

IMPLEMENTASI ALGORITMA NAÏVE BAYES DALAM PENYEDERHANAAN TEKS UNTUK ANAK DISLEKSIA

Ika Rahmawati^{1*}, Mustafa²,

1,2Teknik Informatikan, Universitas Islam Sultan Agung, Indonesia

[1rahmawatiika959@gmail.com](mailto:rahmawatiika959@gmail.com), [2cakmustafa@unissula.ac.id](mailto:cakmustafa@unissula.ac.id)

Diterima: 10-07- 2025

Direvisi: 20-07-2025

Disetujui: 01-08-2025

ABSTRAK

Disleksia merupakan gangguan belajar spesifik yang ditandai dengan kesulitan membaca, menulis, dan mengeja, serta sering kali memengaruhi kemampuan anak untuk mengikuti pelajaran di sekolah. Permasalahan yang sering terjadi pada anak disleksia meliputi kesulitan membedakan huruf tertentu seperti "b" dan "d" atau "p" dan "q". Penelitian ini bertujuan mengembangkan sistem klasifikasi teks untuk membantu anak-anak dengan disleksia memahami bacaan melalui penyederhanaan teks. Sistem dibangun menggunakan algoritma Multinomial Naïve Bayes yang mengklasifikasikan teks menjadi kategori "sederhana" dan "tidak sederhana". Dataset disusun secara manual dengan mengumpulkan kalimat dari buku dan majalah anak, lalu mengidentifikasi kata atau frasa sulit bagi anak usia 10–12 tahun. Metode penelitian mencakup tahapan preprocessing (tokenisasi, stopword removal, stemming), pelabelan teks, pembagian data latih dan uji (70:30), pelatihan model dengan optimasi parameter α , serta evaluasi menggunakan akurasi, balanced accuracy, precision, recall, dan F1-score. Hasil penelitian menunjukkan akurasi sebesar 93,72% pada data latih dan 92,49% pada data uji, dengan balanced accuracy masing-masing 94,23% dan 92,59%. Nilai precision, recall, dan F1-score rata-rata mencapai 0,92, menandakan kinerja model yang konsisten pada kedua kelas. Visualisasi confusion matrix menunjukkan sebagian besar prediksi tepat, meskipun masih terdapat kesalahan klasifikasi pada teks dengan makna ambigu. Penelitian ini membuktikan bahwa Naïve Bayes efektif dalam mengklasifikasikan tingkat kesederhanaan teks berbahasa Indonesia, serta berpotensi menjadi solusi teknologi pendukung pendidikan inklusif bagi anak-anak dengan disleksia..

Kata kunci : Disleksia, Naïve Bayes, NLP

PENDAHULUAN

Disleksia adalah salah satu kondisi belajar khusus yang menyebabkan kesulitan dalam membaca, menulis, dan mengeja, meskipun individu yang mengalaminya memiliki tingkat kecerdasan yang normal atau bahkan di atas rata-rata. Istilah "disleksia" berasal dari bahasa Yunani, di mana "dys" berarti kesulitan dan "lexia" berarti kata-kata. Gangguan ini memengaruhi sekitar 5–15 persen anak-anak di seluruh dunia, termasuk sekitar 5 juta anak di Indonesia[1]. Dampaknya sangat signifikan pada perkembangan akademik dan psikososial, terutama karena anak disleksia sering kali kesulitan memahami teks dan menghadapi tantangan dalam memori jangka pendek dan konsentrasi. Kesulitan ini dapat mengurangi rasa percaya diri mereka dan memperburuk kesenjangan dalam prestasi akademik [2][3].

Penyebab utama disleksia bersifat genetik, meskipun faktor eksternal seperti kurangnya dukungan orang tua dan sistem pendidikan yang tidak inklusif juga berperan. Anak-anak disleksia memerlukan pendekatan pembelajaran yang berbeda, termasuk teks bacaan yang lebih sederhana dan lebih mudah dipahami. Tanpa upaya untuk menyederhanakan teks, mereka sering kali merasa frustrasi, kehilangan minat belajar, dan tidak mampu mencapai potensi mereka. Dalam konteks ini, penyediaan teks yang sesuai menjadi tantangan penting yang memerlukan solusi inovatif berbasis teknologi[4][5].

Kemajuan dalam teknologi informasi dan komunikasi, terutama di bidang Natural Language Processing (NLP), membuka peluang untuk mengatasi tantangan ini secara lebih efektif dan efisien. Salah satu pendekatan yang menjanjikan adalah penerapan algoritma Naïve Bayes untuk klasifikasi teks [6]. Algoritma ini dikenal karena kemampuannya yang sederhana tetapi efektif dalam mengidentifikasi elemen-elemen dalam teks yang sulit dipahami. Dengan pendekatan probabilistik, Naïve Bayes memungkinkan penyederhanaan teks secara otomatis tanpa mengurangi makna utamanya [7]. Hal ini sangat relevan dalam konteks pendidikan inklusif untuk anak-anak dengan disleksia.

Pada penelitian sebelumnya telah menunjukkan berbagai pendekatan dalam mendeteksi dan menangani gangguan perkembangan serta aplikasi Naïve Bayes dalam klasifikasi data. memanfaatkan metode Naïve Bayes dalam sistem pakar untuk mendiagnosis gangguan perkembangan anak seperti hiperaktivitas dan disleksia dengan tingkat akurasi yang tinggi [8]. Dalam konteks pendidikan, Darmayanti dkk., 2023 mengevaluasi aplikasi Marbel Reading untuk meningkatkan kemampuan membaca anak dengan disleksia [9], sementara Oktamarina tahun 2022 menggunakan metode kualitatif studi kasus untuk menganalisis gangguan disleksia pada anak usia dini, menyoroti pentingnya deteksi dini [10]. Sinaga tahun 2023 mengaplikasikan Naïve Bayes dalam klasifikasi penerima program Indonesia Pintar, mencapai akurasi 88,89% dalam memprediksi kelayakan [11]. Penelitian Lestari dkk. (2024) mengembangkan sistem diagnosis disleksia berbasis Naïve Bayes dengan tingkat akurasi tinggi. Di bidang klasifikasi data Pendidikan [12]. Pebdika dkk. (2023) menggunakan Naïve Bayes untuk mengidentifikasi penerima Program Indonesia Pintar (PIP) dan berhasil mencapai akurasi 94,54% [13]. Selain itu, penelitian terbaru di bidang NLP Bahasa Indonesia menunjukkan bahwa teknik text simplification baik secara leksikal maupun sintaksis dapat membantu meningkatkan keterbacaan bagi pembaca dengan hambatan bahasa, termasuk disleksia.

Berdasarkan uraian tersebut, terdapat beberapa masalah penelitian yang diidentifikasi seperti, Belum adanya sistem otomatis penyederhanaan teks bahasa Indonesia yang ditujukan khusus untuk anak dengan disleksia. Pemanfaatan algoritma Naïve Bayes dalam konteks penyederhanaan teks masih jarang, terutama dalam bahasa Indonesia. Perlunya pengembangan teknologi NLP yang sesuai dengan karakteristik bahasa Indonesia untuk mendukung pendidikan inklusif.

Penelitian ini bertujuan untuk mengembangkan sistem klasifikasi teks berbasis algoritma Naïve Bayes, yang mengkategorikan teks ke dalam dua kategori: "sederhana" dan "tidak sederhana" [14]. Penelitian ini menggunakan teks bacaan berbahasa Indonesia yang ditujukan untuk anak usia sekolah dasar. Pendekatan ini diharapkan dapat membantu anak-anak dengan disleksia memahami teks bacaan secara lebih baik, meningkatkan keterampilan membaca mereka, dan menumbuhkan rasa percaya diri. Selain itu, sistem ini juga memberikan manfaat bagi guru, orang tua, dan penyusun materi pembelajaran dalam menyaring dan mengkurasi teks bacaan secara lebih efektif [15].

Teknologi ini tidak hanya menawarkan solusi langsung bagi anak-anak dengan disleksia tetapi juga berkontribusi pada pengembangan pendidikan inklusif di Indonesia. Implementasi teknologi untuk menyederhanakan teks membantu mendukung upaya pemerintah dalam meningkatkan akses ke pendidikan yang setara bagi semua anak, termasuk mereka yang memiliki kebutuhan khusus. Penelitian sebelumnya, seperti yang dilakukan oleh Yuyun dkk., 2021, menunjukkan bahwa

algoritma Naïve Bayes berbasis web telah berhasil digunakan untuk sistem pakar diagnosis disleksia. Namun, aplikasi teknologi untuk menyederhanakan teks secara otomatis masih sangat jarang, terutama dalam konteks bahasa Indonesia [16].

Dengan mengintegrasikan pendekatan berbasis NLP dan algoritma Naïve Bayes, penelitian ini tidak hanya bertujuan untuk menyediakan solusi praktis bagi anak-anak dengan disleksia tetapi juga memperkuat hubungan antara teknologi, pendidikan inklusif, dan tujuan sosial [17]. Penelitian ini diharapkan dapat menjadi pijakan awal dalam pengembangan teknologi bantu pembelajaran untuk anak berkebutuhan khusus, mendukung inklusivitas, dan mempersempit kesenjangan dalam akses pendidikan [18].

DASAR TEORI DISLEKSIA

Disleksia, berasal dari kata Yunani "dys" yang berarti "sulit dalam" dan "lex" yang berarti "kata", didefinisikan sebagai ketidakmampuan untuk memahami informasi melalui proses pembelajaran. Sebagai sindrom, disleksia menyebabkan kesulitan dalam mempelajari dan mengintegrasikan komponen kata dan kalimat serta waktu, arah, dan masa. Anak-anak yang didiagnosis dengan disleksia mengalami kesulitan membaca karena kelainan syaraf yang terjadi pada otak mereka. Jenis kesulitan belajar membaca bervariasi, tetapi semuanya menunjukkan bahwa ada gangguan dalam fungsi otak. Berpikir linear dan berbicara, anak-anak dengan disleksia memiliki bagian otak kiri yang lebih kecil daripada manusia normal. Inilah yang membedakan pemrosesan data dan kemampuan berbahasa [19][20].

NAÏVE BAYES

Naïve Bayes adalah pengklasifikasi probabilitas sederhana yang menghasilkan sekumpulan kemungkinan dengan menjumlahkan frekuensi yang menggabungkan nilai dari dataset tertentu. Algoritma ini mengambil semua karakteristik independen atau tidak saling ketergantungan yang diberikan oleh nilai variable kelas, dan menggunakan teorema Bayes untuk melakukannya. Nilai penyederhanaan, yang didasarkan pada Naïve Bayes, menyatakan bahwa jika nilai output diberikan kepada atribut, nilai-nilai tersebut secara kondisional tidak berhubungan satu sama lain. Keuntungan dari Metode Naïve Bayes hanya membutuhkan jumlah data pelatihan yang relatif kecil untuk menentukan perkiraan parameter yang diperlukan untuk proses pengklasifikasian [21][22].

Multinomial Naïve Bayes

Model Multinomial Naïve Bayes, yang merupakan pengembangan dari algoritma Bayes, digunakan untuk mengklasifikasikan teks atau dokumen. Rumus klasifikator Multinomial Naïve Bayes menghitung kedua jumlah kemunculan kata untuk menentukan kelas dokumen [23].

$$p(c, d) = \frac{N_c}{N} x P(t_1, C) x \dots x P(t_n, C)$$

Keterangan :

$p(c, d)$: Probabilitas suatu dokumen termasuk kelas c.

N_c : Jumlah kelas c pada seluruh dokumen.

N : Jumlah seluruh dokumen.

t_n : Kata dokumen d ke-n.

$P(t_n, C)$: Probabilitas kata ke-n dengan diketahui kelas c.

Rumus probabilitas kata ke-n yang digunakan dengan pembobotan kata TF-IDF dapat dilihat pada keterangan dibawah ini.

$$P(t_n, C) = \frac{W_{ct} + 1}{(\sum W'_{ct} + B')}$$

Keterangan :

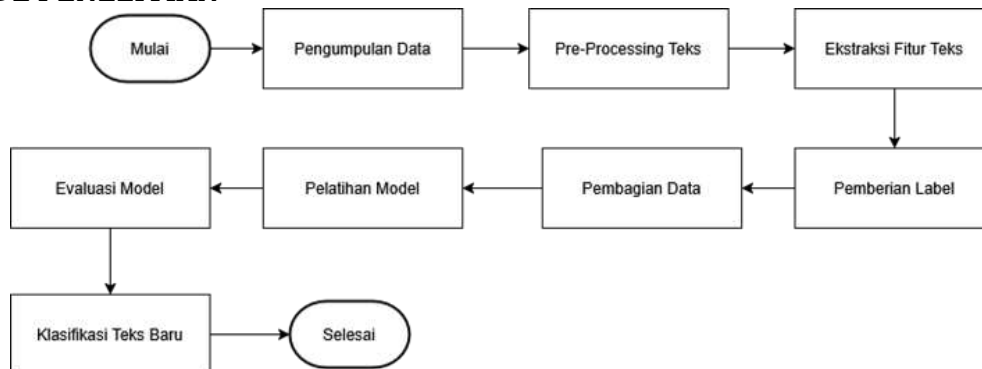
W_{ct} : Nilai pembobotan tfidf atau W dari term t di dalam kategori c .

$\sum W'_{ct}$: Jumlah total W dari keseluruhan term pada kategori c .

B' : Jumlah W kata unik (nilai IDF tidak dikali dengan tf) pada seluruh dokumen.

Salah satu metode pembelajaran probabilistic yang digunakan dalam pengolahan Natural Language Processing (NLP) adalah algoritma multinomial Naïve Bayes yang didasarkan pada teorema Bayes [24]. Algoritma ini bergantung pada konsep "frekuensi", yaitu berapa kali kata tersebut muncul dalam sebuah dokumen. Model ini menjelaskan dua fakta: apakah kata tersebut muncul atau tidak, dan frekuensi kemunculannya dalam dokumen. Multinomial Naïve Bayes adalah model Naïve Bayes yang paling umum digunakan dalam klasifikasi teks.

METODE PENELITIAN



Gambar 1 Flowchart Metode Penelitian

1. Pengumpulan Data

Pada tahap awal penelitian ini, proses pengumpulan data dilakukan terhadap teks bacaan anak yang akan dianalisis tingkat kesederhanaannya. 500 kalimat teks yang digunakan berasal dari berbagai sumber, termasuk buku bacaan anak sekolah dasar, artikel pendidikan anak, dan konten pembelajaran dasar dari situs web edukatif yang tersedia secara online. Sumber-sumber ini dipilih karena memiliki gaya bahasa yang relevan untuk anak-anak, memiliki variasi dalam struktur kalimat dan tingkat kompleksitas kosakata yang berbeda. Setelah data dikumpulkan, setiap teks diberi label manual sebagai "Sederhana" atau "Tidak Sederhana". Kriteria yang digunakan untuk pelabelan ini adalah kesederhanaan bahasa. Kriteria ini termasuk penggunaan kosakata umum, struktur kalimat yang sederhana, dan kemudahan pembacaan bagi anak-anak, terutama mereka yang mengalami disleksia. Proses ini sangat penting untuk memastikan bahwa data yang digunakan untuk pelatihan dan pengujian model klasifikasi dengan algoritma Naïve Bayes tepat dan memenuhi kebutuhan pengguna akhir.

2. Preprocessing Teks

Preprocessing teks adalah langkah penting dalam Natural Language Processing (NLP). Tujuan dari langkah ini adalah untuk membersihkan dan menyiapkan data teks sehingga model pembelajaran mesin dapat lebih mudah menganalisisnya. Beberapa tahapan utama preprocessing dalam penelitian ini termasuk setword, tokenisasi, dan stemming[25].

a. Tokenisasi

Tokenisasi, proses membagi teks yang dibaca menjadi bagian yang lebih kecil, seperti kalimat, adalah langkah pertama dalam praproses data. Ini dilakukan untuk membuat analisis dan pemrosesan data lebih mudah.

b. Stemming

Stemming dilakukan untuk mengubah kata menjadi dasar atau akar katanya. Misalnya "memahami" menjadi "paham."

c. Stopword Removal

Tahapan preprocessing ini menghasilkan data teks yang lebih bersih dan terstruktur. Ini memungkinkan model klasifikasi untuk lebih baik mendeteksi dan menyederhanakan teks bacaan secara lebih akurat.

3. Pemberian Label

Label berfungsi sebagai penanda kategori untuk setiap teks yang digunakan, sehingga tahap pemberian label merupakan langkah penting dalam membangun model klasifikasi. Dalam penelitian ini, setiap teks yang dibaca setelah proses preprocessing akan diberi label "Sederhana" atau "Tidak Sederhana". Teks yang diberi label "Sederhana" memiliki struktur kalimat yang mudah dipahami, kosakata umum, dan sesuai dengan kemampuan pemahaman anak disleksia. Namun, label "Tidak Sederhana" digunakan untuk teks yang memiliki struktur kalimat yang kompleks, penggunaan kata-kata yang sulit atau teknis, dan makna yang cenderung ambigu atau sulit dipahami. Kriteria kebahasaan dan kesesuaian tingkat keterbacaan digunakan untuk melakukan proses pelabelan ini secara manual. Tahap ini akan menghasilkan data latih dan data uji yang akan digunakan untuk mempelajari pola bahasa yang membedakan teks sederhana dan tidak sederhana.

4. Pembagian Data

Data kemudian dibagi menjadi dua bagian utama setelah proses pelabelan selesai: data latihan (training data) dan data uji (testing data). Tujuan dari bagian ini adalah untuk membedakan data yang digunakan untuk membangun model (melatih algoritma) dari data yang digunakan untuk menguji atau mengevaluasi kinerja model yang dilatih. Pembagian data ini biasanya dilakukan secara rasional; misalnya, dua puluh persen untuk data uji dan delapan puluh persen untuk data latih. Algoritma Naïve Bayes dapat membedakan teks sederhana dan tidak sederhana dengan menggunakan data latih untuk mengenalkan pola dan karakteristik teks. Sementara itu, data uji digunakan untuk mengevaluasi kemampuan model untuk mengklasifikasikan data baru yang belum pernah dilihat sebelumnya. Pembagian data yang tepat sangat penting untuk memastikan bahwa model tidak hanya berfungsi dengan baik pada data pelatihan tetapi juga dapat generalisasi dengan baik pada data di luar pelatihan.

5. Pelatihan Model

Pada tahap pelatihan (training) model, algoritma Naïve Bayes digunakan untuk membuat model klasifikasi dari data latihan yang telah diproses sebelumnya. Semua teks dalam data latih telah ditransformasi menjadi representasi numerik, biasanya dalam bentuk vektor fitur dengan metode seperti TF-IDF. Selanjutnya, model Naïve Bayes menganalisis kemungkinan setiap kata muncul dalam masing-masing kelas label, "sederhana" dan "tidak sederhana". Sistem menggunakan Teorema Bayes untuk menghitung kemungkinan suatu teks akan dimasukkan ke dalam kelas tertentu berdasarkan frekuensi kata-kata yang muncul dalam kelas tersebut di data latihan. Proses pelatihan ini berdasarkan prinsip ini. Metode ini dikenal sebagai "Naïve" Bayes karena menggunakan asumsi independensi antar fitur (kata). Hasil dari tahap pelatihan

ini adalah model klasifikasi, yang dapat digunakan untuk mengklasifikasikan data uji atau data baru setelah menemukan pola dalam teks untuk menentukan apakah kalimat atau teks termasuk kategori sederhana atau tidak.

6. Evaluasi Model

Dalam penelitian ini, evaluasi model dilakukan untuk mengukur kinerja algoritma klasifikasi yang digunakan. Tiga metrik utama digunakan: akurasi, precision, dan recall. Akurasi mengukur persentase prediksi yang benar dari keseluruhan data yang diuji, memberikan gambaran umum seberapa baik model mengenali data secara keseluruhan. Precision mengukur ketepatan prediksi positif, yaitu seberapa banyak dari hasil prediksi positif yang benar atau relevan. Sementara recall menilai kemampuan model dalam menemukan semua data positif yang sebenarnya akurat. Dengan menggunakan ketiga metrik ini, evaluasi model menjadi lebih komprehensif baik dari segi ketepatan maupun kelengkapan hasil prediksi. Dengan demikian, menjadi lebih mudah untuk menilai keandalan model pada tugas klasifikasi teks bacaan yang diberikan kepada anak-anak yang didiagnosis dengan disleksia.

7. Implementasi

Agar model dapat digunakan secara nyata, implementasi adalah proses menerapkan model setelah proses pelatihan dan evaluasi dengan menggunakan Flask. Sistem ini dibuat untuk menerima teks sebagai masukan, memprosesnya menggunakan model klasifikasi yang telah dilatih, dan menghasilkan kategori keterbacaan serta rekomendasi untuk perbaikan jika diperlukan. Tahapan ini tidak hanya menunjukkan kinerja teknis sistem, tetapi juga menguji seberapa baik model dapat memberikan hasil yang relevan dan bermanfaat dalam konteks penggunaannya. Tahap ini sekaligus menandai implementasi nyata dari teknik yang telah dikembangkan. Selain itu, itu menunjukkan bahwa teknik itu efektif dalam menyelesaikan masalah klasifikasi teks bagi anak-anak yang didiagnosis dengan disleksia.

HASIL DAN DISKUSI

Penelitian ini bertujuan untuk mengembangkan sistem klasifikasi teks berbasis algoritma Naïve Bayes untuk membantu anak-anak disleksia memahami teks bacaan yang sesuai dengan tingkat kesederhanaan mereka. Pada bagian ini, hasil dari penelitian dijelaskan mulai dari tahap pengumpulan data hingga analisis data yang diperoleh.

1. Pengumpulan Data

Proses pengumpulan dataset pada penelitian ini dilakukan secara manual karena belum tersedia dataset yang sesuai dengan kebutuhan penelitian. Dataset ini disusun dengan cara mereview beberapa buku anak, dan majalah anak. Kemudian memilih beberapa kalimat yang sulit bagi anak-anak untuk memahami disleksia yang berusia 10-12 tahun.

Tahapan pengumpulan data dilakukan sebagai berikut:

- Mengumpulkan beberapa buku anak, majalah anak, atau juga bisa dari E
- Book Mereview buku anak dan majalah anak, proses pengumpulan dimulai dengan mereview kalimat yang sulit dipahami. Setiap kalimat memuat pemahaman sulit dan solusinya dicatat untuk membuat dataset.

Tabel 1 Contoh Hasil Pengumpulan Dataset

NO	Teks Asli	Kata Sulit
1	Benar, Kanda. Semoga ia menjadi putra yang baik	Kanda, ia, putra
2	Ia sadar bahwa tingkah lakunya kemarin tidak seperti anak yang mandiri	Tingkah laku, mandiri
3	Buku cerita itu penuh dengan petualangan yang menarik	petualangan
4	Dia merencanakan untuk pergi berlibur ke tempat yang jauh	Merencanakan, berlibur
5	iklim di samarinda ini cukup panas, pada dasarnya beriklim tropis	iklim; pada dasarnya; beriklim tropis

2. Preprocessing Teks

Beberapa tahapan utama preprocessing dalam penelitian ini termasuk tokenisasi, stopwords removal, dan stemming

```
# Initialize stopword remover, stemmer, tokenizer
stopword_factory = StopWordRemoverFactory()
# Removed 'itu', 'ini' from stopwords as they might be important for readability level
stopwords = stopword_factory.get_stop_words() + ['yang', 'di', 'ke', 'dan', 'dari', 'pada', 'untuk', 'tidak']
stemmer = StemmerFactory().create_stemmer()
tokenizer = RegexpTokenizer(r'\w+')
```

Gambar 2 Preprocessing Teks

Gambar 2 menunjukkan langkah-langkah awal pengolahan teks dengan Python, terutama untuk teks berbahasa Indonesia. Stopword remover, stemmer, dan tokenizer adalah tiga komponen penting yang digunakan. Pertama, objek `stopword_factory` digunakan untuk mengakses daftar kata umum, atau stopwords, seperti "yang", "di", "ke", dan sebagainya, yang biasanya dihapus karena tidak penting. Meskipun demikian, kata-kata seperti "itu" dan "ini" disimpan karena dianggap penting untuk menjaga makna kalimat tetap jelas. Oleh karena itu, objek `stemmer` dibuat untuk mengubah kata ke bentuk aslinya, seperti "berlari" menjadi "berlari", sehingga analisis teks menjadi lebih konsisten. Terakhir, `tokenizer` digunakan untuk memecah teks menjadi kata-kata terpisah dengan menggunakan pola tertentu, hanya mengambil kata yang terdiri dari huruf dan angka. Proses ini membantu mempersiapkan teks sebelum dianalisis lebih lanjut.

3. Pemberian Label

Labeling adalah proses menandai atau mengklasifikasikan data teks ke dalam kategori tertentu sesuai tujuan analisis. Dalam langkah ini, setiap data teks diberi label yang menunjukkan kelas atau output yang diinginkan, seperti "positif" atau "negatif" untuk analisis sentimen atau "spam" dan "tidak spam" untuk klasifikasi email. Jika data yang telah dilabeli sebelumnya tersedia, label dapat diberikan secara otomatis atau manual oleh manusia.

```
# Prepare texts and labels
texts = []
labels = []
numeric_features_list = []

for index, row in df.iterrows():
    for label, col in [(0, 'Teks Asli'), (1, 'Teks Sederhana')]:
        text = str(row[col])
        processed, ratio, count, avg_len, sent_count, avg_sent_len, noun_r, verb_r, fk, syll = preprocess_text(text)
        if processed:
            texts.append(processed)
            labels.append(label)
            # Make sure the number of numerical features is consistent with the preprocess_text output
            numeric_features_list.append([
                ratio, count, avg_len, sent_count, avg_sent_len, noun_r, verb_r, fk, syll
```

Gambar 3 Labeling

Gambar 3 menampilkan potongan kode Python yang merupakan bagian dari proses persiapan data untuk pelatihan model, terutama pada tahap pemberian label dan ekstraksi fitur teks. Kode ini melabelkan dua jenis teks, "Teks Asli" dan "Teks Sederhana", masing-masing dengan label 0 dan 1. Proses ini dilakukan melalui perulangan for. Untuk setiap baris data (row) dalam DataFrame, sistem membaca kolom tertentu dan mengubah isi kolom tersebut menjadi string. Selanjutnya, teks diproses melalui fungsi `preprocess_text`, yang mengembalikan versi teks yang telah dibersihkan bersama dengan beberapa fitur numerik, seperti rasio kata, panjang rata-rata, jumlah kalimat, dan rasio jenis kata tertentu. Jika hasil pra-proses valid, teks yang diproses akan dimasukkan ke dalam daftar teks, labelnya akan dimasukkan ke dalam daftar label, dan daftar fitur numerik akan dimasukkan ke dalam daftar fitur numerik. Oleh karena itu, gambar tersebut menunjukkan cara sistem menghubungkan setiap teks dengan label kategorikal dan fitur numerik untuk keperluan pelatihan model klasifikasi.

4. Pembagian Data

Data dibagi menjadi data pelatihan dan pengujian melalui proses pembagian data. Dalam kode ini, fungsi `train_test_split` membagi data fitur (`X_selected`) dan label (`y_res`) menjadi 30% data pengujian dan 70% data pelatihan. Untuk memastikan hasil pembagian tetap konsisten, parameter `random_state=42` digunakan, dan `stratify=y_res` memastikan distribusi label tetap seimbang di kedua bagian data. Variabel `X_train`, `X_test`, `y_train`, dan `y_test` menyimpan hasilnya.

5. Pelatihan Model

Proses pelatihan model menggunakan algoritma Multinomial Naive Bayes ditunjukkan pada gambar. Pertama, proses pencarian parameter terbaik menghasilkan nilai parameter `alpha`, yang kemudian disimpan ke dalam variabel `clf`. Selanjutnya, metode `fit` digunakan untuk melatih model dengan data latih `X_train` dan `y_train`. Dengan proses ini, model dapat mempelajari pola dari data latih untuk melakukan prediksi pada data baru.

```
# Train final model
clf = MultinomialNB(alpha=best_alpha)
clf.fit(X_train, y_train)
```

Gambar 4 Code Pelatihan Model

Gambar 4 menunjukkan proses pelatihan model akhir menggunakan algoritma Multinomial Naive Bayes, yang merupakan salah satu algoritma klasifikasi yang paling umum digunakan untuk menangani masalah klasifikasi dengan data diskret dan dalam pemrosesan data teks. Pada titik ini, objek model `clf` dibuat menggunakan konstruktor `MultinomialNB()` dengan parameter `alpha` yang telah ditetapkan sebelumnya melalui proses pencarian nilai terbaik—biasanya menggunakan metode seperti pencarian grid atau cross-validation. Nilai `alpha` berfungsi sebagai parameter regularisasi yang

menyesuaikan kekuatan pembobotan fitur model untuk mencegah overfitting. Setelah membuat model, langkah berikutnya adalah melakukan pelatihan dengan memanggil fungsi.fit() menerima dua argumen utama: X_train sebagai data fitur dan y_train sebagai label atau target data pelatihan. Dengan menggunakan proses pelatihan ini, model dapat mempelajari hubungan antara label dan fitur sehingga dapat mengidentifikasi pola-pola tertentu yang akan digunakan untuk membuat prediksi tentang data baru. Dengan kata lain, model membuat representasi matematis dari data berdasarkan distribusi nilai fitur yang terkait dengan masing-masing kelas dalam label.

6. Evaluasi

Proses evaluasi model yang sederhana namun menyeluruh digambarkan di sini. Untuk mengetahui seberapa baik model mempelajari data pelatihan, model pertama melakukan prediksi terhadap data latihan (X_train), dan hasilnya dibandingkan dengan label asli untuk menghitung akurasi dan akurasi yang seimbang. Selanjutnya, model menggunakan probabilitas untuk memprediksi data uji (X_test), dan kemudian menerapkan ambang batas (threshold) sebesar 0.6 untuk menentukan label akhir. Ini membantu dalam mengatur sensitivitas model terhadap kelas yang spesifik. Terakhir, model diuji pada data yang diuji. Berbagai metrik, termasuk akurasi, akurasi yang seimbang, skor F1, matriks kekacauan, dan laporan klasifikasi, memberikan penilaian menyeluruh tentang kinerja model, terutama untuk data yang mungkin tidak seimbang.

Tabel 2 Hasil akurasi dan balanced accuracy model

Dataset	Akurasi	Balanced Accuracy
Data Latih	0.9372	0.9423
Data Uji	0.9249	0.9259

Tabel 2 menunjukkan hasil evaluasi model berdasarkan dua metrik utama, akurasi yang seimbang dan akurasi, baik untuk data latihan maupun uji. Tabel menunjukkan akurasi 0,9372 untuk data latihan dan 0,9249 untuk data uji, yang menunjukkan bahwa model mampu memprediksi dengan benar lebih dari 93% data pelatihan dan sekitar 92% data pengujian, menunjukkan performa yang konsisten. Di sisi lain, akurasi seimbang ditemukan pada kedua data uji dan latihan, masing-masing dengan nilai 0,9423 dan 0,9259, masing-masing. Ini menunjukkan bahwa model tidak hanya akurat secara keseluruhan tetapi juga adil dalam memprediksi masing-masing kelas, dengan nilai yang cukup tinggi pada kedua dataset ini. Oleh karena itu, model yang dibangun tidak menunjukkan indikasi overfitting yang signifikan dan memiliki kemampuan generalisasi yang baik.

Selain akurasi, evaluasi model juga dilakukan dengan menggunakan metrik tambahan seperti ketepatan, recall, dan skor F1. Metrik-metrik ini sangat penting jika data memiliki distribusi kelas yang tidak seimbang karena dapat memberikan gambaran lebih detail tentang bagaimana model berfungsi terhadap masing-masing kelas. Hasil evaluasi lengkap berdasarkan metrik tersebut disajikan dalam Tabel 3.

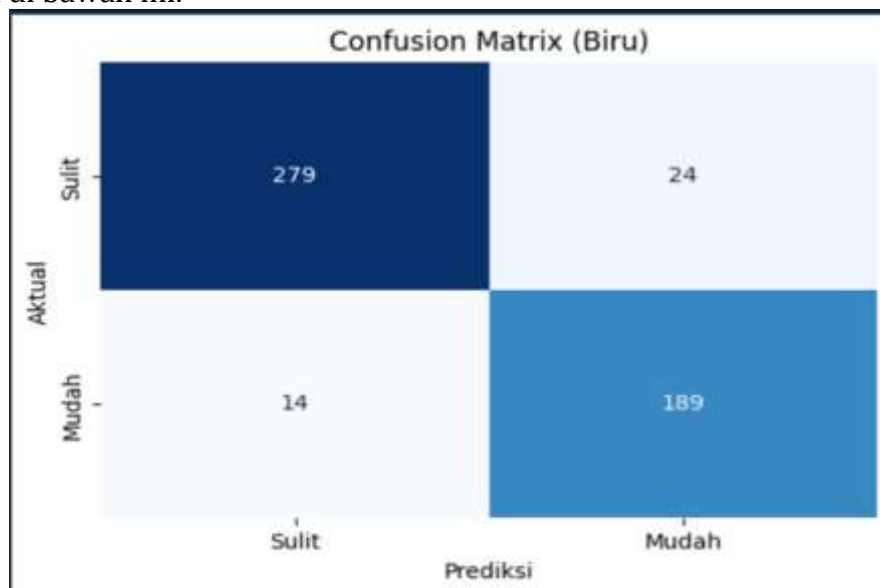
Tabel 3 Hasil precision, recall dan f1-score

evaluasi	precision	recall	f1-score	support
0	0.95	0.92	0.94	303
1	0.89	0.93	0.91	203

evaluasi	precision	recall	f1-score	support
accuracy	-	-	0.92	506
macro avg	0.92	0.93	0.92	506
weighted avg	0.93	0.92	0.93	506

Tabel 3 menunjukkan hasil evaluasi model berdasarkan metrik precision, recall, dan F1-score untuk masing-masing kelas. Untuk kelas 0, model memiliki precision sebesar 0,95, yang berarti bahwa 95% dari semua prediksi yang dianggap sebagai kelas 0 benar; recall sebesar 0.92, yang berarti bahwa model berhasil menangkap 92 persen dari semua data kelas 0, dan nilai F1-nya sebesar 0.94 menunjukkan keseimbangan yang baik antara precision dan recall. Untuk kelas 1, precision 0.89 dan recall 0.93, yang menghasilkan F1-score 0.91, menunjukkan bahwa model juga cukup baik dalam mengenali kelas 1, meskipun precision-nya sedikit lebih rendah dari kelas 0. Secara keseluruhan, akurasi model adalah 0.92, yang menunjukkan bahwa model dapat memprediksi dengan benar 92 persen dari total data uji. Nilai metrik makro rata-rata untuk precision, recall, dan F1-score masing-masing adalah 0.92, yang menunjukkan performa rata-rata tanpa mempertimbangkan proporsi kelas. Nilai metrik berat yang memperhitungkan jumlah data dari masing-masing kelas menunjukkan nilai yang serupa, yaitu 0.92 untuk precision dan F1-score, dan 0.92 untuk recall.

Selain itu, sebagai bagian dari proses evaluasi, visualisasi Matriks Konflik dilakukan untuk melihat bagaimana model bekerja dalam mengklasifikasikan data secara lebih rinci. Matriks Konflik menunjukkan jumlah prediksi yang benar dan salah untuk masing-masing kelas, sehingga lebih mudah untuk menemukan jenis kesalahan yang paling sering dilakukan oleh model. Confusion matrix digambarkan dalam skema warna biru di bawah ini.



Gambar 5 Confusion Matrix

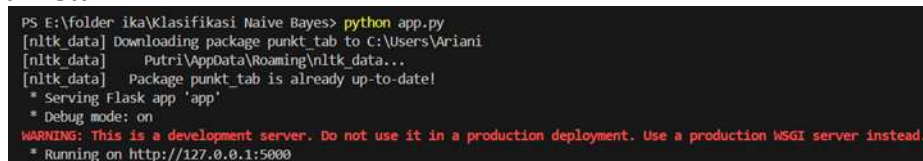
Gambar 5 Terbukti bahwa model mengklasifikasikan 279 data kelas Sulit secara benar dan 189 data kelas Mudah secara benar; namun, 24 data kelas Sulit yang diprediksi sebagai Mudah dan 14 data kelas Mudah yang diprediksi sebagai Sulit mengalami kesalahan klasifikasi. Hasil ini menunjukkan bahwa model cenderung lebih sering keliru dalam mengklasifikasikan data kelas Sulit daripada data kelas Mudah, tetapi secara umum jumlah prediksi yang benar cukup besar. Hasil metrik sebelumnya

seperti ketepatan dan recall didukung oleh visualisasi ini. Ini juga memperjelas jalur kesalahan dalam proses prediksi.

7. Implementasi

Sangat penting untuk memahami bahwa tahap ini merupakan langkah akhir dalam pengembangan sistem klasifikasi teks sebelum membahas lebih lanjut tentang proses implementasi. Agar model dapat digunakan secara nyata, implementasi adalah proses menerapkan model setelah proses pelatihan dan evaluasi. Sistem ini dibuat untuk menerima teks sebagai masukan, memprosesnya menggunakan model klasifikasi yang telah dilatih, dan menghasilkan kategori keterbacaan serta rekomendasi untuk perbaikan jika diperlukan. Tahapan ini tidak hanya menunjukkan kinerja teknis sistem, tetapi juga menguji seberapa baik model dapat memberikan hasil yang relevan dan bermanfaat dalam konteks penggunaannya. Tahap ini sekaligus menandai implementasi nyata dari teknik yang telah dikembangkan. Selain itu, itu menunjukkan bahwa teknik itu efektif dalam menyelesaikan masalah klasifikasi teks bagi anak-anak yang didiagnosis dengan disleksia.

Setelah pengembangan dan pengujian sistem klasifikasi teks selesai, langkah selanjutnya adalah memulai aplikasi untuk digunakan oleh pengguna. Perintah terminal digunakan untuk melakukan proses ini, yang mengeksekusi file utama program berbasis framework Flask. Eksekusi ini akan memulai server lokal, yang memungkinkan pengguna mengakses antarmuka melalui web browser dan mencoba fitur klasifikasi teks secara instan.



```
PS E:\folder_ika\Klasifikasi Naive Bayes> python app.py
[nltk_data] Downloading package punkt_tab to C:\Users\Ariani
[nltk_data] Putri\AppData\Roaming\nltk_data...
[nltk_data] Package punkt_tab is already up-to-date!
* Serving Flask app 'app'
* Debug mode: on
WARNING: This is a development server. Do not use it in a production deployment. Use a production WSGI server instead.
* Running on http://127.0.0.1:5000
```

Gambar 6 Tamapilan Terminal

Gambar 6 menunjukkan bagaimana terminal menjalankan aplikasi klasifikasi teks berbasis Flask. Aplikasi dimulai dengan perintah `python app.py`, yang secara otomatis mengunduh dan memuat paket seperti `punkt_tab` dari pustaka NLTK. Server Flask akan berjalan dalam mode pengembangan (`debug mode: on`) setelah semua dependensi selesai. Baris terakhir tampilan menunjukkan bahwa aplikasi dijalankan dengan berhasil dan Anda dapat mengaksesnya melalui alamat lokal `http://127.0.0.1:5000`. Alamat ini merupakan endpoint antarmuka pengguna, di mana Anda dapat menulis teks dan melihat hasil klasifikasi.

Setelah pengembangan model klasifikasi teks menggunakan metode Naive Bayes selesai, langkah berikutnya adalah memasukkan model ke dalam UI yang interaktif dan mudah digunakan. Tujuan implementasi antarmuka ini adalah untuk membuat sistem dapat diakses secara langsung oleh pengguna yang tidak membutuhkan pemahaman teknis yang mendalam tentang bagaimana model di balik layar berfungsi. Dengan adanya tampilan antarmuka ini, proses klasifikasi teks menjadi lebih praktis dan bermanfaat, sehingga menjadi lebih mudah untuk menguji, mengevaluasi, dan menggunakan sistem sesuai dengan tujuan awal, yaitu membantu anak-anak dengan disleksia memahami teks dengan mudah.



Gambar 7 Tampilan Halaman Awal

Gambar 7 Tampilan awal antarmuka sistem klasifikasi yang dibuat dengan framework Flask sebagai backend dan HTML/CSS untuk frontend ditunjukkan di atas. Di bagian atas antarmuka, Anda akan menemukan identitas pengembang sistem dan judul aplikasi, "Penerapan Naive Bayes untuk Klasifikasi Penyederhanaan Teks Bacaan Anak dengan Disleksia". Pengguna memiliki kolom input di bawah judul, yang memungkinkan mereka memasukkan teks bacaan yang ingin dianalisis. Sistem akan memproses teks dan mengklasifikasikannya berdasarkan model yang telah dilatih setelah pengguna memasukkan teks ke dalam kotak yang tersedia dan menekan tombol "Analisis". Output sistem akan ditampilkan dalam bentuk kategori keterbacaan, seperti mudah dipahami, perlu disederhanakan, atau sulit dipahami. Selain itu, jika ditemukan kata-kata yang dianggap sulit, akan diberikan rekomendasi untuk menyederhanakannya. Tampilan ini menunjukkan bahwa sistem telah diimplementasikan dengan sukses dalam bentuk aplikasi berbasis web yang sederhana namun efektif. Tujuan akhirnya adalah untuk membuat hasil klasifikasi lebih mudah bagi pengguna akhir.

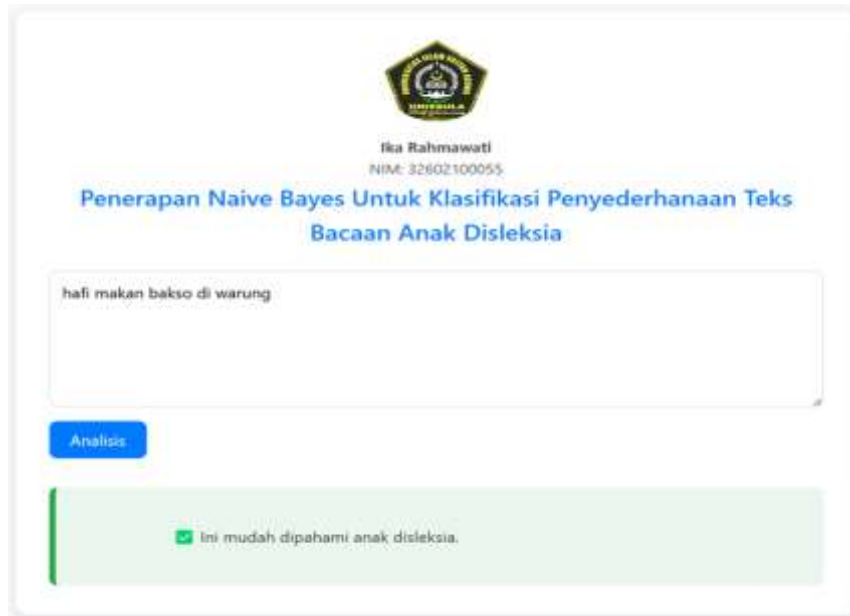
Tahap penting berikutnya adalah menguji sistem dengan berbagai teks masukan setelah pembuatan model klasifikasi dan desain antarmuka sistem. Tujuan dari ujian ini adalah untuk mengetahui apakah sistem dapat melaksanakan fungsinya dengan mengklasifikasikan teks berdasarkan tingkat keterbacaan dan, jika diperlukan, memberikan rekomendasi untuk menyederhanakan kata. Tahapan ini juga menunjukkan sejauh mana sistem dapat memahami konteks kata yang mungkin sulit dipahami oleh anak-anak dengan disleksia dan menggantinya dengan istilah yang lebih familiar. Selain itu, penggunaan ini berfungsi sebagai penghubung antara konsep yang dibangun secara teoritis dengan contoh dunia nyata yang dapat digunakan



Gambar 8 Tampilan Hasil Klasifikasi Sulit dan Saran

Gambar 8 Bagaimana sistem memberikan hasil klasifikasi untuk teks yang dimasukkan oleh pengguna digambarkan di atas. Dalam contoh ini, pengguna memasukkan kalimat "kanda sangat marah jika kamu melanggar aturannya" ke dalam kolom analisis. Setelah mengklik tombol "Analisis", sistem menggunakan model Naive Bayes yang telah dilatih sebelumnya untuk memproses teks, dan kemudian menampilkan hasil klasifikasinya. Hasil sistem menunjukkan bahwa anak-anak dengan disleksia mungkin kesulitan memahami kalimat. Ikon peringatan (\triangle) dan latar berwarna kuning menunjukkan hal ini. Untuk membantu menyederhanakan teks, sistem juga menyarankan penggantian kata: "kanda" diganti dengan "kakak", dan "sangat" diganti dengan "paling". Ini menunjukkan bahwa sistem tidak hanya dapat menemukan kata-kata yang sulit, tetapi juga dapat menawarkan kata-kata alternatif yang lebih umum dan mudah dipahami oleh pengguna tujuan. Tampilan ini menunjukkan bagaimana integrasi antara backend (model klasifikasi) dan frontend (antarmuka pengguna) berhasil, dan menunjukkan bagaimana sistem dapat menghasilkan output yang bermanfaat dalam konteks inklusif pendidikan. Meskipun sistem ini dapat digunakan secara online melalui server lokal (<http://127.0.0.1:5000>), dapat dengan mudah dikembangkan lebih lanjut.

Setelah tahap pengembangan dan pengujian model klasifikasi selesai, sistem diuji secara langsung untuk mengetahui kemampuan sistem untuk menghasilkan hasil klasifikasi teks yang akurat. Berbagai variasi kalimat, baik yang sederhana maupun yang mengandung kosakata kompleks, dimasukkan dalam pengujian ini. Proses ini dilakukan untuk memastikan bahwa sistem tidak hanya dapat bekerja di lingkungan pengembangan secara teoritis, tetapi juga dapat menangani input teks yang relevan dengan anak-anak dengan disleksia. Dengan kata lain, fase ini berfungsi sebagai penghubung antara kemajuan teknologi dan keuntungan praktisnya dalam mendukung kebutuhan pendidikan yang lebih inklusif.



Gambar 9 Tampilan Hasil Klasifikasi Teks yang Mudah

Dalam gambar 9, sistem klasifikasi keterbacaan teks yang berbasis metode Naive Bayes menunjukkan hasilnya. Pengguna memasukkan kalimat "hafi makan bakso di warung" ke dalam kolom input teks, dan kemudian menekan tombol "Analisis". Sistem memproses teks dan menghasilkan penilaian keterbacaan berdasarkan model klasifikasi yang telah dilatih sebelumnya. Sistem menunjukkan bahwa kalimat ini mudah dipahami oleh anak disleksia dalam kasus ini, seperti yang ditunjukkan oleh notifikasi berwarna hijau dengan ikon centang. Ini menunjukkan bahwa pembaca anak disleksia tidak menemukan kosakata atau struktur kalimat yang sulit. Kalimat ini dinilai sederhana, menggunakan kata-kata umum, dan memiliki struktur yang jelas dan mudah dipahami. Tampilan ini menunjukkan bahwa sistem dapat mengenali teks yang memenuhi persyaratan keterbacaan tertentu. Ini juga menunjukkan bahwa klasifikasi tidak sekadar mendeteksi masalah, tetapi juga dapat memberikan umpan balik positif terhadap kalimat yang sesuai. Sebagai bagian dari umpan balik pembelajaran yang membangun, fitur ini penting, terutama bagi pendidik atau orang tua yang ingin memastikan teks yang mereka gunakan sesuai untuk anak dengan kebutuhan khusus.

KESIMPULAN

Berdasarkan hasil penelitian, sistem klasifikasi teks berbasis algoritma Naive Bayes yang dikembangkan berhasil mengelompokkan teks bacaan berbahasa Indonesia ke dalam dua kategori, yaitu sederhana dan tidak sederhana, sesuai dengan tujuan penelitian. Sistem ini membantu proses identifikasi teks yang sesuai untuk anak-anak dengan disleksia, sehingga memudahkan pemahaman bacaan tanpa mengurangi makna dasar. Dengan demikian, keterampilan membaca dan rasa percaya diri anak diharapkan meningkat.

Hasil evaluasi menunjukkan bahwa model memiliki akurasi 93,72% dan balanced accuracy 94,23% pada data latih, serta akurasi 92,49% dan balanced accuracy 92,59% pada data uji, menandakan kinerja yang konsisten tanpa indikasi overfitting yang signifikan. Metrik tambahan pada data uji juga mendukung performa tersebut, dengan precision, recall, dan F1-score rata-rata makro masing-masing sebesar 0,92. Untuk kelas sederhana (label 0), model mencapai precision 0,95 dan recall 0,92, sedangkan untuk kelas tidak sederhana (label 1) precision 0,89 dan recall 0,93.

Visualisasi confusion matrix menunjukkan bahwa model mengklasifikasikan dengan benar 279 data kelas tidak sederhana dan 189 data kelas sederhana, dengan kesalahan klasifikasi sebanyak 24 data tidak sederhana yang diprediksi sebagai sederhana dan 14 data sederhana yang diprediksi sebagai tidak sederhana. Meskipun kesalahan lebih sering terjadi pada kelas tidak sederhana, secara umum jumlah prediksi benar cukup tinggi.

Sistem ini berpotensi diterapkan pada platform web dan dapat menjadi alat efektif bagi guru, orang tua, serta penyusun materi pembelajaran untuk menyaring teks bacaan bagi anak-anak disleksia. Pengembangan selanjutnya disarankan untuk memperluas dataset dengan berbagai bahasa dan membandingkan metode klasifikasi lain guna meningkatkan akurasi dan adaptabilitas sistem.

DAFTAR PUSTAKA

- [1] R. L. Ginting, A. Y. K. Siburian, T. E. Sianturi, S. M. Sianturi, N. B. Ginting, dan S. A. Pratiwi, "Bimbingan Konseling Bagi Anak Cerdas Istimewa Dan Kesulitan Belajar (Disleksia, Disgrafia, Diskalkulia)," *J. Pendidik. Berkarakter*, vol. 1, no. 6, hal. 134–145, 2023.
- [2] M. U. Hasballah dan N. B. M. Rabi, "Meningkatkan Kemampuan Literasi Dini Pada Anak Disleksia Melalui Metode 'Vakta,'" *J. Buah Hati*, vol. 9, no. 1, hal. 43–56, 2022, doi: 10.46244/buahhati.v9i1.1735.
- [3] D. Yuliana Putri, A. Siti Lathifah, C. Mukholis Aji Prasetyo, dan S. Suparmi, "Peran Guru dalam Meningkatkan Keterampilan Membaca Anak Disleksia," *Wahana Karya Ilm. Pendidik.*, vol. 8, no. 01, hal. 26–36, 2024, doi: 10.35706/wkip.v8i01.11578.
- [4] E. Rahmawati dan M. Muhroji, "Pengaruh Media Puzzle Huruf untuk Meningkatkan Kemampuan Membaca Anak Disleksia," *Ideguru J. Karya Ilm. Guru*, vol. 9, no. 3, hal. 1408–1413, 2024, doi: 10.51169/ideguru.v9i3.1103.
- [5] L. Budiani dan I. B. Putrayasa, "Kesulitan Membaca Kata Anak Disleksia Usia 7-12 Tahun di Sekolah Dasar," *J. Educ. Action Res.*, vol. 7, no. 3, hal. 376–381, 2023, doi: 10.23887/jear.v7i3.66560.
- [6] A. Harbani, S. Syafril, J. Warmansyah, dan A. F. Arif, "Penerapan Metode Naive Bayes untuk Penentuan Penanganan Pasien pada Penyakit ISPA," *Digit. Transform. Technol.*, vol. 4, no. 2, hal. 1012–1023, 2025, doi: 10.47709/digitech.v4i2.5190.
- [7] M. Afriansyah, Joni Saputra, V. Y. P. Ardhana, dan Yuan Sa'adati, "Algoritma Naive Bayes Yang Efisien Untuk Klasifikasi Buah Pisang Raja Berdasarkan Fitur Warna," *J. Inf. Syst. Manag. Digit. Bus.*, vol. 1, no. 2, hal. 236–248, 2024, doi: 10.59407/jismdb.v1i2.438.
- [8] N. A. Prakoso Indaryono, "Analisa Perbandingan Algoritma Random Forest Dan Naïve Bayes Untuk Klasifikasi Curah Hujan Berdasarkan Iklim Di Indonesia," *JIPi (Jurnal Ilm. Penelit. dan Pembelajaran Inform.)*, vol. 9, no. 1, hal. 158–167, 2024, doi: 10.29100/jipi.v9i1.4421.
- [9] N. Darmayanti, N. Hayati, A. Rohali, dan Z. E. Marpaung, "Pendidikan dan Bimbingan Anak Berkebutuhan Khusus (ABK) Disleksia," *El-Mujtama J. Pengabd. Masy.*, vol. 4, no. 2, hal. 854–862, 2023, doi: 10.47467/elmujtama.v4i2.4431.
- [10] L. Oktamarina dkk., "BHARASUMBA: Jurnal Multidisipliner GANGGUAN GEJALA DISLEKSIA PADA ANAK USIA DINI," *J. Multidisipliner*, vol. 02, no. 01, hal. 104–118, 2022.

- [11] T. Sinaga, "Penerapan Algoritma Naive Bayes dalam Pemrosesan Bahasa Alamiah," no. December, 2023, [Daring]. Tersedia pada: <https://www.researchgate.net/publication/376720851>
- [12] Siti Alyunita Mega Lestari, Akim M.H. Pardede, dan Magdalena Simanjuntak, "Prediksi Disleksia pada Anak menggunakan Metode Naive Bayes," *J. Kaji. dan Penelit. Umum*, vol. 2, no. 5, hal. 37–51, 2024, doi: 10.47861/jkpu-nalanda.v2i5.1287.
- [13] E. P. Adamansyah dan A. Yudhistira, "Evaluasi Opini Publik di Media Sosial X terhadap Kebijakan Pajak Pertambahan Nilai 12% di Indonesia Menggunakan Naive Bayes dan Decision Tree," *J. Pendidik. dan Teknol. Indones.*, vol. 5, no. 3, hal. 831–843, 2025, doi: 10.52436/1.jpti.710.
- [14] M. Yusuf Rismanda Gaja, I. Maulana, dan O. Komarudin, "Analisis Sentimen Opini Pengguna Aplikasi Vidio Pada Ulasan Playstore Menggunakan Algoritma Naive Bayes," *JATI (Jurnal Mhs. Tek. Inform.)*, vol. 7, no. 4, hal. 2767–2774, 2024, doi: 10.36040/jati.v7i4.7197.
- [15] F. A. Ramadhan, S. H. Sitorus, dan T. Rismawan, "Penerapan Metode Multinomial Naive Bayes untuk Klasifikasi Judul Berita Clickbait dengan Term Frequency - Inverse Document Frequency," *J. Sist. dan Teknol. Inf.*, vol. 11, no. 1, hal. 70, 2023, doi: 10.26418/justin.v11i1.57452.
- [16] Yuyun, N. Hidayah, dan S. Sahibu, "Algoritma Multinomial Naive Bayes Untuk Klasifikasi Sentimen Pemerintah Terhadap Penanganan Covid-19 Menggunakan Data Twitter," *J. RESTI*, vol. 5, no. 4, hal. 820–826, 2021, doi: 10.29207/resti.v5i4.3146.
- [17] N. F. Az-haari, D. Juardi, dan A. Jamaludin, "Analisis Sentimen Terhadap Boikot Brand Pro-Israel Pada Twitter Menggunakan Metode Naive Bayes," *JATI (Jurnal Mhs. Tek. Inform.)*, vol. 8, no. 3, hal. 4256–4261, 2024, doi: 10.36040/jati.v8i3.9888.
- [18] Gilbert, Syariful Alam, dan M. Imam Sulisty, "Analisis Sentimen Berdasarkan Ulasan Pengguna Aplikasi Mypertamina Pada Google Playstore Menggunakan Metode Naive Bayes," *STORAGE J. Ilm. Tek. dan Ilmu Komput.*, vol. 2, no. 3, hal. 100–108, 2023, doi: 10.55123/storage.v2i3.2333.
- [19] S. Surayya dan H. Mubarak, "Pengaruh Aplikasi Marbel Membaca Terhadap Kemampuan Membaca Anak Disleksia," *Ibriez J. Kependidikan Dasar Islam Berbas. Sains*, vol. 6, 2021, doi: 10.21154/ibriez.v6i2.165.
- [20] N. Haifa, A. Mulyadiprana, dan R. Respati, "Pengenalan Ciri Anak Pengidap Disleksia," *PEDADIDAKTIKA J. Ilm. Pendidik. Guru Sekol. Dasar*, vol. 7, no. 2, hal. 21–32, 2020, doi: 10.17509/pedadidaktika.v7i2.25035.
- [21] E. Martantoh dan N. Yanih, "Implementasi Metode Naive Bayes Untuk Klasifikasi Karakteristik Kepribadian Siswa Di Sekolah MTS Darussa'adah Menggunakan Php Mysql," *J. Teknol. Sist. Inf.*, vol. 3, no. 2, hal. 166–175, 2022, doi: 10.35957/jtsi.v3i2.2896.
- [22] S. Susilawati dan M. Iqbal, "Penerapan Metode Naive Bayes Untuk Mengidentifikasi Sentimen Pengguna Pada Ulasan Aplikasi ReelShort di Google Play Store," *Simkom*, vol. 10, no. 1, hal. 49–59, 2025, doi: 10.51717/simkom.v10i1.686.
- [23] A. Oktaviyani, A. Heryati, dan M. F. A. Alie, "Penerapan Metode Naive Bayes Untuk Klasifikasi Kategori Olah Pangan (Studi Kasus Dinas Kesehatan Kota Palembang)," *AnoaTIK J. Teknol. Inf. dan Komput.*, vol. 2, no. 1, hal. 30–38, 2024, doi:

- 10.33772/anoatik.v2i1.30.
- [24] M. K. Insan, U. Hayati, dan O. Nurdiawan, "Analisis Sentimen Aplikasi Brimo Pada Ulasan Pengguna Di," *J. Mhs. Tek. Inform.*, vol. 7, no. 1, hal. 478–483, 2023.
- [25] F. Rumaisa, Y. Puspitarani, A. Rosita, A. Zakiah, dan S. Violina, "Penerapan Natural Language Processing (NLP) di bidang pendidikan," *J. Inov. Masy.*, vol. 1, no. 3, hal. 232–235, 2021, doi: 10.33197/jim.vol1.iss3.2021.799.