frac{85}{85} + \frac{4}{5}\right)}{3}$$

$$Recall = \left(\frac{12}{5}\right) / 3$$

$$Recall = 0{,}8$$

$$F1\ Score = \frac{(2 *\ 0.6 * 1)}{(0.6 + 1)} + \frac{(2 *\ 1 * 0.94)}{(1 + 0.94)} + \frac{(2 *\ 0.8 * 1)}{(0.8 + 1)}$$

$$F1\ Score = \frac{0.75 + 0.9691 + 0.8889}{3}$$

$$F1\ Score = 0.8693$$

Dilihat dari perhitungan *confusion matrix* di atas, *accuracy* yang didapatkan sebesar 95%, *precision* 98.1%, *recall* 80%, dan *f1-score* sebesar 86.93% dimana dapat dikatakan algoritma yang dibangun sangat baik untuk memprediksi data stunting. Algoritma KNN mendapatkan hasil akurasi yang sangat baik karena karakteristik dari KNN sendiri mampu menangani data yang tidak terlalu banyak (Cholil et al., 2021). Tidak hanya berdasarkan data saja, melainkan juga dari pemilihan nilai K yang terbaik juga dapat mempengaruhi tingkat akurasi yang baik pula (Angreni et al., 2019).

4.3.2 Pengujian Klasifikasi *Decision Tree*

Dalam proses pengujian menggunakan algoritma *Decision Tree*, dilakukan pencarian Gain dan Entrophy, untuk membentuk struktur seperti pohon (*tree*), yang disebut sebagai pohon keputusan.

1) Mencari Nilai S dan Nilai Si

   Perhitungan Gain dan Entrophy Data Training:

   Rumus Entrophy:

$$Entropy(S) = \sum_{i=0}^{n} -pi$$

   Keterangan :

   S = himpunan kasus

   n = Jumlah partisi S

   pi = proporsi Si terhadap S

   Rumus Gain

$$Gain(S, A) = Entropy(S) - \sum_{i=1}^{n} \frac{Si}{S} * Entropy(Si)$$

Keterangan:

S = himpunan kasus

A = atribut

n = jumlah partisi atribut A

Diketahui:

S = 90

Si Tinggi (3) = 4

Si Rendah (2) = 80

Si Tidak (1) = 6

Misalnya Menghitung Nilai Entrophy Total::

$$Entrophy\ Total = (\left(-\frac{4}{90}\right) * IMLOG2\left(\frac{4}{90}\right) + \left(-\frac{80}{90}\right) * IMLOG2\left(\frac{80}{90}\right) + \left(-\frac{6}{90}\right) * IMLOG2\left(\frac{6}{90}\right))$$

$$Entrophy\ Total = 0{,}611141734$$

Misalnya Menghitung Nilai Entrophy Var 1 (umur_kepala_keluarga):

$$Entrophy(Var\ 1) = (\left(-\frac{0}{1}\right) * IMLOG2\left(\frac{0}{1}\right) + \left(-\frac{1}{1}\right) * IMLOG2\left(\frac{1}{1}\right) + \left(-\frac{0}{1}\right) * IMLOG2\left(\frac{0}{1}\right))$$

$$Entrophy(Var\ 1) = 0$$

Misalnya Menghitung Nilai Gain Var 1 (umur_kepala_keluarga):

$$Gain = (0{,}611141734) - \left(\left(\frac{1}{90}\right) * 0\right) + \left(\left(\frac{1}{90}\right) * 0\right), \ldots + \left(\left(\frac{1}{90}\right) * 0\right)$$

$$Gain = 0{,}611141734$$

Hasil Perhitungan Entrophy dan Gain Data Training:

Tabel 4. 10 Hasil Perhitungan *Entrophy* dan *Gain Data Training*

		Jumlah (S)	Si Tinggi (3)	Si Rendah (2)	Si Tidak (1)	Entrophy	Gain
Total		90	4	80	6	0,611141734	
Var 1							0,611141734
	21	1	0	1	0	0	
	22	1	0	1	0	0	
	…	…	…	…	…	…	
	83	1	0	1	0	0	
	94	1	0	1	0	0	
Var 2							0,611141734
	0	55	0	49	6	0	
	…	…	…	…	…	…	
	47	1	0	1	0	0	
Var 3							0,005766784
	0	87	4	77	6	0,626249948	
	1	3	0	3	0	0	
Var 4							1,148146489
	1	12	1	11	0	0	
	3	77	3	68	6	0,627667895	
Var 5							0,611141734
	1	35	4	31	0	0	
	2	0	0	0	0	0	
	3	55	0	49	6	0	
Var 6							0,611141734
	0	55	0	49	6	0	
	2	33	4	29	0	0	
Var 7							0,611141734
	0	57	0	51	6	0	
	2	14	1	13	0	0	
Var 8							0,611141734
	1	7	0	3	4	0	
	7	78	3	75	0	0	
Var 9							0,611141734
	1	49	0	43	6	0	
	4	1	0	1	0	0	
Var 10							0,611141734
	0	34	3	31	0	0	
	3	55	0	49	6	0	
Var 11							0,611141734
	0	18	1	17	0	0	
	3	55	0	49	6	0	

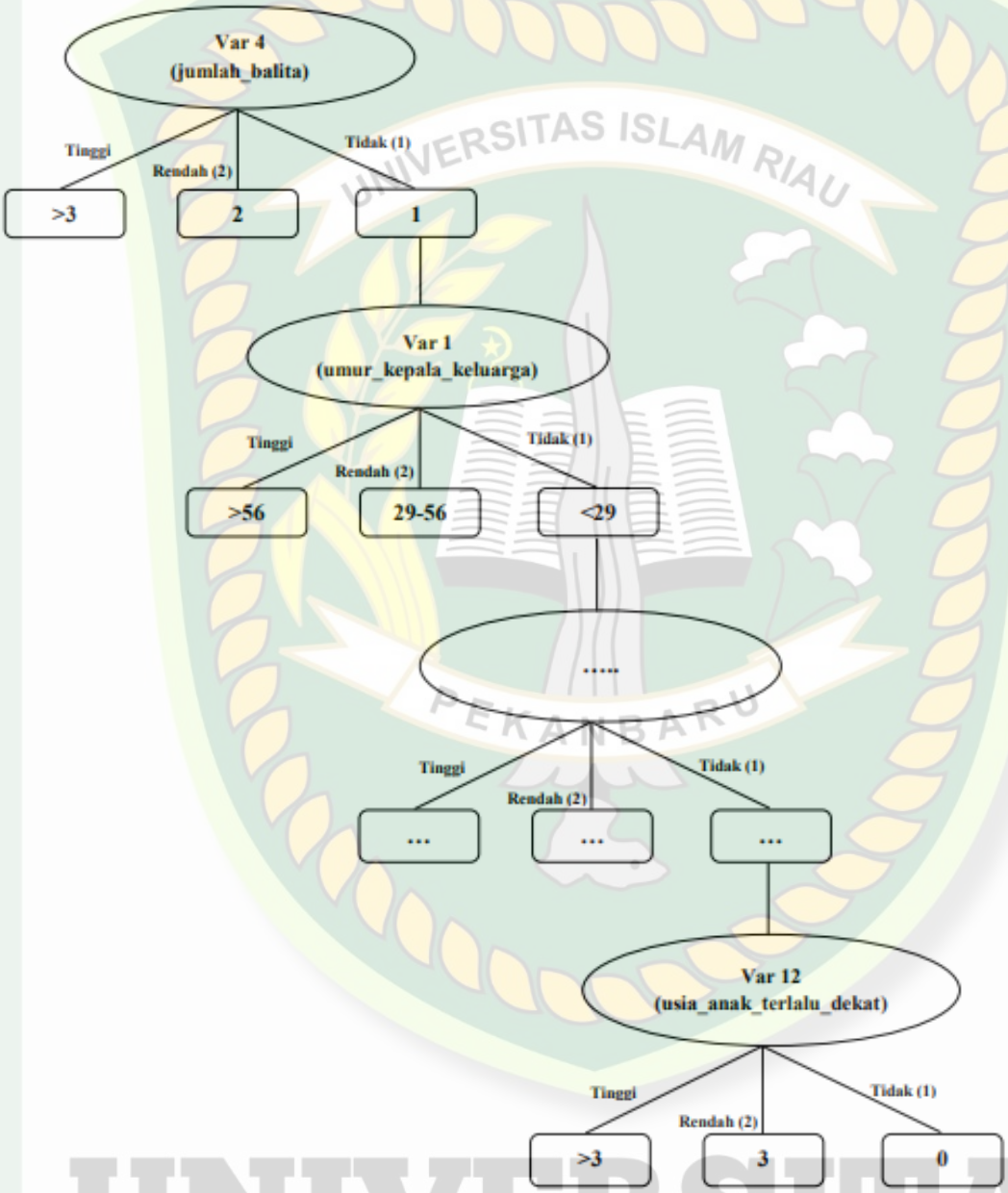
Var 12							0
	3	90	4	80	6	0,611141734	
Var 13							0,611141734
	0	55	0	49	6	0	
	2	33	4	29	0	0	
Max Gain							**1,148146489**

Hasil Perhitungan Gain dan Entrophy Data Testing:

Tabel 4. 11 Hasil Perhitungan *Gain* dan *Entrophy Data Testing*

		Jumlah (S)	**Si Tinggi (3)**	**Si Rendah (2)**	**Si Tidak (1)**	**Entrophy**	**Gain**
Total		10	1	5	4	1,360964047	
Var 1							0,611141734
	27	1	0	1	0	0	
	…	…	…	…	…	…	
	79	1	0	0	1	0	
Var 2							0,611141734
	0	5	0	1	4	0	
	…	…	…	…	…	…	
	40	1	0	1	0	0	
Var 3							0,471927012
	0	9	1	4	4	1,392147224	
	1	1	0	1	0	0	
Var 4							0,71838045
	1	3	0	3	0	0	
	3	7	1	2	4	1,378783493	
Var 5							0,611141734
	1	5	1	4	0	0	
	3	5	0	1	4	0	
Var 6							0,611141734
	0	5	0	1	4	0	
	2	5	1	4	0	0	
Var 7							0,611141734
	0	5	0	1	4	0	
	2	1	0	1	0	0	
Var 8							0,611141734
	1	4	0	1	3	0	
	7	4	0	4	0	0	
Var 9							0,611141734
	1	6	0	2	4	0	
	3	2	1	1	0	0	
Var 10							0,611141734
	0	5	1	4	0	0	
	3	5	0	1	4	0	
Var 11							0,611141734
	0	3	0	3	0	0	
	3	5	0	1	4	0	

Var 12							0,459923507
	3	10	1	5	4	1,360964047	
Var 13							0,611141734
	0	5	0	1	4	0	
Max Gain							**0,71838045**



Gambar 4. 3 Pohon Keputusan *Decision Tree*

Jadi dari hasil keputusan (*tree*) yang menjadi akar adalah Var 4 (jumlah_balita) dengan nilai 1 sebagai label Tidak (1), nilai 3 sebagai label Rendah (2), nilai >3 sebagai label Tinggi (3). Kemudian variabel yang mempengaruhi yaitu Var 1 (umur_kepala_keluarga), Var 2 (umur_istri), Var 5 (pus_kb), Var 6 (pus_hamil_kb3), Var 7 (ber_kb_kb4), Var 8 (tidak_punya_sumber_air_minum_layak), Var 9 (tidak_punya_jamban_layak), Var 10 (istri_terlalu_muda), Var 11 (istri_terlalu_tua), Var 13 (terlalu_banyak_anak), Var 3 (jumlah_baduta), Var 12 (usia_anak_terlalu_dekat).

Misalnya pengecekan keputusan pada data ke 91 yaitu diperoleh bahwa dicek pada var 4 dengan nilai data 1 maka masuk dalam label Tidak (1). Kemudian dicek kembali pada var 1 dengan nilai data 29 maka masuk dalam label Tidak (1). Masuk pengecekan var 2 dengan nilai data maka masuk ke dalam label Rendah (2). Proses keputusan pada data ke 91 yaitu pada var 2 menghasilkan label Rendah (2).

2) Hasil Klasifikasi *Decision Tree*

Hasil klasifikasi *Decision Tree* pada data testing data 91 sampai data 100 menghasilkan klasifikasi yaitu 1 (Stunting) sesuai dengan class aktualnya.

Tabel 4. 12 Hasil Klasifikasi *Decision Tree*

No	No. KK		Klasifikasi Decision Tree	Aktual Label
91	14100520030400	001003004	1	2
92	14100520180200	001003004	1	2
93	14100520030200	002008011	1	1
94	14100520030400	001010012	1	3
95	14100510010005	001015025	1	1
96	14100520260300	001005006	1	2
97	14100520260100	001001001	1	2
98	14100520260200	001004005	1	2
99	14100510010004	003036036	1	1
100	14100510010002	004004002	1	1

Pada tabel di atas dapat dilihat bahwa algoritma *Decision Tree* tidak berbeda jauh dengan KNN karena memiliki hasil prediksi yang konsisten yaitu label 1.

3) Output Confusion Matrix

Berdasarkan pengujian yang dilakukan, dapat dihitung nilai accuracy menggunakan persamaan 2.8, nilai precision menggunakan persamaan 2.9, nilai recall menggunakan persamaan 2.10, dan nilai F1-Score menggunakan persamaan 2.11.

Tabel 4. 13 *Confusion Matrix* Algoritma *Decision Tree*

Predicted / **Aktual**	**Tinggi**	**Rendah**	**Tidak**
Tinggi	6	0	0
Rendah	4	85	1
Tidak	0	0	4

$$Accuracy = \frac{6 + 85 + 4}{6 + 4 + 85 + 0 + 4 + 1}$$

$$= \frac{6 + 85 + 4}{6 + 4 + 85 + 0 + 4 + 1}$$

$$= \frac{95}{100}$$

$$= 0{,}95$$

$$Precision = \frac{\left(\frac{6}{6+0} + \frac{85}{85+5} + \frac{4}{4+0}\right)}{3}$$

$$= \frac{\left(\frac{6}{6} + \frac{85}{90} + \frac{4}{4}\right)}{3}$$

$$= \left(\frac{53}{18}\right) / 3$$

$$= 0{,}981$$

$$Recall = \frac{\left(\frac{6}{10} + \frac{85}{85} + \frac{4}{5}\right)}{3}$$

$$= \left(\frac{12}{5}\right) / 3$$

$$= 0{,}8$$

$$F1\ Score = \frac{(2 * 0.6 * 1)}{(0.6 + 1)} + \frac{(2 * 1 * 0.94)}{(1 + 0.94)} + \frac{(2 * 0.8 * 1)}{(0.8 + 1)}$$

$$= \frac{0.75 + 0.9691 + 0.8889}{3}$$

$$= 0.8693$$

Dilihat dari perhitungan *confusion matrix* di atas, *accuracy* yang didapatkan sebesar 95%, *precision* 98.1%, *recall* 80%, dan *f1-score* sebesar 86.93% dimana dapat dikatakan algoritma yang dibangun sangat baik untuk memprediksi data stunting. Algoritma *Decision Tree* mendapatkan hasil *accuracy* yang sangat baik karena algoritma *Decision Tree* memiliki struktur sederhana sehingga merupakan salah satu algoritma tercepat untuk mengidentifikasi variabel signifikan dan hubungan antar dua variabel (Upadhyay et al., 2021).

4.3.3 Pengujian Klasifikasi *Naïve Bayes*

Pada proses pengujian menggunakan algoritma *Naïve Bayes* dengan metode probabilitas dan statistic yang ditemukan oleh ilmuan Inggris Thomas Bayes, yaitu memprediksi peluang dimasa yang akan datang berdasarkan pengalaman dimasa sebelumnya. Teorema tersebut dikombinasikan dengan naïve dimana asumsi kondisi antar variabel yang saling bebas. Klasifikasi naïve bayes mengasumsikan bahwa ada atau tidak ciri tertentu dari sebuah kelas tidak ada kaitannya dengan ciri dari kelas lainnya.

Rumus teorema bayes :

$$P(H|X) = \frac{P(X|H) * P(H)}{P(X)}$$

Keterangan:

X = data dengan kelas diketahui

H = Hipotesis data X adalah suatu kelas spesifik

P(H|X) = Probabilitas hipotesis H sesuai kondisi X (posterior probability).

P(H) = Probabilitas hipotesis H (prior probability)

P(X|H) = Probabilitas X sesuai kondisi terhadap hipotesis H

P(X) = Probabilitas X

Rumus distribusi Gaussian Naive Bayes :

$$P(X_i) = x_i|Y = y_i = \frac{1}{\sqrt{2\pi\sigma ij}} \exp^{-\frac{(xi-\mu ij)^2}{2\sigma^2 ij}}$$

Keterangan :

P = Peluang

xi = Nilai atribut ke - i

yi = kelompok ke - i

π j = rata-rata kelompok ke - j

σ^2j = variansi kelompok ke – j

Parameter π j bisa didapat dari mean sampel xi (x) dari semua data latih yang menjadi milik kelas yi, sedangkan σ^2j dapat diperkirakan dari variansi sampel (s2) dari data latih.

Perhitungan Prior atau P(c) :

$$P(c) = \frac{Nc}{N}$$

Dimana:

P(c) = Prior dari kelas c

Nc = Banyaknya dokumen pada data latih yang menjadi kategori c

N = Banyaknya dokumen pada data latih

1) Menghitung Probabilitas Class

Diketahui jumlah dokumen yang digunakan untuk data training yaitu 90 data dan data testing yaitu 10 data.

Tabel 4. 14 *N Class Data Training* dan *Data Testing*

Nama Label	N Class
Class Tinggi	4
Class Rendah	80
Class Tidak	6
∑Doc Training	**90**
Class Tinggi	1
Class Rendah	5
Class Tidak	4
∑Doc Testing	**10**

Misalnya menghitung Probabilitas Class 3 (Tinggi):

$$P(Tinggi) = \frac{4}{90}$$

$$P(Tinggi) = 0{,}044444444$$

Misalnya menghitung Probabilitas Class 2 (Rendah):

$$P(Rendah) = \frac{80}{90}$$

$$P(Rendah) = 0{,}888888889$$

Misalnya menghitung Probabilitas Class 1 (Tidak):

$$P(Tidak) = \frac{6}{90}$$

$$P(Tidak) = 0{,}066666667$$

2) Menghitung Teorema Bayes

Misalnya Menghitung Teorema Bayes untuk Fitur Pertama Field 21 pada Class Tinggi:

$$P(w|3) = \frac{0 * 0}{4}$$

$$P(w|3) = \frac{0}{4}$$

$$P(w|3) = 0$$

Misalnya Menghitung Teorema Bayes untuk Fitur Pertama Field 21 pada Class Rendah:

$$P(w|2) = \frac{1 * 1}{80}$$

$$P(w|2) = \frac{1}{80}$$

$$P(w|2) = 0{,}0125$$

Misalnya Menghitung Teorema Bayes untuk Fitur Pertama Field 21 pada Class Tidak:

$$P(w|1) = \frac{0 * 0}{6}$$

$$P(w|1) = \frac{0}{6}$$

$$P(w|1) = 0$$

Hasil Perhitungan Conditional Probabilitas Feature atau Variabel :

Tabel 4. 15 Hasil Perhitungan *Conditional Probabilitas Feature* atau *Variabel*

Variabel	Field	Conditional Probabilitas		
		P(W|3)	P(W|2)	P(W|1)
Var 1	21	0	0,0125	0
	22	0	0,0125	0
	…	…	…	…
	94	0	0,0125	0
Var 2	0	0	0,6125	1
	..	…	…	…
	47	0	0,0125	0
Var 3	0	1	0,9625	1
	1	0	0,0375	0
Var 4	1	0,25	0,1375	0
	2	0	0,0125	0
	3	0,75	0,85	1
Var 5	1	1	0,3875	0
	2	0	0	0
	3	0	0,6125	1
Var 6	0	0	0,6125	1
	1	0	0,025	0
	2	1	0,3625	0
Var 7	0	0	0,6375	1
	1	0,75	0,2	0
	2	0,25	0,1625	0
Var 8	1	0	0,0375	0,666666667
	4	0	0	0,333333333
	5	0,25	0,0125	0
	6	0	0,0125	0
	7	0,75	0,9375	0
Var 9	1	0	0,5375	1
	2	0,5	0,275	0
	3	0,5	0,175	0
	4	0	0,0125	0
Var 10	0	0,75	0,3875	0
	1	0,25	0	0
	3	0	0,6125	1
Var 11	0	0,25	0,2125	0
	1	0,75	0,175	0
	3	0	0,6125	1
Var 12	3	1	1	1
Var 13	0	0	0,6125	1
	1	0	0,025	0
	2	1	0,3625	0

Untuk fitur bertipe numerik (kontinu) yaitu contohnya pada var 1 dengan nilai field 40,41,42,3, dst data field cenderung kontinu atau berkelanjutan. Distribusi Gaussian biasanya dipilih untuk merepresentasikan probabilitas bersyarat dari fitur kontinu pada sebuah kelas P(xi|Y), sedangkan distribusi Gaussian dikarakteristikkan dengan dua parameter: mean, m , dan variansi, 2 s . Untuk setiap kelas yi, probabilitas bersyarat kelas yi untuk fitur xi adalah.

3) Menghitung Mean dan Standard Deviasi

Langkah pertama memisahkan data Field masing-masing label:

a) Data Dengan Label Tinggi

Tabel 4. 16 Data Dengan Label Tinggi

No	No. KK	Var 1	Var 2	Var 3	Var 4	Var 5	Var 6	Var 7	Var 8	Var 9	Var 10	Var 11	Var 12	Var 13
1	14100520260300 001001001	46	39	0	3	1	2	2	7	2	0	1	3	2
38	14100520210300 001008009	41	39	0	3	1	2	1	7	2	0	1	3	2
70	14100520030400 001007008	32	19	0	1	1	2	1	7	3	1	0	3	2
73	14100520030400 001008009	42	40	0	3	1	2	1	5	3	0	1	3	2

b) Data Dengan Label Rendah

Tabel 4. 17 Data Dengan Label Rendah

No	No. KK	Var 1	Var 2	Var 3	Var 4	Var 5	Var 6	Var 7	Var 8	Var 9	Var 10	Var 11	Var 12	Var 13
2	14100520260200 001001001	58	0	0	3	3	0	0	7	1	3	3	3	0
3	14100520260300 001002002	66	0	0	3	3	0	0	7	3	3	3	3	0
4	14100520260200 001002002	53	0	0	3	3	0	0	7	3	3	3	3	0
…	…	…	…	…	…	…	…	…	…	…	…	…	…	…
88	14100520030100 002016018	62	30	0	3	1	2	2	7	2	0	0	3	2
89	14100520260100 001006008	43	38	0	3	1	2	1	7	1	0	1	3	2
90	14100520210300 001017020	40	34	0	1	1	2	1	7	2	0	0	3	2

c) Data Dengan Label Tidak

Tabel 4. 18 Data Dengan Label Tidak

No	No. KK	Var 1	Var 2	Var 3	Var 4	Var 5	Var 6	Var 7	Var 8	Var 9	Var 10	Var 11	Var 12	Var 13
15	14100510010005 001017028	50	0	0	3	3	0	0	4	1	3	3	3	0
16	14100510010005 001025039	68	0	0	3	3	0	0	4	1	3	3	3	0
17	14100520030200 003021024	54	0	0	3	3	0	0	1	1	3	3	3	0
64	14100520260300 003056069	57	0	0	3	3	0	0	1	1	3	3	3	0
65	14100520030200 002008010	65	0	0	3	3	0	0	1	1	3	3	3	0
80	14100520030200 002014017	58	0	0	3	3	0	0	1	1	3	3	3	0

Langkah kedua menghitung nilai Mean terhadap Field masing-masing label:

Sehingga dari hasil memisahkan data Field terhadap masing-masing label, berikut hasil nilai Mean dan Standar Deviasi masing-masing label:

d) Hasil Nilai Mean

Tabel 4. 19 Hasil Nilai Mean

No	Label	Var 1	Var 2	Var 3	Var 4	Var 5	Var 6	Var 7	Var 8	Var 9	Var 10	Var 11	Var 12	Var 13
1	Tinggi (3)	40	34	0	2,5	1	2	1,25	6,5	2,5	0,25	0,75	3	2
2	Rendah (2)	53	13	0,03	2,71	2,22	0,75	0,52	6,74	1,66	1,83	2,01	3	0,75
3	Tidak (1)	59	0	0	3	3	0	0	2	1	3	3	3	0

e) Hasil Nilai Standar Deviasi

Tabel 4. 20 Hasil Nilai Standar Deviasi

No	Label	Var 1	Var 2	Var 3	Var 4	Var 5	Var 6	Var 7	Var 8	Var 9	Var 10	Var 11	Var 12	Var 13
1	Tinggi (3)	5,90	10,18	0	1	0	0	0,5	1	0,57	0,5	0,5	0	0
2	Rendah (2)	14,74	17,55	0,19	0,70	0,98	0,96	0,76	1,166	0,81	1,47	1,28	0	0,96
3	Tidak (1)	6,74	0	0	0	0	0	0	1,54	0	0	0	0	0

4) Menghitung Distribusi Gaussian Naïve Bayes

Misalnya menghitung distribusi gaussian pada data ke 91 pada label Tinggi terhadap Var 1:

$$P(X_i) = x_i|Y = y_i = \frac{1}{\sqrt{2 * 3{,}14 * 5{,}909032634}} \exp - \frac{(29-40{,}25)^2}{2*5{,}909032634^2}$$

$$P(X_i) = x_i|Y = y_i = 0{,}02680205$$

Hasil perhitungan distribusi gaussian:

Tabel 4. 21 Hasil Perhitungan Distribusi Gaussian

No	Label	Var 1	Var 2	Var 3	Var 4	Var 5	Var 6	Var 7	Var 8	Var 9	Var 10	Var 11	Var 12	Var 13
91	Tinggi (3)	0,027	0,097	0	0,130	0	0	0,50	0,35	0,36	0,50	0,18	0	0
	Rendah (2)	0,028	0,07	0,89	0,02	0,18	0,17	0,37	0,36	0,11	0,15	0,10	0	0,17
	Tidak (1)	9,62	0	0	0	0	0	0	0,0017	0	0	0	0	0
…	…	…	…	…	…	…	…	…	…	…	…	…	…	…
100	Tinggi (3)	7,53	0,00043	0	0,35	0	0	0,025	1,078	0,018	1,52	2,26	0	0
	Rendah (2)	0,02	0,07	0,89	0,43	0,30	0,30	0,36	2,059	0,32	0,24	0,26	0	0,30
	Tidak (1)	0,0016	0	0	0	0	0	0	0,26	0	0	0	0	0

5) Hasil Klasifikasi *Naïve Bayes*

Misalnya menghitung posterior data ke 91 pada label Tinggi:

$$P(d|c) = (0{,}02680205 * 0{,}097052818 * 0{,}129550438 * 0{,}498021814 * 0{,}352154602 * 0{,}360944195 * 0{,}498021814 * 0{,}183211987) * (0{,}044444444)$$

$$P(d|c) = 8{,}65081E-08 \; atau \; 0{,}0000000865081$$

Hasil perhitungan posterior:

Tabel 4. 22 Hasil Klasifikasi *Naïve Bayes*

No	No. KK	Nilai Posterior	Nilai Max	Hasil Klasifikasi	Aktual Label
91	Tinggi (3)	8,65	8,65	3	2
	Rendah (2)	4,98			
	Tidak (1)	1,12			
92	Tinggi (3)	2,36	2,3685E-08	3	2
	Rendah (2)	1,74			
	Tidak (1)	2,91			
93	Tinggi (3)	1,76	0,00012	1	1
	Rendah (2)	7,77			
	Tidak (1)	0,00012			
94	Tinggi (3)	1,23	1,23	3	3
	Rendah (2)	8,26			
	Tidak (1)	1,06			
95	Tinggi (3)	1,06	0,00066	1	1
	Rendah (2)	7,37			
	Tidak (1)	0,00066			
96	Tinggi (3)	4,733	3,79	1	2
	Rendah (2)	8,28			
	Tidak (1)	3,79			
97	Tinggi (3)	2,26	1,76	1	2
	Rendah (2)	1,58			
	Tidak (1)	1,76			
98	Tinggi (3)	3,07	3,07	3	2
	Rendah (2)	4,37			
	Tidak (1)	5,79			
99	Tinggi (3)	5,29	0,0024	1	1
	Rendah (2)	9,98			
	Tidak (1)	0,0024			
100	Tinggi (3)	8,47	2,82	1	1
	Rendah (2)	2,12			
	Tidak (1)	2,82			

Pada tabel di atas dapat dilihat bahwa algoritma *naive bayes* dapat memprediksi data stunting secara baik. Berbeda dengan algoritma KNN dan *decision tree* sebelumnya, hasil prediksi dari algoritma ini lebih bervariatif.

6) Output Confusion Matrix

Berdasarkan pengujian yang dilakukan, dapat dihitung nilai accuracy menggunakan persamaan 2.8, nilai precision menggunakan persamaan 2.9, nilai recall menggunakan persamaan 2.10, dan nilai F1-Score menggunakan persamaan 2.11.

Tabel 4. 23 *Confusion Matrix* Algotitma *Naïve Bayes*

Predicted / **Aktual**	**Tinggi**	**Rendah**	**Tidak**
Tinggi	10	2	0
Rendah	0	80	0
Tidak	0	3	5

$$Accuracy = \frac{10 + 80 + 5}{10 + 0 + 80 + 5 + 5 + 0}$$

$$= \frac{95}{100}$$

$$= 0{,}95$$

$$Precision = \frac{\left(\frac{10}{10+2} + \frac{80}{80+0} + \frac{3}{3+5}\right)}{3}$$

$$= \frac{\left(\frac{10}{12} + \frac{80}{80} + \frac{3}{8}\right)}{3}$$

$$= \left(\frac{106}{48}\right) / 3$$

$$= 0{,}368$$

$$Recall = \frac{\left(\frac{10}{10+0} + \frac{80}{80+5} + \frac{5}{5+0}\right)}{3}$$

$$= \frac{\left(\frac{10}{10} + \frac{80}{85} + \frac{5}{5}\right)}{3}$$

$$= \left(\frac{50}{17}\right) / 3$$

$$= 0{,}3268$$

$$F1\ Score = \frac{(2 * 1 * 0.83)}{(1+0.83)} + \frac{(2 * 0.94 * 1)}{(0.94+1)} + \frac{(2 * 1 * 0.625)}{(1+0.625)}$$

$$= \frac{0.905 + 0.9691 + 0.7692}{3}$$

$$= 0.8811$$

Pada perhitungan di atas, algoritma ini mendapatkan *accuracy* sebesar 95%, *precision* 36.8%, *recall* 32%, dan *f1-score* 88.11%. Dari perhitungan tersebut, terdapat perbedaan antara *accuracy* terhadap *precision* dan *recall*, hal tersebut dapat mempengaruhi kinerja algoritma, hal ini akan menjadi fatal jika algoritma memprediksi pasien yang tidak terkena stunting padahal terkena stunting (Martin & Nilawati, 2019).

4.3.4 Pengujian Klasifikasi *Random Forest*

Random Forest merupakan salah satu metode CART (Classification and Regression Tree) dalam data mining dan tidak memerlukan asumsi apapun. Metode ini menggunakan konsep pohon keputusan (decision tree). *Random forest* pada dasarnya sama saja seperti *Decision Tree*, yang membedakan keduanya adalah, *Random Forest* membuat 3 model pohon dan mengambil data secara acak.

1) Membuat Model *Decision Tree*

   Inisialisasi model *decision tree* dibuat secara acak dalam pengambilan data. Model terdiri dari 3 model sebagai berikut:

a) Model 1 Decision Tree

Tabel 4. 24 Model 1 *Decision Tree*

No	No. KK	Var 1	Var 2	Var 3	Var 4	Var 5	Var 6	Var 7	Var 8	Var 9	Var 10	Var 11	Var 12	Var 13	label
1	14100520260300 001001001	46	39	0	3	1	2	2	7	2	0	1	3	2	3
2	14100520260200 001001001	58	0	0	3	3	0	0	7	1	3	3	3	0	2
3	14100520260300 001002002	66	0	0	3	3	0	0	7	3	3	3	3	0	2
4	14100520260200 001002002	53	0	0	3	3	0	0	7	3	3	3	3	0	2
5	14100520260200 001003003	43	34	0	3	1	2	2	7	1	0	0	3	2	2
29	14100520260300 001010013	35	29	0	1	1	2	2	7	1	0	0	3	2	2
30	14100520260100 001004006	34	0	0	2	3	0	0	7	1	3	3	3	0	2
35	14100520210300 001006007	70	0	0	3	3	0	0	7	2	3	3	3	0	2
42	14100520030100 002002002	65	0	0	3	3	0	0	7	1	3	3	3	0	2
54	14100520210300 001011014	39	28	0	3	1	2	2	7	2	0	0	3	2	2
55	14100520210300 001012015	56	0	0	3	3	0	0	7	2	3	3	3	0	2
56	…	…	…	…	…	…	…	…	…	…	…	…	…	…	…
57	14100520030400 001002002	64	0	0	3	3	0	0	7	3	3	3	3	0	2
58	14100520180200 001002002	72	0	0	3	3	0	0	7	2	3	3	3	0	2
59	14100520180200 001003003	66	0	0	3	3	0	0	7	2	3	3	3	0	2

b) Model 2 *Decision Tree*

Tabel 4. 25 Model 2 *Decision Tree*

No	No. KK	Var 1	Var 2	Var 3	Var 4	Var 5	Var 6	Var 7	Var 8	Var 9	Var 10	Var 11	Var 12	Var 13	label
6	14100520260200 001004004	52	0	0	3	3	0	0	7	1	3	3	3	0	2
7	14100520260300 001003004	39	33	0	1	1	2	2	1	3	0	0	3	2	2
8	14100520260300 001004005	58	0	0	3	3	0	0	7	3	3	3	3	0	2
62	14100520210300 001016019	37	32	0	3	1	1	0	7	2	0	0	3	1	2
63	14100520030400 001003003	57	0	0	3	3	0	0	7	3	3	3	3	0	2
64	14100520260300 003056069	57	0	0	3	3	0	0	1	1	3	3	3	0	1
65	14100520030200 002008010	65	0	0	3	3	0	0	1	1	3	3	3	0	1
84	14100520030100 002007008	61	0	0	3	3	0	0	7	1	3	3	3	0	2
85	14100520030100 002014016	47	37	0	3	1	2	2	1	2	0	1	3	2	2
86	…	…	…	…	…	…	…	…	…	…	…	…	…	…	…
87	14100520180200 001007008	57	0	0	3	3	0	0	7	2	3	3	3	0	2
88	14100520030100 002016018	62	30	0	3	1	2	2	7	2	0	0	3	2	2
90	14100520210300 001017020	40	34	0	1	1	2	1	7	2	0	0	3	2	2

c) Model 3 *Decision Tree*

Tabel 4. 26 Model 3 *Decision Tree*

No	No. KK	Var 1	Var 2	Var 3	Var 4	Var 5	Var 6	Var 7	Var 8	Var 9	Var 10	Var 11	Var 12	Var 13	label
9	14100520260300 001006007	71	0	0	3	3	0	0	7	1	3	3	3	0	2
10	14100520260300 001007009	32	28	0	1	1	2	1	7	1	0	0	3	2	2
13	14100520260300 001009012	63	0	0	3	3	0	0	7	1	3	3	3	0	2
50	14100520210300 001010012	48	46	0	3	1	2	2	7	1	0	1	3	2	2
51	14100520210300 001011013	74	0	0	3	3	0	0	7	2	3	3	3	0	2
52	14100520030100 002003004	22	21	0	3	1	1	0	7	2	0	0	3	1	2
71	14100520260200 001014017	44	37	0	3	1	2	1	7	1	0	1	3	2	2
72	14100520180200 001004005	53	0	0	3	3	0	0	7	2	3	3	3	0	2
73	14100520030400 001008009	42	40	0	3	1	2	1	5	3	0	1	3	2	3
74	14100520260200 001015018	65	0	0	3	3	0	0	7	1	3	3	3	0	2
79	….	…	…	…	…	…	…	…	…	…	…	…	…	...	…
80	14100520030200 002014017	58	0	0	3	3	0	0	1	1	3	3	3	0	1
89	14100520260100 001006008	43	38	0	3	1	2	1	7	1	0	1	3	2	2

2) Menghitung Gain dan Entrophy Model

a) Gain dan Entrophy Model 1

Misalnya Menghitung Nilai Entrophy Total:

$$Entrophy\ Total$$
$$= (\left(-\frac{1}{30}\right) * IMLOG2\left(\frac{1}{30}\right) + \left(-\frac{26}{30}\right) * IMLOG2\left(\frac{26}{30}\right) + \left(-\frac{3}{30}\right) * IMLOG2\left(\frac{3}{30}\right))$$
$$Entrophy\ Total = 0{,}674679923$$

Misalnya Menghitung Nilai Entrophy Variabel Degree pada Atribut 2:

$$Entrophy(Var\ 1)$$
$$= (\left(-\frac{0}{3}\right) * IMLOG2\left(\frac{0}{3}\right) + \left(-\frac{0}{3}\right) * IMLOG2\left(\frac{0}{3}\right) + \left(-\frac{3}{3}\right) * IMLOG2\left(\frac{3}{3}\right))$$
$$Entrophy(Var\ 1) = 0$$

Misalnya Menghitung Nilai Gain Variabel Degree:

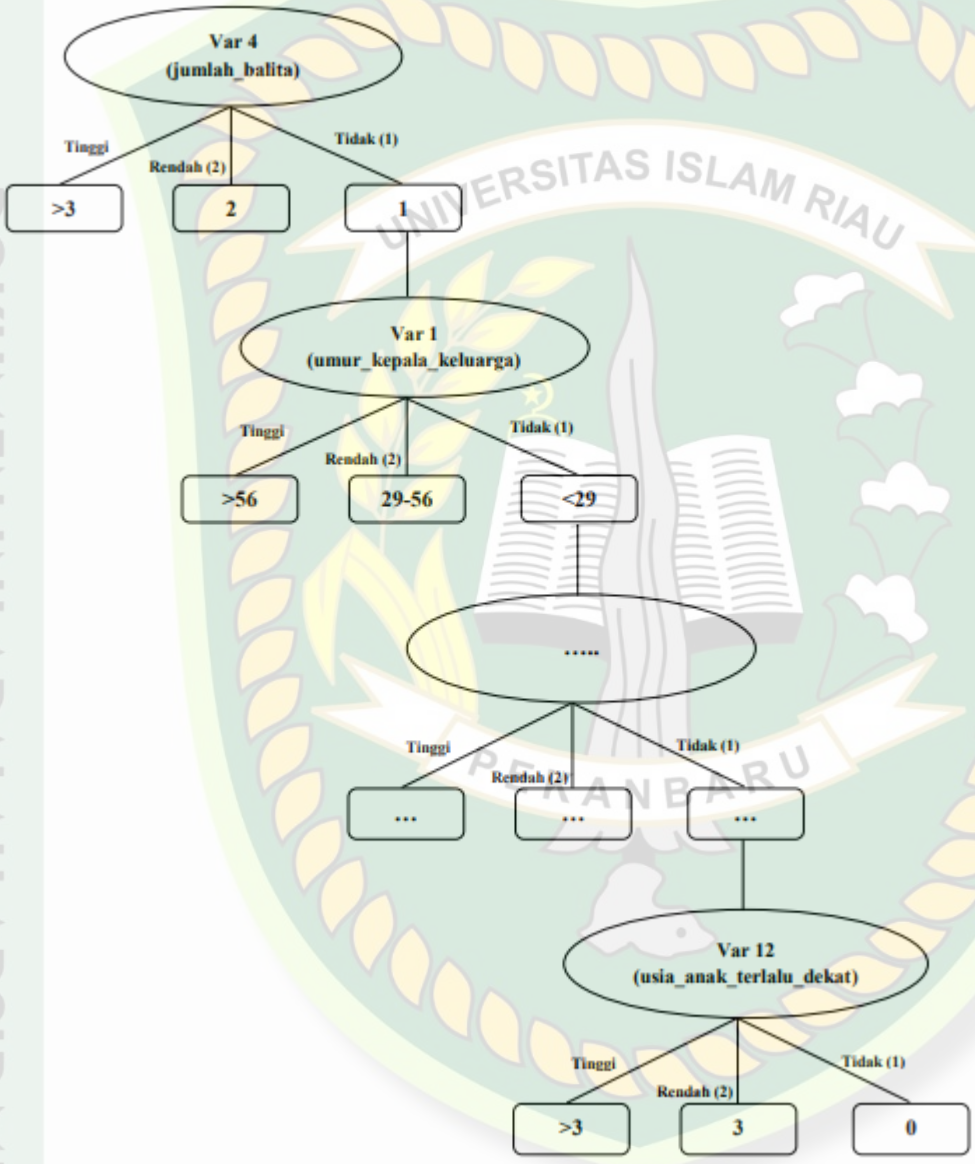
$$Gain = (0{,}674679923) - \left(\left(\frac{3}{30}\right) * 0\right) + \left(\left(\frac{2}{30}\right) * 0\right), \ldots + \left(\left(\frac{2}{30}\right) * 0\right)$$
$$Gain = 0{,}674679923$$

Hasil Perhitungan Entrophy dan Gain Model 1:

Tabel 4. 27 Hasil Perhitungan *Entrophy* dan *Gain* Model 1

		Jumlah (S)	Si Tinggi (3)	Si Rendah (2)	Si Tidak (1)	Entrophy	Gain
Total		30	1	26	3	0,674679923	
Var 1							0,674679923
	34	3	0	3	0	0	
	…	…	…	…	…	…	
	80	2	0	2	0	0	
Var 2							0,674679923
	0	20	0	17	3	0	
	…	…	…	…	…	…	
	47	1	0	1	0	0	
Var 3							0,01428158
	0	28	1	24	3	0,707569654	
	1	2	0	2	0	0	
Var 4							1,319648242
	1	3	0	3	0	0	
	3	26	1	22	3	0,744194214	
Var 5							0,674679923
	1	10	1	9	0	0	
	3	20	0	17	3	0	
Var 6							0,674679923
	0	20	0	17	3	0	
	2	10	1	9	0	0	
Var 7							0,674679923
	0	20	0	17	3	0	
	2	8	1	7	0	0	
Var 8							0,674679923
	1	1	0	0	1	0	
	7	26	1	25	0	0	
Var 9							0,674679923
	1	19	0	16	3	0	
	3	3	0	3	0	0	
Var 10							0,674679923
	0	10	1	9	0	0	
	3	20	0	17	3	0	
Var 11							0,674679923
	0	6	0	6	0	0	
	3	20	0	17	3	0	
Var 12							0
	3	30	1	26	3	0,674679923	
Var 13							0,674679923
	0	20	0	17	3	0	
	2	10	1	9	0	0	
Max Gain Model 1							**1,319648242**

Nilai max gain pada model 1 menjadi node atau akar yaitu berapa pada variabel Var 4 (jumlah_balita).



Gambar 4. 4 Pohon Keputusan *Random Forest* Model 1

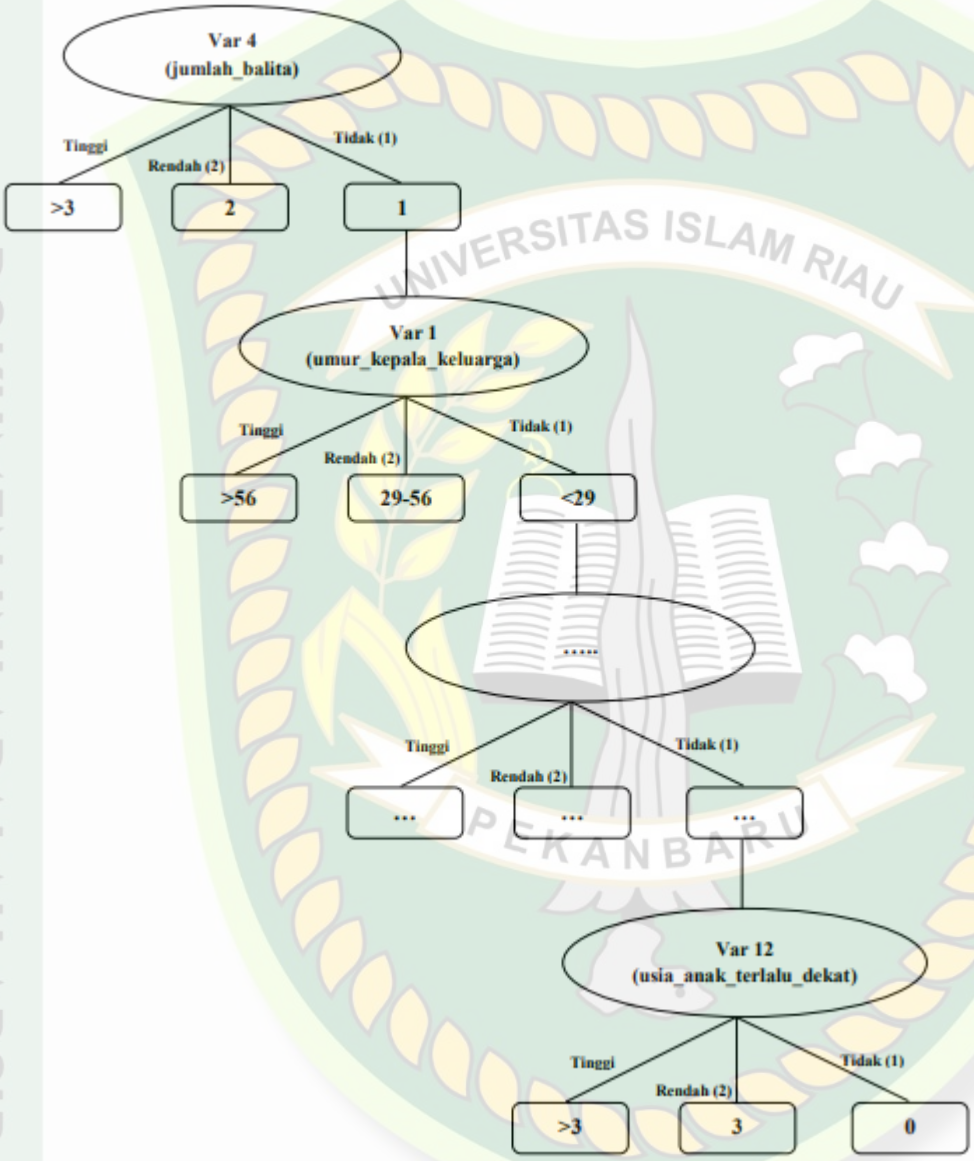
b) *Gain* dan *Entrophy* Model 2

Hasil Perhitungan *Entrophy* dan *Gain* Model 2:

Tabel 4. 28 Hasil Perhitungan *Entrophy* dan *Gain* Model 2

		Jumlah (S)	Si Tinggi (3)	Si Rendah (2)	Si Tidak (1)	Entrophy	Gain
Total		30	1	27	2	0,560825177	
Var 1							0,560825177
	21	1	0	1	0	0	
	…	…	…	…	…	…	
	94	1	0	1	0	0	
Var 2							0,560825177
	0	19	0	17	2	0	
	…	…	…	…	…	…	
Var 3							0
	0	30	1	27	2	0,560825177	
Var 4							1,093788815
	1	5	0	5	0	0	
	3	25	1	22	2	0,639556365	
Var 5							0,560825177
	1	11	1	10	0	0	
	3	19	0	17	2	0	
Var 6							0,560825177
	0	19	0	17	2	0	
Var 7							0,560825177
	0	20	0	18	2	0	
	2	4	0	4	0	0	
Var 8							0,560825177
	1	5	0	3	2	0	
	7	25	1	24	0	0	
Var 9							0,560825177
	1	13	0	11	2	0	
	3	7	0	7	0	0	
Var 10							0,560825177
	0	11	1	10	0	0	
	3	19	0	17	2	0	
Var 11							0,560825177
	0	6	0	6	0	0	
	3	19	0	17	2	0	
Var 12							0
	3	30	1	27	2	0,560825177	
Var 13							0,560825177
	0	19	0	17	2	0	
Max Gain Model 2							**1,093788815**

Nilai max gain pada model 2 menjadi node atau akar yaitu berapa pada variabel Var 4 (jumlah_balita).



Gambar 4. 5 Pohon Keputusan *Random Forest* Model 2

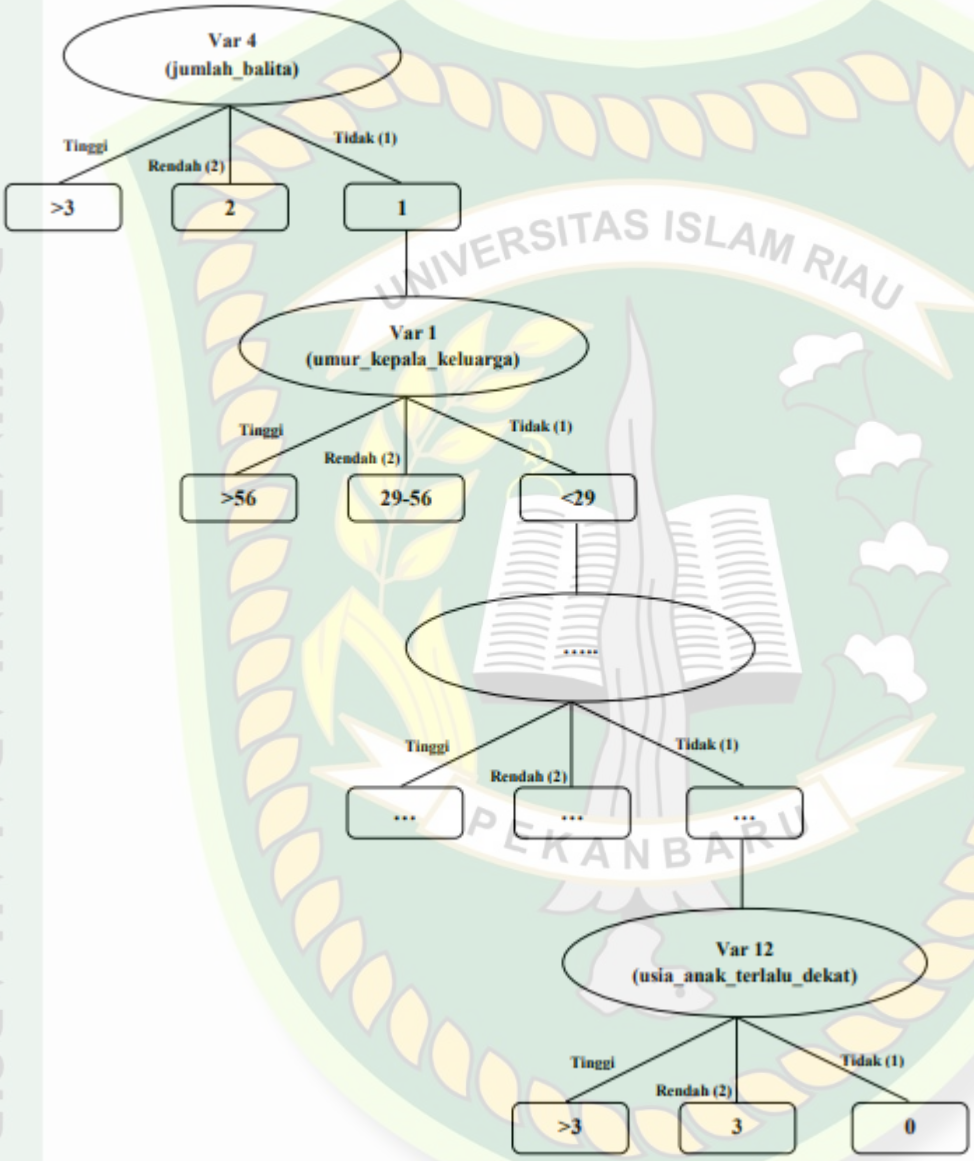
c) *Gain* dan *Entrophy* Model 3

Hasil Perhitungan *Entrophy* dan *Gain* Model 3:

Tabel 4. 29 Hasil Perhitungan *Entrophy* dan *Gain* Model 3

		Jumlah (S)	Si Tinggi (3)	Si Rendah (2)	Si Tidak (1)	Entrophy	Gain
Total		30	2	27	1	0,560825177	
Var 1							0,560825177
	22	1	0	1	0	0	
	…	…	…	…	…	…	
	74	1	0	1	0	0	
Var 2							0,560825177
	0	16	0	15	1	0	
	…	…	…	…	…	…	
	46	1	0	1	0	0	
Var 3							0,005157971
	0	29	2	26	1	0,574828144	
	1	1	0	1	0	0	
Var 4							0,966569599
	1	4	1	3	0	0	
	3	26	1	24	1	0,468166641	
Var 5							0,560825177
	1	14	2	12	0	0	
	3	16	0	15	1	0	
Var 6							0,560825177
	0	16	0	15	1	0	
Var 7							0,560825177
	0	17	0	16	1	0	
Var 8							0,560825177
	1	1	0	0	1	0	
	7	27	1	26	0	0	
Var 9							0,560825177
	1	17	0	16	1	0	
Var 10							0,560825177
	0	13	1	12	0	0	
	3	16	0	15	1	0	
Var 11							0,560825177
	0	6	1	5	0	0	
	3	16	0	15	1	0	
Var 12							0
	3	30	2	27	1	0,560825177	
Var 13							0,560825177
	0	16	0	15	1	0	
	2	13	2	11	0	0	
Max Gain Model 3							**0,966569599**

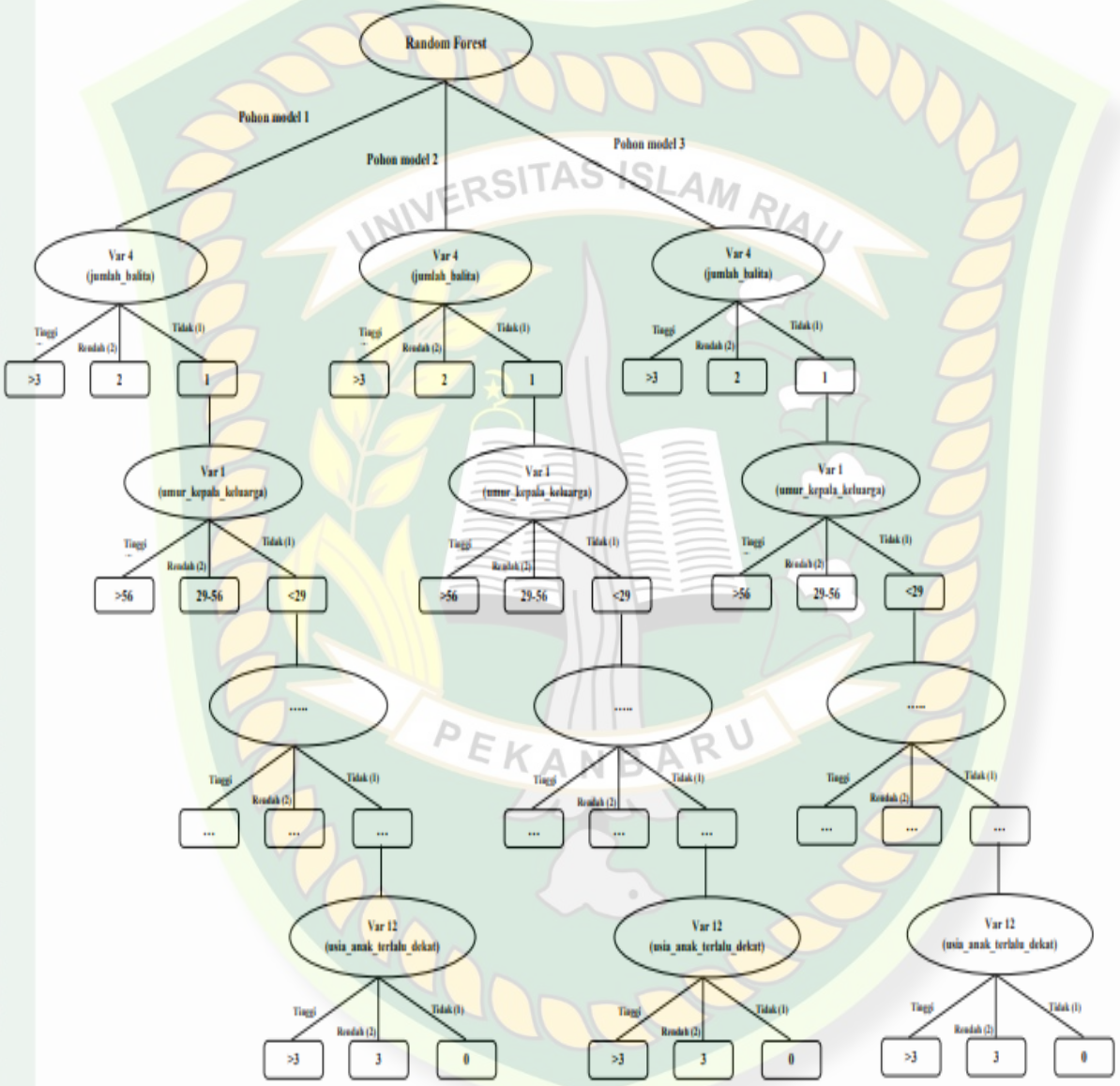
Nilai max gain pada model 3 menjadi node atau akar yaitu berapa pada variabel Var 4 (jumlah_balita).



Gambar 4. 6 Pohon Keputusan *Random Forest* Model 3

Dari 3 model Decision Tree yang telah dibuat, maka Random Forest akan menggabungkan 3 model tersebut. Sehingga jika divisualisasikan adalah sebagai berikut:

Dari 3 model Decision Tree yang telah dibuat, maka Random Forest akan menggabungkan 3 model tersebut. Sehingga jika divisualisasikan adalah sebagai berikut:



Gambar 4. 7 Pohon Keputusan *Random Forest*

3) Hasil Klasifikasi Random Forest

Pada hasil Random Forest diatas, didapat bahwa setiap model tree hanya memiliki akar dan daun (leaf). Sehingga Setiap data test akan ditentukan berdasarkan model yang dibuat dengan melihat akar dan leaf nya.

Pada model 1, 2, 3, fitur yang akan dilihat adalah Var 4 (jumlah_balita). Sehingga setiap satu data test akan selalu memiliki 3 klasifikasi dari 3 model. Untuk menentukan kelas yang akan diklasifikasi akan menggunakan sistem majorit vactory yang dimana jika 2 model mengklasifikasi yes dan 1 model mengklasifikasi no. Maka data tersebut akan diklasifikasi yes.

Tabel 4. 30 Hasil Klasifikasi *Random Forest*

No	Klasifikasi Model 1	Klasifikasi Model 2	Klasifikasi Model 3	Hasil Klasifikasi Random Forest	Aktual Label
91	3	3	3	3	2
92	3	3	3	3	2
93	1	1	1	1	1
94	1	1	1	1	3
95	1	1	1	1	1
96	3	3	3	3	2
97	1	1	1	1	2
98	1	1	1	1	2
99	1	1	1	1	1
100	1	1	1	1	1

Pada hasil pengujian di atas dapat dilihat bahwa algoritma *random forest* memiliki hasil yang bervariatif seperti algoritma *naive bayes*. Hal tersebut dapat terjadi karena algoritma ini memiliki sifat yang mirip dengan *decision tree*, yaitu membuat sebuah *rule* pohon yang bercabang secara acak (Godishala et al., 2022).

4) Output Confusion Matrix

Berdasarkan pengujian yang dilakukan, dapat dihitung nilai accuracy menggunakan persamaan 2.8, nilai precision menggunakan persamaan 2.9, nilai recall menggunakan persamaan 2.10, dan nilai F1-Score menggunakan persamaan 2.11.

Tabel 4. 31 *Confusion Matrix* Algoritma *Random Forest*

Predicted / **Aktual**	**Tinggi**	**Rendah**	**Tidak**
Tinggi	10	2	1
Rendah	0	80	0
Tidak	0	3	4

$$Accuracy = \frac{10 + 80 + 4}{10 + 0 + 80 + 5 + 5 + 0}$$

$$= \frac{94}{100}$$

$$= 0{,}94$$

$$Precision = \frac{\left(\frac{10}{10+3} + \frac{80}{80+0} + \frac{3}{3+4}\right)}{3}$$

$$= \frac{\left(\frac{10}{13} + \frac{80}{80} + \frac{3}{7}\right)}{3}$$

$$= \left(\frac{41}{13}\right) / 3$$

$$= 0{,}3504$$

$$Recall = \frac{\left(\frac{10}{10+0} + \frac{80}{80+5} + \frac{4}{4+1}\right)}{3}$$

$$= \frac{\left(\frac{10}{10} + \frac{80}{85} + \frac{4}{5}\right)}{3}$$

$$= \left(\frac{233}{85}\right) / 3$$

$$= 0{,}3042$$

$$F1\ Score = \frac{(2 * 1 * 0.77)}{(1+0.77)} + \frac{(2 * 1 * 0.94)}{(1+0.94)} + \frac{(2 * 0.8 * 0.57)}{(0.8+0.57)}$$

$$= \frac{0.8700 + 0.9691 + 0.82852}{3}$$

$$= 0.8892$$

Pada perhitungan evaluasi algoritma *random forest* di atas didapatkan nilai *accuracy* sebesar 94%, *precision* 35.4%, *recall* 30.42%, dan *f1-score* 88.92%. Seperti algoritma *naive bayes* sebelumnya, antara *precision* dan *recall* memiliki hasil yang kurang baik dibandingkan dengan akurasinya. Hal ini dapat berakibat fatal apabila algoritma tersebut digunakan, bisa terjadi keluarga yang beresiko stunting akan tetapi terprediksi tidak beresiko stunting.

Tabel 4. 32 Perbandingan Algoritma

Algoritma Machine Learning	Accuracy	Precission	Recall	F1-Score
K-NN	92.42 %	85.13 %	72.13 %	76.15 %
Decision Tree	99.76 %	99.73 %	99.58 %	99.65 %
Naive Bayes	41.69 %	43.22 %	74.53 %	33.95 %
Random Forest	99.25%	99.27 %	98.6 %	99.93%

Pada tabel di atas dapat dilihat bahwa *accuracy*, *precision*, *recall*, serta *f1-score* terbaik adalah algoritma *decision tree*. Hasil tersebut diperkuat dengan penelitian serupa dimana algoritma tersebut lebih unggul dibandingkan dengan algoritma yang lain (Putri et al., 2024).

BAB V
KESIMPULAN DAN SARAN

5.1 Kesimpulan

Dari penelitian yang telah dilakukan dalam klasifikasi keluarga beresiko *stunting* pada balita di Kecamatan Merbau dengan melakukan perbandingan algoritma *machine learning* didapatkan kesimpulan sebagai berikut:

1. Klasifikasi keluarga beresiko *stunting* yang dapat meminimalisir peningkatan angka *stunting* di Provinsi Riau yaitu dengan membuat sistem yang dapat mengklasifikasi keluarga beresiko *stunting* menggunakan bahasa pemrograman Python. Data yang digunakan terdiri dari 3 jenis data yaitu data PK, KB dan KPD. Dari data tersebut dilakukan *preprocessing* sebagai proses menyiapkan data dengan melakukan *label encoder* yaitu mengubah setiap nilai dalam kolom menjadi angka berurutan. Jumlah dataset yang digunakan yaitu berjumlah 3349 data yang memiliki label rendah, tinggi dan tidak. Jumlah data pada label Rendah yaitu 660. Jumlah data pada label Tinggi yaitu 448 dan jumlah data pada label Tidak yaitu 2241. Pembagian dataset terdiri dari data *training* dan data *testing* dengan rasio perbandingan 90:10. Kemudian dilakukan klasifikasi, proses klasifikasi menerapkan metode KDD sebagai kerangka kerja data mining.
2. Implementasi algoritma *K-Nearest Neighbor, Random Forest, Decision Tree* dan *Naïve Bayes* dalam klasifikasi keluarga beresiko stunting di Kecamatan Merbau dilakukan dengan menerapkan kerangka kerja *Knowledge Discovery in Databases* (KDD). Pada kerangka kerja KDD terdapat beberapa tahapan yaitu data *selection, preprocessing/cleaning, transformation,* data mining dan *interpretation/evaluation.* Proses klasifikasi dilakukan menggunakan bahasa pemrograman Python. Setelah pembentukan model dilakukan evaluasi menggunakan *k-fold cross validation* untuk menguji keakuratan metode dalam melakukan klasifikasi.

3. Hasil performa dari algoritma *K-Nearest Neighbor, Random Forest, Decision Tree* dan *Naïve Bayes* dalam melakukan pengklasifikasian keluarga beresiko stunting di Kecamatan Merbau yaitu pada akurasi *k-fold cross validation* didapatkan akurasi tertinggi pada algoritma *Decision Tree* yang mendapatkan nilai 99.76% dengan *precision* 99.73%, *recall* 99.58%, *dan* f-1 *score* 99.65%. Pada algoritma *Random Forest* mendapatkan nilai *accuracy* 99.25%. Kemudian, pada algoritma K-NN mendapatkan 92.42% dan akurasi terendah yaitu algoritma *naïve bayes* dengan nilai 41.69%.
4. Algoritma *Decision tree* mendapatkan akurasi tertinggi tidak lepas pengaruh dari dataset yang digunakan. Dataset tersebut tergolong sedikit, sehingga algoritma ini dapat mempelajari pola secara mudah yang terdapat pada data tersebut. Selain itu, algoritma ini juga mampu mengeliminasi perhitungan atau data-data yang kiranya tidak diperlukan yang berdampak kepada proses komputasi yang lebih cepat (Alfraidi et al., 2023).
5. Algoritma *Naive bayes* mendapatkan akurasi terendah karena di dalam pembelajaran mesin tidak bisa diukur menggunakan satu probabilitas saja, sehingga menyebabkan ketidakakuratan algoritma dalam memproses data uji (Suprianto, 2020).

5.2 Saran

Adapun saran yang dapat diberikan pada penelitian selanjutnya adalah melakukan pembuatan aplikasi dalam bentuk *website* maupun *mobile* secara *online* dengan tujuan untuk memudahkan dalam melalukan klasifikasi keluarga beresiko *stunting* dan melihat hasil klasifikasi, serta dapat diberikan pula informasi mengenai alternatif saran pada keluarga yang memiliki resiko *stunting.*

DAFTAR PUSTAKA

Adriansyah, I., Mahendra, M. D., Rasywir, E., & Pratama, Y. (2022). Perbandingan Metode Random Forest Classifier dan SVM Pada Klasifikasi Kemampuan Level Beradaptasi Pembelajaran Jarak Jauh Siswa. *Bulletin of Informatics and Data Science*, *1*(2), 98. https://doi.org/10.61944/bids.v1i2.49

Aheto, M. K., & Dagne, G. A. (2021). Geostatistical analysis, web-based mapping, and environmental determinants of under-5 stunting: evidence from the 2014 Ghana Demographic and Health Survey. In *Articles Lancet Planet Health* (Vol. 5).

Akseer, N., Tasic, H., Onah, M. N., Wigle, J., Rajakumar, R., Sanchez-Hernandez, D., Akuoku, J., Black, R. E., Horta, B. L., Nwuneli, N., Shine, R., Wazny, K., Japra, N., Shekar, M., & Hoddinott, J. (2022). *Economic costs of childhood stunting to the private sector in low-and middle-income countries*. https://doi.org/10.1016/j

Alfraidi, W., Shalaby, M., & Alaql, F. (2023). Modeling EV Charging Station Loads Considering On-Road Wireless Charging Capabilities. *World Electric Vehicle Journal*, *14*(11), 313. https://doi.org/10.3390/wevj14110313

Angreni, I. A., Adisasmita, S. A., Ramli, M. I., & Hamid, S. (2019). Pengaruh Nilai K pada Metode K-Nearest Neighbor (KNN) terhadap Tingkat Akurasi Indentifikasi Kerusakan Jalan. *Rekayasa Sipil*, *7*(2), 63. https://doi.org/10.22441/jrs.2018.v07.i2.01

Annisa, R. (2019). ANALISIS KOMPARASI ALGORITMA KLASIFIKASI DATA MINING UNTUK PREDIKSI PENDERITA PENYAKIT JANTUNG. *Jurnal Teknik Informatika Kaputama (JTIK)*, *3*(1).

Arisandi, R. R. R., Warsito, B., & Hakim, A. R. (2022). Aplikasi Naïve Bayes Classifier (Nbc) Pada Klasifikasi Status Gizi Balita Stunting Dengan Pengujian K-Fold Cross Validation. *Jurnal Gaussian*, *11*(1), 130–139. https://doi.org/10.14710/j.gauss.v11i1.33991

Asha Kiranmai, S., & Jaya Laxmi, A. (2018). Data mining for classification of power quality problems using WEKA and the effect of attributes on classification accuracy. *Protection and Control of Modern Power Systems*, *3*(1). https://doi.org/10.1186/s41601-018-0103-3

Atmarita, & Zahrani, Y. (2018). *Situasi Balita Pendek (Stunting) di Indonesia*. Kementerian Kesehatan.

Barile, C., Casavola, C., Pappalettera, G., & Kannan, V. P. (2022). Damage Progress Classification in AlSi10Mg SLM Specimens by Convolutional Neural Network

and k-Fold Cross Validation. *Materials*, *15*(13). https://doi.org/10.3390/ma15134428

BKKBN. (2020). Pedoman Pelaksanaan Pendataan Keluarga Tahun 2021. In *Jakarta*.

BKKBN. (2021). Perban 12 Ran Pasti. *Peraturan Badan Kependudukan Dan Keluarga Berencana Nasional Republik Indonesia Nomor 12 Tahun 2021 Tentang Rencana Aksi Nasional Percepatan Penurunan Angka Stunting Indonesia Tahun 2021-2024*, 1–162.

Boucot, A., & Poinar Jr., G. (2010). Stunting. *Fossil Behavior Compendium*, 243–243. https://doi.org/10.1201/9781439810590-c34

Chazar, C., & Widhiaputra, B. E. (2020). INFORMASI (Jurnal Informatika dan Sistem Informasi) Machine Learning Diagnosis Kanker Payudara Menggunakan Algoritma Support Vector Machine. *Jurnal Informatika Dan Sistem Informasi*.

Cholil, S. R., Handayani, T., Prathivi, R., & Ardianita, T. (2021). Implementasi Algoritma Klasifikasi K-Nearest Neighbor (KNN) Untuk Klasifikasi Seleksi Penerima Beasiswa. *IJCIT (Indonesian Journal on Computer and Information Technology)*, *6*(2), 118–127.

Desiani, A. (2022). Perbandingan Implementasi Algoritma Naïve Bayes dan K-Nearest Neighbor Pada Klasifikasi Penyakit Hati. *SIMKOM*, *7*(2), 104–110. https://doi.org/10.51717/simkom.v7i2.96

Devita, R. N., Herwanto, H. W., & Wibawa, A. P. (2018). Perbandingan Kinerja Metode Naive Bayes dan K-Nearest Neighbor untuk Klasifikasi Artikel Berbahasa indonesia. *Jurnal Teknologi Informasi Dan Ilmu Komputer*, *5*(4), 427. https://doi.org/10.25126/jtiik.201854773

Ditjen Bangda. (2021). *Monitoring Pelaksanaan 8 Aksi Konvergensi Intervensi Penurunan Stunting Terintegrasi*. Ditjen Bina Pembangunan Daerah-Kementerian Dalam Negeri.

Donnellan, E., Aslan, S., Fastrich, G. M., & Murayama, K. (2022). How Are Curiosity and Interest Different? Naïve Bayes Classification of People's Beliefs. In *Educational Psychology Review* (Vol. 34, Issue 1, pp. 73–105). Springer. https://doi.org/10.1007/s10648-021-09622-9

Duan, J., & Gao, R. (2021). Research on college English teaching based on data mining technology. *Eurasip Journal on Wireless Communications and Networking*, *2021*(1). https://doi.org/10.1186/s13638-021-02071-6

Dwita, I., & Tarigan, S. (2023). *Evaluasi Algoritma Klasifikasi Machine Learning Kategori Nilai Akhir Tunjangan Kinerja Pegawai*. *6*(3), 251–261.

Enri, U. (2018). PENERAPAN ALGORITMA C4.5 DALAM PEMILIHAN PROGRAM STUDI FAKULTAS ILMU KOMPUTER (Studi Kasus Sekolah Menengah Atas Negeri 1 Tambun Utara). *Jurnal Rekayasa Informasi*, *7*(1), 1–7.

Godishala, A. K., Yassin, H., Veena, R., & Lai, D. T. C. (2022). Breast Cancer Tumor Image Classification Using Deep Learning Image Data Generator. *2022 7th International Conference on Image, Vision and Computing, ICIVC 2022*, 418–423. https://doi.org/10.1109/ICIVC55077.2022.9886202

Hafid, H. (2023). Penerapan K-Fold Cross Validation untuk Menganalisis Kinerja Algoritma K-Nearest Neighbor pada Data Kasus Covid-19 di Indonesia. *Journal of Mathematics*, *6*(2), 161–168.

Hatem, M. Q. (2022). Skin lesion classification system using a K-nearest neighbor algorithm. *Visual Computing for Industry, Biomedicine, and Art*, *5*(1). https://doi.org/10.1186/s42492-022-00103-6

Haviluddin. (2011). Memahami Penggunaan UML (Unified Modelling Language). *Memahami Penggunaan UML (Unified Modelling Language)*, *6*(1), 1–15.

Hu, G., Yang, Z., Zhu, M., Huang, L., & Xiong, N. (2018). Automatic classification of insulator by combining k-nearest neighbor algorithm with multi-type feature for the Internet of Things. *Eurasip Journal on Wireless Communications and Networking*, *2018*(1), 1–10. https://doi.org/10.1186/s13638-018-1195-1

Id, I. D. (2021). MACHINE LEARNING: Teori, Studi Kasus dan Implementasi Menggunakan Python. In *Ur Press*. https://doi.org/10.5281/zenodo.5113507

Kamel, H., Abdulah, D., & Al-Tuwaijari, J. M. (2019). Cancer Classification Using Gaussian Naive Bayes Algorithm. *Proceedings of the 5th International Engineering Conference, IEC 2019*, 165–170. https://doi.org/10.1109/IEC47844.2019.8950650

Kantor camat Merbau. (2023). *Kecamatan Pulau Merbau*. Kantor Camat Merbau.

Latifah, R., Wulandari, E. S., & Kreshna, P. E. (2019). Model Decision Tree Untuk Prediksi Jadwal Kerja Menggunakan Scikit-Learn. *Jurnal Universitas Muhammadiyah Jakarta*, 1–6.

Li, L., Elhajj, M., Feng, Y., & Ochieng, W. Y. (2023). Machine learning based GNSS signal classification and weighting scheme design in the built environment: a comparative experiment. *Satellite Navigation*, *4*(1). https://doi.org/10.1186/s43020-023-00101-w

Li, M., Xu, H., & Deng, Y. (2019). Evidential decision tree based on belief entropy. *Entropy*, *21*(9). https://doi.org/10.3390/e21090897

Lyu, Z., Yu, Y., Samali, B., Rashidi, M., Mohammadi, M., Nguyen, T. N., & Nguyen, A. (2022). Back-Propagation Neural Network Optimized by K-Fold Cross-Validation for Prediction of Torsional Strength of Reinforced Concrete Beam. *Materials*, *15*(4). https://doi.org/10.3390/ma15041477

Mardi Yuli. (2016). Jurnal Edik Informatika Data Mining : Klasifikasi Menggunakan Algoritma C4.5 Yuli Mardi. *Jurnal Edik Informatika*.

Marito Putry, N., & Nurina Sari, B. (2022). KOMPARASI ALGORITMA KNN DAN NAÏVE BAYES UNTUK KLASIFIKASI DIAGNOSIS PENYAKIT DIABETES MELITUS. *Jurnal Sains Dan Manajemen*, *10*(1).

Martin, M., & Nilawati, L. (2019). Recall dan Precision Pada Sistem Temu Kembali Informasi Online Public Access Catalogue (OPAC) di Perpustakaan. *Paradigma - Jurnal Komputer Dan Informatika*, *21*(1), 77–84. https://doi.org/10.31294/p.v21i1.5064

Mediacenter Riau. (2023). *Angka Stunting Riau di 9 Kabupaten dan Kota Turun*. PPID Riau.

Muslim, M. A., Prasetiyo, B., Mawarni, Eva Laily Harum Herowati, A. J., Mirqotus'saadah, & Rukmana, Siti Hardiyanti Nurzahputra, A. (2019). *Data Mining Algoritma C4.5*.

Nalatissifa, H., Gata, W., Diantika, S., & Nisa, K. (2021). Perbandingan Kinerja Algoritma Klasifikasi Naive Bayes, Support Vector Machine (SVM), dan Random Forest untuk Prediksi Ketidakhadiran di Tempat Kerja. *Jurnal Informatika Universitas Pamulang*, *5*(4), 578. https://doi.org/10.32493/informatika.v5i4.7575

Nasrullah, A. H. (2021). Implementasi Algoritma Decision Tree Untuk Klasifikasi Data Peserta Didik. *Jurnal Pilar Nusa Mandiri*, *7*(2), 217.

Olsa, E. D., Sulastri, D., & Anas, E. (2018). Hubungan Sikap dan Pengetahuan Ibu Terhadap Kejadian Stunting pada Anak Baru Masuk Sekolah Dasar di Kecamanatan Nanggalo. *Jurnal Kesehatan Andalas*, *6*(3), 523. https://doi.org/10.25077/jka.v6i3.733

Peryanto, A., Yudhana, A., & Umar, R. (2020). Klasifikasi Citra Menggunakan Convolutional Neural Network dan K Fold Cross Validation. *Journal of Applied Informatics and Computing (JAIC)*, *4*(1), 45.

Ponum, M., Khan, S., Hasan, O., Mahmood, M. T., Abbas, A., Iftikhar, M., & Arshad, R. (2020). Stunting diagnostic and awareness: Impact assessment study of sociodemographic factors of stunting among school-going children of Pakistan. *BMC Pediatrics*, *20*(1). https://doi.org/10.1186/s12887-020-02139-0

Prasetiya, T., Ali, I., Rohmat, C. L., & Nurdiawan, O. (2020). Klasifikasi Status Stunting Balita Di Desa Slangit Menggunakan Metode K-Nearest Neighbor. *INFORMATICS FOR EDUCATORS AND PROFESSIONALS*, *4*(2), 93–104.

Prihandoyo M Teguh. (2018). *Unified Modeling Language (UML) Model Untuk Pengembangan Sistem Informasi Akademik Berbasis Web*. *03*(01), 126–129.

Purnamawati, A., Nugroho, W., Putri, D., & Hidayat, W. F. (2020). Deteksi Penyakit Daun pada Tanaman Padi Menggunakan Algoritma Decision Tree, Random Forest, Naïve Bayes, SVMdan KNN. *InfoTekJar: Jurnal Nasional Informatika*

Dan Teknologi Jaringan, *5*(1), 212–215. https://doi.org/10.30743/infotekjar.v5i1.2934

Putri, A. I., Syarif, Y., Jayadi, P., Arrazak, F., & Salisah, F. N. (2024). Implementasi Algoritma Decision Tree dan Support Vector Machine (SVM) untuk Prediksi Risiko Stunting pada Keluarga. *MALCOM: Indonesian Journal of Machine Learning and Computer Science*, *3*(2), 349–357. https://doi.org/10.57152/malcom.v3i2.1228

Rahayu, A., Yulidasari, F., Putri, A. O., & Anggraini, L. (2018). Study Guide - Stunting dan Upaya Pencegahannya. In *Buku stunting dan upaya pencegahannya*.

Ridwan, M., Suyono, H., & Sarosa, M. (2013). Penerapan Data Mining Untuk Evaluasi Kinerja Akademik Mahasiswa Menggunakan Algoritma Naive Bayes Classifier. *Eeccis*, *7*(1), 59–64. https://doi.org/10.1038/hdy.2009.180

Rita Kirana, Aprianti, N. W. H. (2022). Pengaruh Media Promosi Kesehatan Terhadap Perilaku Ibu Dalam Pencegahan Stunting Di Masa Pandemi Covid-19 (Pada Anak Sekolah Tk Kuncup Harapan Banjarbaru). *Jurnal Inovasi Penelitian*, *2*(9), 2899–2906.

Ruan, S., Li, H., Li, C., & Song, K. (2020). Class-specific deep feature weighting for naïve bayes text classifiers. *IEEE Access*, *8*, 20151–20159. https://doi.org/10.1109/ACCESS.2020.2968984

Samudra, A. Y. (2019). Pendekatan Random Forest untuk Model Peramalan Harga Tembakau Rajangan Di Kabupaten Temanggung. In *Universitas Sanata Dharma* (Vol. 8, Issue 5). Universitas Sanata Dharma.

Satriawan, E. (2018). *Strategi Nasional Percepatan Pencegahan Stunting 2018-2024 (National Strategy for Accelerating Stunting Prevention 2018-2024)*. Tim Nasional Percepatan Penanggulangan Kemiskinan.

Schumacher, D., & Jr, L. F. (2023). *The Hitchhickers's Guide to Machine Learning Algorithms*.

Soufitri, F. (2019). Perancangan Data Flow Diagram Untuk Sistem Informasi Sekolah (Studi Kasus Pada SMP Plus Terpadu). *Regional Development Industry & Health Science, Technology and Art of Life*, *2*(1), 240–246.

Sulistyawati, F., & Widarini, N. P. (2022). Kejadian Stunting Masa Pandemi Covid-19 Stunting Incidents During the COVID-19 Pandemic. *Medika Respati : Jurnal Ilmiah Kesehatan*, *17*(Februari), 37–46.

Supangat, S., Amna, A. R., & Rahmawati, T. (2018). Implementasi Decision Tree C4.5 Untuk Menentukan Status Berat Badan dan Kebutuhan Energi Pada Anak Usia 7-12 Tahun. *Teknika*, *7*(2), 73–78. https://doi.org/10.34148/teknika.v7i2.90

Suprianto, S. (2020). Implementasi Algoritma Naive Bayes Untuk Menentukan

Lokasi Strategis Dalam Membuka Usaha Menengah Ke Bawah di Kota Medan (Studi Kasus: Disperindag Kota Medan). *Jurnal Sistem Komputer Dan Informatika (JSON)*, *1*(2), 125. https://doi.org/10.30865/json.v1i2.1939

Syahril Dwi Prasetyo, Shofa Shofiah Hilabi, & Fitri Nurapriani. (2023). Analisis Sentimen Relokasi Ibukota Nusantara Menggunakan Algoritma Naïve Bayes dan KNN. *Jurnal KomtekInfo*, *10*, 1–7. https://doi.org/10.35134/komtekinfo.v10i1.330

Syarli, S., & Muin, A. (2016). Metode Naive Bayes Untuk Prediksi Kelulusan (Studi Kasus: Data Mahasiswa Baru Perguruan Tinggi). *Jurnal Ilmiah Ilmu Komputer*, *2*(1), 22–26.

Upadhyay, D., Manero, J., Zaman, M., & Sampalli, S. (2021). Intrusion Detection in SCADA Based Power Grids: Recursive Feature Elimination Model with Majority Vote Ensemble Algorithm. *IEEE Transactions on Network Science and Engineering*, *8*(3), 2559–2574. https://doi.org/10.1109/TNSE.2021.3099371

Uwiringiyimana, V., Osei, F., Amer, S., & Veldkamp, A. (2022). Bayesian geostatistical modelling of stunting in Rwanda: risk factors and spatially explicit residual stunting burden. *BMC Public Health*, *22*(1). https://doi.org/10.1186/s12889-022-12552-y

Widiastuti, N. I., Rainarli, E., & Dewi, K. E. (2017). Peringkasan dan Support Vector Machine pada Klasifikasi Dokumen. *Jurnal Infotel*, *9*(4), 416. https://doi.org/10.20895/infotel.v9i4.312

Yuliani, E., Yunding, J., Haerianti, M., Marendeng Majene, Stik., & Kartini Majene, J. R. (2018). *PELATIHAN KADER KESEHATAN DETEKSI DINI STUNTING PADA BALITA DI DESA BETTENG (Health Cadre Training About Early Detection Of Stunting Toddler In Betteng Village)*.

Zhang, C., Wang, X., Chen, S., Li, H., Wu, X., & Zhang, X. (2021). A modified random forest based on Kappa measure and binary artificial bee colony algorithm. *IEEE Access*. https://doi.org/10.1109/ACCESS.2021.3105796

Zhao, W., Shang, L., & Sun, J. (2019). Power quality disturbance classification based on time-frequency domain multi-feature and decision tree. *Protection and Control of Modern Power Systems*, *4*(1). https://doi.org/10.1186/s41601-019-0139-z

SURAT KEPUTUSAN DEKAN FAKULTAS TEKNIK UNIVERSITAS ISLAM RIAU
NOMOR : 0395/KPTS/FT-UIR/2024
TENTANG PENGANGKATAN TIM PEMBIMBING PENELITIAN DAN PENYUSUNAN SKRIPSI

DEKAN FAKULTAS TEKNIK

Membaca : Surat Ketua Program Studi Teknik Informatika Nomor : 161/TA-TI/FT/2023 tentang persetujuan dan usulan pengangkatan Tim Pembimbing penelitian dan penyusunan Skripsi.

Menimbang : 1. Bahwa untuk menyelesaikan perkuliahan bagi mahasiswa Fakultas Teknik perlu membuat Skripsi.

2. Untuk itu perlu ditunjuk Tim Pembimbing penelitian dan penyusunan Skripsi yang diangkat dengan Surat Keputusan Dekan.

Mengingat : 1. Undang - Undang Nomor 12 Tahun 2012 Tentang Pendidikan Tinggi
2. Peraturan Presiden Republik Indonesia Nomor 8 Tahun 2012 Tentang Kerangka Kualifikasi Nasional Indonesia
3. Peraturan Pemerintah Republik Indonesia Nomor 37 Tahun 2009 Tentang Dosen
4. Peraturan Pemerintah Republik Indonesia Nomor 66 Tahun 2010 Tentang Pengelolaan dan Penyelenggaraan Pendidikan
5. Peraturan Menteri Pendidikan Nasional Nomor 63 Tahun 2009 Tentang Sistem Penjaminan Mutu Pendidikan
6. Peraturan Menteri Pendidikan dan Kebudayaan Republik Indonesia Nomor 49 Tahun 2014 Tentang Standar Nasional Pendidikan Tinggi
7. Statuta Universitas Islam Riau Tahun 2018
8. Peraturan Universitas Islam Riau Nomor 001 Tahun 2018 Tentang Ketentuan Akademik Bidang Pendidikan Universitas Islam Riau

MEMUTUSKAN

Menetapkan : 1. Mengangkat saudara-saudara yang namanya tersebut dibawah ini sebagai Tim Pembimbing Penelitian & penyusunan Skripsi Mahasiswa Fak. Teknik Program Studi Teknik Informatika.

No	Nama	Pangkat	Jabatan
1.	Dr. Arbi Haza Nasution, B.IT., M.IT.	Lektor Kepala	Pembimbing

2. Mahasiswa yang akan dibimbing :

Nama : Syarifah Kusuma Maharani
NPM : 193510512
Program Studi : Teknik Informatika
Jenjang Pendidikan : Strata Satu (S1)
Judul Skripsi : Perbandingan Metode Machine Learning dalam Klasifikasi Potensi Keluarga Beresiko Stunting pada Kecamatan Merbau

3. Keputusan ini mulai berlaku pada tanggal ditetapkannya dengan ketentuan bila terdapat kekeliruan dikemudian hari segera ditinjau kembali.

Ditetapkan di : Pekanbaru
Pada Tanggal : 18 Ramadhan 1445 H
28 Maret 2024 M

Dekan,

Prof. Dr. Eng. Ir. Muslim.,ST.,MT.,IPU
NPK : 1016047901

Tembusan disampaikan :
1. Yth. Bapak Rektor UIR di Pekanbaru.
2. Yth. Sdr. Ketua Program Studi Teknik Informatika FT-UIR
3. Arsip

**Surat ini ditandatangani secara elektronik*

YAYASAN LEMBAGA PENDIDIKAN ISLAM (YLPI) RIAU
UNIVERSITAS ISLAM RIAU
Jalan Kaharuddin Nasution No. 113 P. Marpoyan Pekanbaru Riau Indonesia – Kode Pos: 28284
Telp. +62 761 674674 Fax. +62 761 674834 *Website*: www.uir.ac.id *Email*: info@uir.ac.id

F.A.3.10

KARTU BIMBINGAN TUGAS AKHIR
SEMESTER GANJIL TA 2023/2024

NPM : 193510512
Nama Mahasiswa : SYARIFAH KUSUMA MAHARANI
Dosen Pembimbing : 1. Dr ARBI HAZA NASUTION B.IT.(Hons), M.IT
Program Studi : TEKNIK INFORMATIKA
Judul Tugas Akhir : Perbandingan Metode *Machine Learning* dalam Klasifikasi Potensi Keluarga Bersiko Stunting pada Kecamatan Merbau
Judul Tugas Akhir (Bahasa Inggris) : Comparison of Machine Learning Methods in Classification of Stunting Risk Family Potential in Merbau District
Lembar Ke : .....

NO	Hari/Tanggal Bimbingan	Materi Bimbingan	Hasil / Saran Bimbingan	Paraf Dosen Pembimbing
1.	Selasa / 9 Agustus 2023	Algoritma yang digunakan	Penambahan Algoritma	
2.	Senin / 11 September 2023	Metode yang dipakai	Perubahan metode yang lebih efektif	
3.	Rabu / 27 September 2023	Penulisan	Revisi typo dan perbaikan tulisan	
4.	Kamis / 5 Oktober 2023	Acc proposal		
5.	Kamis / 14 Maret 2024	Bimbingan bab 4-5	Penambahan perhitungan algoritma	
6.	Kamis / 21 Maret 2024	ACC sidang kompre		

Pekanbaru, 3 April 2024
Wakil Dekan I/ Ketua Departemen/Ketua Prodi

MTKZNTEWNTEY

( )

Catatan :
1. Lama bimbingan Tugas Akhir/ Skripsi maksimal 2 semester sejak TMT SK Pembimbing diterbitkan
2. Kartu ini harus dibawa setiap kali berkonsultasi dengan pembimbing dan HARUS dicetak kembali setiap memasuki semester baru melalui SIKAD
3. Saran dan koreksi dari pembimbing harus ditulis dan diparaf oleh pembimbing
4. Setelah skripsi disetujui (ACC) oleh pembimbing, kartu ini harus ditandatangani oleh Wakil Dekan I/ Kepala departemen/Ketua prodi
5. Kartu kendali bimbingan asli yang telah ditandatangani diserahkan kepada Ketua Program Studi dan kopiannya dilampirkan pada skripsi.
6. Jika jumlah pertemuan pada kartu bimbingan tidak cukup dalam satu halaman, kartu bimbingan ini dapat di download kembali melalui SIKAD

SURAT KEPUTUSAN DEKAN FAKULTAS TEKNIK UNIVERSITAS ISLAM RIAU
NOMOR : 0417/KPTS/FT-UIR/2024
TENTANG PENETAPAN DOSEN PENGUJI SKRIPSI MAHASISWA FAK. TEKNIK UNIV. ISLAM RIAU

DEKAN FAKULTAS TEKNIK

Menimbang : 1. Bahwa untuk menyelesaikan studi S.1 bagi mahasiswa Fakultas Teknik Univ. Islam Riau dilaksanakan Ujian Skripsi/Komprehensif sebagai tugas akhir. Untuk itu perlu ditetapkan mahasiswa yang telah memenuhi syarat untuk ujian dimaksud serta dosen penguji.

2. Bahwa penetapan mahasiswa yang memenuhi syarat dan dosen penguji yang bersangkutan perlu ditetapkan dengan Surat Keputusan Dekan.

Mengingat : 1. Undang - Undang Nomor 12 Tahun 2012 Tentang Pendidikan Tinggi
2. Peraturan Presiden Republik Indonesia Nomor 8 Tahun 2012 Tentang Kerangka Kualifikasi Nasional Indonesia
3. Peraturan Pemerintah Republik Indonesia Nomor 37 Tahun 2009 Tentang Dosen
4. Peraturan Pemerintah Republik Indonesia Nomor 66 Tahun 2010 Tentang Pengelolaan dan Penyelenggaraan Pendidikan
5. Peraturan Menteri Pendidikan Nasional Nomor 63 Tahun 2009 Tentang Sistem Penjaminan Mutu Pendidikan
6. Peraturan Menteri Pendidikan dan Kebudayaan Republik Indonesia Nomor 49 Tahun 2014 Tentang Standar Nasional Pendidikan Tinggi
7. Statuta Universitas Islam Riau Tahun 2018
8. Peraturan Universitas Islam Riau Nomor 001 Tahun 2018 Tentang Ketentuan Akademik Bidang Pendidikan Universitas Islam Riau

MEMUTUSKAN

Menetapkan : 1. Mahasiswa Fakultas Teknik Universitas Islam Riau yang tersebut namanya dibawah ini :

Nama	: Syarifah Kusuma Maharani
NPM	: 193510512
Program Studi	: Teknik Informatika
Jenjang Pendidikan	: Strata Satu (S1)
Judul Skripsi	: Perbandingan Metode Machine Learning dalam Klasifikasi Potensi Keluarga Beresiko Stunting Pada Kecamatan Merbau

2. Penguji Skripsi/Komprehensif mahasiswa tersebut terdiri dari :

1. Dr. Arbi Haza Nasution, B.IT., M.IT.	Sebagai Ketua Merangkap Penguji
2. Nesi Syafitri, S.Kom., M.Cs.	Sebagai Anggota Merangkap Penguji
3. Ir. Des Suryani, M.Sc.	Sebagai Anggota Merangkap Penguji

3. Laporan hasil ujian serta berita acara telah sampai kepada Pimpinan Fakultas selambat-lambatnya 1(satu) bulan setelah ujian dilaksanakan.

4. Keputusan ini mulai berlaku pada tanggal ditetapkannya dengan ketentuan bila terdapat kekeliruan dikemudian hari segera ditinjau kembali.

KUTIPAN : Disampaikan kepada yang bersangkutan untuk dapat dilaksanakan dengan sebaik-baiknya.

Ditetapkan di : Pekanbaru
Pada Tanggal : 23 Ramadhan 1445 H
02 April 2024 M

Dekan,

Dr. Deddy Purnomo Retno, S.T., M.T.
NPK : 1005057702

Tembusan disampaikan :
1. Yth. Rektor UIR di Pekanbaru.
2. Yth. Ketua Program Studi Teknik Informatika FT-UIR
3. Yth. Pembimbing dan Penguji Skripsi
3. Mahasiswa yang bersangkutan
5. Arsip

**Surat ini ditandatangani secara elektronik*

YAYASAN LEMBAGA PENDIDIKAN ISLAM (YLPI) RIAU
UNIVERSITAS ISLAM RIAU
FAKULTAS TEKNIK
PROGRAM STUDI TEKNIK INFORMATIKA
Jalan Kaharuddin Nasution No. 113 P. Marpoyan Pekanbaru Riau Indonesia – Kode Pos: 28284
Telp. +62 761 674674 *Website*: www.eng.uir.ac.id *Email*: fakultas_teknik@uir.ac.id

BERITA ACARA UJIAN SKRIPSI

Berdasarkan Surat Keputusan Dekan Fakultas Teknik Universitas Islam Riau, Pekanbaru, tanggal 02 April 2024, Nomor: 0417/KPTS/FT-UIR/2024, maka pada hari Selasa tanggal 02 April 2024, telah dilaksanakan Ujian Skripsi Program Studi Teknik Informatika Fakultas Teknik Universitas Islam Riau, Jenjang Studi S1, Tahun Akademik 2023/2024 berikut ini.

1. Nama : Syarifah Kusuma Maharani
2. NPM 193510512
3. Judul Skripsi : Perbandingan Metode Machine Learning dalam Klasifikasi Potensi Keluarga Beresiko Stunting Pada Kecamatan Merbau
4. Waktu Ujian : 10.00 WIB s.d. Selesai
5. Tempat Pelaksanaan Ujian : Ruang Sidang Fakultas Teknik UIR

Dengan keputusan Hasil Ujian Skripsi:

~~Lulus*~~/ Lulus dengan Perbaikan*/ ~~Tidak Lulus*~~

** Coret yang tidak perlu.*

Nilai Ujian:

Nilai Ujian Angka = 76,20 Nilai Huruf = (A-)

Tim Penguji Skripsi.

No	Nama	Jabatan	Tanda Tangan
1	Dr. Arbi Haza Nasution, B.IT., M.IT.	Ketua	1.
2	Nesi Syafitri, S.Kom., M.Cs.	Anggota	2.
3	Ir. Des Suryani, M.Sc.	Anggota	3.

Panitia Ujian
Ketua,

Dr. Arbi Haza Nasution, B.IT., M.IT.

NIDN.1023048901

Pekanbaru, 02 April 2024
Mengetahui,
Dekan Fakultas Teknik

Dr. Deddy Purnomo Reino, S.T.,M.T.,GP.A-Utama.
NIDN. 090602372

UNIVERSITAS ISLAM RIAU
FAKULTAS TEKNIK
الجامعة الإسلامية الريوية
Alamat: Jalan Kaharuddin Nasution No.113, Marpoyan, Pekanbaru, Riau, Indonesia - 28284
Telp. +62 761 674674 Email: fakultas_teknik@uir.ac.id Website: www.eng.uir.ac.id

SURAT KETERANGAN BEBAS PLAGIAT

Nomor: 125/A-UIR/5-T/2024

Fakultas Teknik Universitas Islam Riau menerangkan bahwa Mahasiswa/i dengan identitas berikut:

Nama	:	**SYARIFAH KUSUMA MAHARANI**
NPM	:	193510512
Program Studi	:	Teknik Informatika
Jenjang Pendidikan	:	Strata Satu (S1)
Judul Skripsi TA	:	PERBANDINGAN METODE MACHINE LEARNING DALAM KLASIFIKASI POTENSI KELUARGA BERESIKO STUNTING PADA KECAMATAN MERBAU

Dinyatakan **Bebas Plagiat,** berdasarkan hasil pengecekan pada Turnitin menunjukkan angka **Similarity Index < 30%** sesuai dengan peraturan Universitas Islam Riau yang berlaku.

Demikian surat keterangan ini dibuat untuk dapat dipergunakan sebagaimana mestinya.

Mengetahui,

Kaprodi. Teknik Informatika

Apri Siswanto, S.Kom., M.Kom., Ph.D

Pekanbaru, 27 March 2024 M
17 Romadhōn 1445 H

Staff Pemeriksa

Khezi Triandini Dafan, S.E