

**ANALISIS SENTIMEN DENGAN KOMPARASI *RANDOM
FOREST, SVM DAN NAIVE BAYES* MENGGUNAKAN
DATASET 20 APLIKASI EDUKASI ANAK**



Disusun Oleh:

N a m a : Fadhil Albar

NIM : 18523183

**PROGRAM STUDI INFORMATIKA – PROGRAM SARJANA
FAKULTAS TEKNOLOGI INDUSTRI
UNIVERSITAS ISLAM INDONESIA
2025**

HALAMAN PENGESAHAN DOSEN PEMBIMBING

**ANALISIS SENTIMEN DENGAN KOMPARASI *RANDOM
FOREST, SVM DAN NAIVE BAYES* MENGGUNAKAN
DATASET 20 APLIKASI EDUKASI ANAK**

TUGAS AKHIR



Disusun Oleh:

N a m a : Fadhil Albar
NIM : 18523183

Yogyakarta, 29 Juli 2025

Pembimbing,

(Ir. Chandra Kusuma Dewa, S.Kom., M.Cs., Ph.D.)

HALAMAN PENGESAHAN DOSEN PENGUJI

ANALISIS SENTIMEN DENGAN KOMPARASI *RANDOM FOREST*, *SVM* DAN *NAIVE BAYES* MENGGUNAKAN DATASET 20 APLIKASI EDUKASI ANAK

TUGAS AKHIR

Telah dipertahankan di depan sidang penguji sebagai salah satu syarat untuk memperoleh gelar Sarjana Komputer dari Program Studi Informatika – Program Sarjana di Fakultas Teknologi Industri Universitas Islam Indonesia

Yogyakarta, 29 Juli 2025

Tim Penguji

Ir. Chandra Kusuma Dewa, S.Kom., M.Cs.,
Ph.D.

Anggota 1

Rahadian Kurniawan, S.Kom., M.Kom.

Anggota 2

Dr. Yudi Prayudi, S.Si., M.Kom..

Mengetahui,

Ketua Program Studi Informatika – Program Sarjana

Fakultas Teknologi Industri

Universitas Islam Indonesia



(Dhomas Hatta Fudholi, S.T., M.Eng., Ph.D.)

HALAMAN PERNYATAAN KEASLIAN TUGAS AKHIR

Yang bertanda tangan di bawah ini:

Nama : Fadhil Albar

NIM : 18523183

Tugas akhir dengan judul:

ANALISIS SENTIMEN DENGAN KOMPARASI *RANDOM FOREST*, *SVM* DAN *NAIVE BAYES* MENGGUNAKAN DATASET 20 APLIKASI EDUKASI ANAK

Menyatakan bahwa seluruh komponen dan isi dalam tugas akhir ini adalah hasil karya saya sendiri. Apabila di kemudian hari terbukti ada beberapa bagian dari karya ini adalah bukan hasil karya sendiri, tugas akhir yang diajukan sebagai hasil karya sendiri ini siap ditarik kembali dan siap menanggung risiko dan konsekuensi apapun.

Demikian surat pernyataan ini dibuat, semoga dapat dipergunakan sebagaimana mestinya.

Yogyakarta, 29 Juli 2025



(Fadhil Albar)

HALAMAN PERSEMBAHAN

Dengan penuh rasa syukur dan kerendahan hati, saya mempersembahkan karya tulis ilmiah ini sebagai puncak dari sebuah perjalanan panjang yang tidak mungkin dapat saya lalui seorang diri. Skripsi ini adalah wujud terima kasih tulus yang saya haturkan kepada orang-orang terkasih yang telah menjadi pilar kekuatan selama proses pendidikan dan penelitian ini.

Penghargaan tertinggi saya sampaikan kepada Ayah, Ibu, dan seluruh keluarga tercinta. Untuk setiap untaian doa yang tak henti-hentinya dilangitkan, untuk kesabaran tanpa batas dalam memahami setiap waktu dan tenaga yang tercurah, serta untuk dukungan moril dan materiel yang menjadi fondasi utama bagi langkah saya, saya mengucapkan terima kasih yang tak terhingga. Tanpa restu, kepercayaan, dan semangat yang senantiasa kalian berikan, perjalanan ini tentu akan terasa jauh lebih berat. Semoga hasil kerja keras ini dapat menjadi seberkas cahaya kebanggaan dan kebahagiaan untuk keluarga kita.

Rasa terima kasih yang sama besarnya juga saya sampaikan kepada para sahabat dan teman-teman seperjuangan. Kebersamaan kita, yang diwarnai dengan diskusi hingga larut malam, gelak tawa di tengah kepenatan, dan semangat untuk saling membantu saat menghadapi kesulitan, merupakan salah satu bagian terbaik dari proses ini. Perjuangan akademis ini menjadi jauh lebih ringan dan bermakna karena dilalui bersama kalian, yang selalu siap sedia menyemangati ketika salah satu dari kita merasa lelah.

Secara khusus, saya ingin mengucapkan terima kasih yang sebesar-besarnya kepada Bapak Chandra. Bimbingan, arahan yang konstruktif, serta waktu berharga yang telah Bapak luangkan sangatlah berarti. Wawasan dan ilmu yang Bapak berikan tidak hanya menjadi panduan krusial dalam penyelesaian skripsi ini, tetapi juga telah memperluas cakrawala berpikir saya.

Pada akhirnya, saya menyadari bahwa skripsi ini bukanlah buah dari kerja keras saya semata, melainkan sebuah cerminan dari jalinan doa, harapan, dan dukungan dari semua pihak. Semoga karya sederhana ini dapat memberikan manfaat, sekecil apa pun, dan menjadi langkah awal yang baik untuk perjalanan saya selanjutnya di masa depan.

HALAMAN MOTO

“Everything that lives is designed to end. They are perpetually trapped in a never-ending spiral of life and death. However, life is all about the struggle within this cycle. That is what 'we' believe.”

~ YoRHa No.2 Type B

KATA PENGANTAR

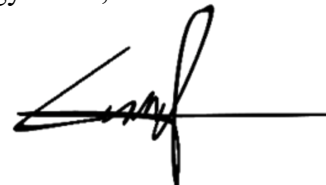
Puji syukur penulis panjatkan kepada Tuhan Yang Maha Esa atas rahmat dan karunia-Nya, sehingga penulis dapat menyelesaikan penelitian skripsi ini dengan baik dan tepat waktu. Penelitian yang berjudul "ANALISIS SENTIMEN DENGAN KOMPARASI RANDOM FOREST, SVM DAN NAIVE BAYES MENGGUNAKAN DATASET 20 APLIKASI EDUKASI ANAK" ini disusun sebagai salah satu syarat untuk menyelesaikan program studi.

Penulis menyadari bahwa penelitian ini tidak akan terwujud tanpa adanya bantuan, bimbingan, dan dukungan dari berbagai pihak. Oleh karena itu, dengan tulus penulis ingin mengucapkan terima kasih kepada:

1. Keluarga tercinta, yang selalu memberikan doa, semangat, dan dukungan tanpa henti.
2. Bapak Chandra selaku dosen pembimbing yang telah meluangkan waktu, tenaga, dan pikiran untuk memberikan arahan dan masukan yang sangat berharga.
3. Teman-teman Serikat Spicy Bistro, Nix's Perch, Merchant House serta Penghuni jalanan kota Sofya yang selalu hadir untuk obrolan malam hari dan sesekali membahas tentang kehidupan khususnya kuliah.
4. Teman-teman lama semasa SD, SMP, SMK serta sahabat online yang masih berhubungan dan berkomunikasi dengan baik.

Penulis menyadari bahwa penelitian ini masih memiliki banyak kekurangan. Oleh karena itu, kritik dan saran yang membangun sangat diharapkan demi perbaikan di masa mendatang. Semoga penelitian ini dapat memberikan manfaat bagi para pembaca dan perkembangan ilmu pengetahuan.

Yogyakarta, 29 Juli 2025



(Fadhil Albar)

SARI

Ulasan pengguna di platform digital seperti Google Play Store telah menjadi sumber data vital bagi pengembang aplikasi, namun volumenya yang masif menyulitkan analisis manual. Penelitian ini bertujuan untuk melakukan analisis sentimen terhadap ulasan pengguna pada 20 aplikasi pendidikan anak berbahasa Indonesia dan membandingkan performa tiga model *machine learning* *Random Forest*, *Support Vector Machine* (SVM), dan *Naïve Bayes*—untuk tugas klasifikasi sentimen. Data sebanyak 9.990 ulasan dikumpulkan dari Google Play Store menggunakan metode *scrapping*, kemudian diberi label secara otomatis menggunakan model RoBERTa. Setelah melalui tahap *pre-processing* dan penanganan data tidak seimbang dengan metode SMOTE, ketiga model diuji menggunakan sembilan skenario rasio pembagian data latih dan data uji (dari 90:10 hingga 10:90). Hasil penelitian menunjukkan bahwa *Random Forest* secara konsisten memberikan kinerja terbaik, dengan akurasi puncak mencapai 92,68% pada rasio 90:10, diikuti oleh SVM dengan kinerja yang sangat kompetitif. Model *Naïve Bayes* menunjukkan performa terendah di semua skenario. Ditemukan pula bahwa rasio data latih yang lebih besar (80% ke atas) secara signifikan meningkatkan akurasi model. Secara keseluruhan, sentimen pengguna terhadap aplikasi pendidikan anak cenderung positif, dan model *Random Forest* direkomendasikan sebagai metode paling andal untuk studi kasus ini.

Kata kunci: Analisis Sentimen, *Random Forest*, *Support Vector Machine* (SVM), *Naïve Bayes*, *Machine Learning*, Ulasan, Aplikasi Pendidikan Anak

GLOSARIUM

Analisis Sentimen	Proses komputasi untuk mengidentifikasi, mengekstrak, dan mengklasifikasikan opini atau polaritas emosional (positif, negatif, netral) dari data teks. Sering juga disebut Opinion Mining.
Data Testing	Bagian dari dataset yang digunakan untuk menguji dan mengevaluasi kinerja model setelah proses pelatihan selesai. Data ini tidak pernah dilihat oleh model selama pelatihan.
Data Training	Bagian dari dataset yang digunakan untuk "melatih" model machine learning agar dapat mempelajari pola-pola yang ada di dalam data
Labeling	Proses pemberian label atau kategori (misalnya, 'positif', 'negatif') pada data mentah. Dalam penelitian ini, labeling dilakukan secara otomatis menggunakan model <i>RoBERTa</i> .
Machine Learning	Proses memecah teks menjadi unit-unit yang lebih kecil yang disebut "token", yang bisa berupa kata, frasa, atau simbol.=.
Wordcloud	Visualisasi data teks di mana ukuran setiap kata sebanding dengan frekuensi kemunculannya dalam teks tersebut. Kata yang paling sering muncul akan ditampilkan paling besar.

DAFTAR ISI

HALAMAN JUDUL	i
HALAMAN PENGESAHAN DOSEN PEMBIMBING.....	ii
HALAMAN PENGESAHAN DOSEN PENGUJI	iii
HALAMAN PERNYATAAN KEASLIAN TUGAS AKHIR	iv
HALAMAN PERSEMBAHAN	v
HALAMAN MOTO	vi
KATA PENGANTAR	vii
SARI.....	viii
GLOSARIUM.....	ix
DAFTAR ISI	x
DAFTAR TABEL	xii
DAFTAR GAMBAR.....	xiii
BAB I PENDAHULUAN	1
1.1 Latar Belakang	1
1.2 Rumusan Masalah	3
1.3 Tujuan Penelitian	3
1.4 Manfaat Penelitian	3
1.5 Batasan Penelitian	3
1.6 Sistematika Penulisan	4
BAB II LANDASAN TEORI	5
2.1 Penelitian Sejenis	5
2.2 Penelitian Dengan Model Serupa.....	7
2.3 Aplikasi Pendidikan Anak	9
2.4 Analisis Sentimen	9
2.5 Playstore.....	10
2.6 Text Mining.....	10
2.7 Random Forest	11
2.8 Support Vector Machine (SVM).....	11
2.9 Naïve Bayes	11
2.10 RoBERTa	12
BAB III METODE PENELITIAN	14
3.1 Alur Penelitian	14
3.2 Pengumpulan Data	14
3.3 Variabel.....	16
3.4 Labeling Menggunakan RoBERTa.....	16
3.5 Pre-processing Data	18
3.6 Penerapan TF-IDF.....	24
3.7 Handling Data Imbalance.....	24
BAB IV HASIL DAN PEMBAHASAN	26
4.1 Pengetesan Performa.....	26
4.2 Testing performa dengan berbagai rasio data uji dan latih	27
4.3 Hasil Tes Performa :	30
BAB V KESIMPULAN DAN SARAN.....	50
5.1 Kesimpulan	50
5.2 Visualisasi	53
5.3 Saran.....	56
DAFTAR PUSTAKA	57

LAMPIRAN	60
----------------	----

DAFTAR TABEL

Tabel 2.1 Tabel Penelitian sejenis	7
Tabel 2.2 Tabel Penelitian Dengan Model Serupa	9
Tabel 3.1 Perbandingan sebelum dan sesudah folding case dan symbol removal.....	18
Tabel 3.2 Perbandingan sebelum dan sesudah Tokenizing	19
Tabel 3.3 Perbandingan Proses Stopwords Removal	21
Tabel 3.4 Perbandingan sebelum dan sesudah Stemming	23
Tabel 4.1 Tabel Ringkasan Hasil Pengetesan Performa	44

DAFTAR GAMBAR

Gambar 3.1 Flowchart Alur Penelitian	14
Gambar 3.2 Capture Scrapping Data Dari Playstore	15
Gambar 3.3 Sintaks Scrapping Data	16
Gambar 3.4 Sintaks Labeling Menggunakan RoBERTa	17
Gambar 3.5 Sintaks folding case dan symbol removal.....	18
Gambar 3.6 Sintaks Tokenizing.....	19
Gambar 3.7 Sintaks Stopwords Removal	21
Gambar 3.8 Sintaks Stemming	23
Gambar 3.9 Sintaks Penerapan TF-IDF	24
Gambar 3.10 Sintaks SMOTE Untuk <i>Handling Data Imbalance</i>	25
Gambar 4.1 Sintaks Splitting Data Rasio	27
Gambar 4.2 Sintaks Testing Model Random Forest.....	27
Gambar 4.3 Sintaks Testing Model SVM.....	28
Gambar 4.4 Sintaks Testing Model Naive Bayes	29
Gambar 4.5 Output Rasio 90:10	31
Gambar 4.6 Output Rasio 80:20	32
Gambar 4.7 Output Rasio 70:30	34
Gambar 4.8 Output Rasio 60:40	35
Gambar 4.9 Output Rasio 50:50	37
Gambar 4.10 Output Rasio 40:60	38
Gambar 4.11 Output Rasio 30:70	40
Gambar 4.12 Output rasio 20:80.....	41
Gambar 4.13 Output rasio 10:90.....	43
Gambar 4.14 Confusion Matrix Random Forest 90:10	45
Gambar 4.15 Confusion Matrix SVM 90:10	45
Gambar 4.16 Confusion Matrix Naive Bayes 90:10.....	46
Gambar 4.17 Gambar Grafik Performa Akurasi Model	47
Gambar 4.17 Gambar Grafik Performa Akurasi Model	47
Gambar 4.18 Gambar Grafik Performa Presisi Model	48

Gambar 4.19 Gambar Grafik Performa <i>Recall</i> Model.....	49
Gambar 4.20 Gambar Grafik Performa Skor F1 Model	49
Gambar 5.1 Gambar <i>Wordcloud</i> Sentimen Positif	53
Gambar 5.2 Gambar <i>Wordcloud</i> Sentimen Negatif.....	54
Gambar 5.3 Gambar <i>Wordcloud</i> Sentimen Netral.....	55

BAB I

PENDAHULUAN

1.1 Latar Belakang

Lanskap interaksi antara penyedia layanan dan pelanggan telah berubah secara drastis di tengah laju transformasi digital yang cepat. Platform digital, terutama ekosistem aplikasi seluler, telah menjadi tempat utama di mana barang dan jasa dikomersialisasikan, digunakan, dan dinilai, terutama dalam bentuk ulasan dan peringkat yang tersedia secara online. Komentar ini, yang datang secara masif dan asli, telah menjadi sumber daya penting untuk keberlangsungan dan inovasi perusahaan dengan memanfaatkan metode analisis otomatis untuk membantu efisiensi perusahaan (Lowell dkk., 2025).

Secara spesifik, Google Play Store, dengan jutaan aplikasi dan miliaran pengguna, menjadi salah satu platform distribusi aplikasi terbesar di dunia. Platform ini mengumpulkan banyak ulasan setiap hari, menjadikannya sebagai sumber informasi yang luar biasa untuk pengalaman, kepuasan, dan keluhan pengguna. Ulasan ini tidak hanya berdampak pada popularitas aplikasi dan jumlah unduhannya, tetapi juga memberikan pengetahuan strategis yang dapat digunakan pengembang untuk mengetahui umpan balik dari pengguna aplikasi yang berguna untuk memperbaiki produk, menemukan bug, dan memahami posisi pesaing mereka di pasar dikarenakan dari hasil penelitian data hasil analisa tersebut berdampak besar terhadap persepsi dan keputusan masyarakat dalam memilih dan menggunakan aplikasi (M. P. Ningrum & Mutia, 2025).

Aplikasi pendidikan anak yang tersedia di Google Play Store untuk anak-anak di Indonesia memiliki potensi yang sangat besar. Ini didorong oleh jumlah orang tua yang lebih banyak menggunakan internet dan jumlah siswa yang lebih muda yang mencari metode pembelajaran tambahan. Pasar ini memiliki banyak pemain, mulai dari platform edutech besar yang materinya sesuai dengan kurikulum nasional untuk anak usia sekolah, hingga aplikasi yang secara khusus ditujukan untuk pendidikan anak usia dini (PAUD). Aplikasi PAUD biasanya berfokus pada pelajaran dasar seperti membaca, menulis, berhitung (calistung), pengenalan warna dan bentuk, serta materi agama dasar seperti mengaji dan huruf hijaiyah. Dalam pasar yang kompetitif ini, aplikasi sangat bergantung pada dua komponen utama: gamifikasi yang kuat untuk menarik perhatian dan keterlibatan anak serta lokalisasi konten yang mendalam, hasil penelitian membuktikan bahwa gamifikasi edukasi berdampak pada

ketertarikan anak dalam mempelajari materi yang sebelumnya dirasakan kurang menarik (D. E. Putri & Mandala, 2021).

Game edukasi, yang dimaksudkan untuk menjadi media pembelajaran alternatif yang lebih menarik daripada pendekatan konvensional. Kemampuannya untuk memvisualisasikan masalah nyata dan menyajikannya secara interaktif membuat aplikasi ini sangat penting. Ini mendorong anak untuk belajar secara aktif belajar dengan melakukan untuk menyelesaikan tantangan dalam permainan. Melalui tantangan, ketepatan, dan animasi, game edukasi dapat meningkatkan kreativitas, daya nalar, dan daya ingat anak. Akibatnya, aplikasi ini tidak hanya membantu guru mengubah metode belajar konvensional menjadi lebih simulatif, tetapi juga menawarkan cara yang menyenangkan bagi siswa untuk belajar konsep seperti simbol, berhitung, dan menyusun kata (Vitianingsih, A. V. 2016)

Tapi ada masalah besar dengan volume dan kecepatan data tekstual yang luar biasa ini. Menganalisis ribuan, jika tidak jutaan, ulasan secara manual adalah tidak praktis, memakan waktu, dan rentan terhadap subjektivitas manusia. Akibatnya, perlu ada sistem otomatis yang dapat mengolah data mentah menjadi wawasan yang dapat ditindaklanjuti dan terstruktur. Tanpa teknologi, banyak informasi penting dari ulasan pengguna akan terabaikan, dalam hal ini identifikasi opini atau pandangan pengguna aplikasi berupa sentimen positif, negatif maupun netral (Cahyono & Praneswara, 2023).

Bidang keilmuan Natural Language Processing (NLP) menawarkan berbagai teknik yang kuat untuk mengatasi masalah tersebut. Analisis sentimen adalah salah satu sub bidang yang paling relevan dalam hal ini. Analisis sentimen adalah proses penggunaan proses pemrosesan bahasa untuk secara sistematis mengidentifikasi, mengekstrak, dan mengklasifikasikan opini atau polaritas emosional positif, negatif, atau netral. Metode NLP dapat dijadikan salah satu opsi solusi untuk penyedia layanan aplikasi untuk bisa mengetahui umpan balik pengguna. (Rivaldi & Wismarini, 2024)

Penelitian ini berkonsentrasi pada teknik analisis sentimen untuk mengevaluasi ulasan pengguna dalam hal ini kebanyakan berasal dari ulasan orang tua pengguna yang cenderung lebih netral dan pengguna langsung anak-anak dengan ulasan yang cenderung lebih awam dan apa adanya. untuk aplikasi pendidikan anak berbahasa Indonesia yang tersedia di Google Play Store dengan pertimbangan mayoritas pengguna aplikasi tersebut merupakan warga negara Indonesia. Serta mengimplementasikan library W11wo yang diharapkan dapat melakukan proses labeling khususnya dalam konten bahasa Indonesia dengan akurat serta dengan membangun sebuah model komparasi sentimen antar model *machine learning* untuk

mengetahui performa model yang paling optimal sehingga diharapkan dapat menjadi bahan penelitian di masa depan, penelitian juga bertujuan untuk menyediakan sebuah kerangka kerja yang dapat secara efektif menggambarkan persepsi publik tentang aplikasi pendidikan anak berbahasa indonesia yang tersedia di Google Play Store.

1.2 Rumusan Masalah

Berdasarkan latar belakang tersebut, penelitian ini dapat di fokuskan pada rumusan masalah berikut :

- a. Bagaimana opini masyarakat selaku pengguna layanan aplikasi pendidikan anak berbahasa indonesia berdasarkan review pada kolom komentar play store, apakah cenderung positif, negatif, maupun netral.
- b. Bagaimana performa metode *Random Forest*, *Support Vector Machine* dan *Naïve Bayes* dalam tingkat akurasi pengesanan dengan berbagai skenario perbandingan rasio data training dan data testing.

1.3 Tujuan Penelitian

- a. Mencari informasi dan menentukan apakah beberapa aplikasi bertema pendidikan untuk anak berbahasa indonesia menggunakan sampling review dari play store tersebut bernilai positif, negatif ataupun netral.
- b. Mengetahui metode paling akurat untuk studi kasus analisis sentimen pada review aplikasi pendidikan anak berbahasa indonesia pada play store.

1.4 Manfaat Penelitian

Adapun Manfaat penelitian ini dapat menjadi bahan referensi dalam pengembangan penelitian yang lebih jauh pada bidang analisis performa model *machine learning* berupa *Random Forest*, *Support Vector Machine* dan *Naïve Bayes* untuk implementasi dalam studi kasus analisis sentimen.

1.5 Batasan Penelitian

- a. Penelitian Berfokus kepada komparasi efektivitas model *machine learning* berupa *Random Forest*, *Support Vector Machine* dan *Naïve Bayes*.
- b. Pengujian berfokus pada pengesanan performa beberapa model *machine learning* dengan skema pengesanan prediksi pada split data training dan testing dengan rasio :

- i. 90 Training : 10 Testing
- ii. 80 Training : 20 Testing
- iii. 70 Training : 30 Testing
- iv. 60 Training : 40 Testing
- v. 50 Training : 50 Testing
- vi. 40 Training : 60 Testing
- vii. 30 Training : 70 Testing
- viii. 20 Training : 80 Testing
- ix. 10 Training : 90 Testing

1.6 Sistematika Penulisan

Dalam tugas akhir ini, untuk sistematika penulisan akan disusun dengan menggunakan susunan berikut:

a. BAB I PENDAHULUAN

Bab ini terdiri dari beberapa subbab berupa latar belakang, rumusan masalah, tujuan penelitian, manfaat penelitian, batasan masalah dan sistematika penulisan.

b. BAB II LANDASAN TEORI

Bab ini menjelaskan tentang studi literatur yang dilakukan oleh penulis mengenai analisis sentimen, model *machine learning*, serta kajian terhadap penelitian sejenis.

c. BAB III METODOLOGI PENELITIAN

Bab ini berfokus pada gambaran umum penelitian, kebutuhan proses pada penelitian dan perancangan sistem pengujian.

d. BAB IV HASIL DAN PEMBAHASAN

Bab ini membahas tentang hasil penerapan proses penelitian dan komparasi performa model *machine learning* berupa *Random Forest*, *Support Vector Machine* dan *Naïve Bayes*.

e. BAB V KESIMPULAN DAN SARAN

Bab ini akan menjelaskan tentang hasil penelitian yang telah dilaksanakan berupa pengujian performa dan komparasi antar model *machine learning* serta saran untuk meningkatkan metode pengujian yang lebih baik untuk penelitian di masa depan

BAB II

LANDASAN TEORI

2.1 Penelitian Sejenis

Dalam kurun waktu beberapa tahun terakhir, penelitian tentang analisis sentimen: dalam literatur yang berfokus pada ulasan aplikasi di Google Play Store. Berbagai studi telah menggunakan dan mengevaluasi metode pembelajaran mesin untuk klasifikasi opini yang akurat dan otomatis. Penelitian-penelitian ini biasanya membahas bagaimana menggunakan algoritma khusus pada dataset ulasan tertentu, melakukan perbandingan kinerja antara berbagai metode, dan cara pendekatan pre-processing data. Beberapa penelitian sebelumnya akan dijabarkan pada bagian berikut:

1. Analisis Sentimen Aplikasi DANA Menggunakan Naïve Bayes dan K-Nearest Neighbor.

Penelitian yang dilakukan oleh (Ramadhan & Sugianto, 2024) ini adalah untuk menganalisa dan membandingkan kinerja algoritma *Naïve Bayes* (NB) dan *K-Nearest Neighbor* (KNN) untuk klasifikasi sentimen pada ulasan aplikasi dompet digital DANA. Data yang digunakan berasal dari 2.000 ulasan pengguna yang dikumpulkan dari Google Play Store dari tahun 2021 hingga Agustus 2023, dan 1.995 data melewati tahap pra-pemrosesan. Hasil penelitian menunjukkan bahwa algoritma K-Nearest Neighbor memiliki akurasi sebesar 87,07% dibandingkan dengan algoritma Naive Bayes, yang memiliki 80,95%.

2. Perbandingan Kinerja Naïve Bayes dan SVM untuk Klasifikasi Sentimen Ulasan Game Genshin Impact.

Studi ini melihat bagaimana algoritma *Naive Bayes* (NB) dan *Support Vector Machine* (SVM) bekerja dalam mengklasifikasikan sentimen dalam ulasan game online Genshin Impact. Data yang digunakan berasal dari 2.000 ulasan berbahasa Inggris yang diambil dari Google Play Store. Hasil penelitian ini menunjukkan bahwa SVM lebih unggul dalam klasifikasi sentimen pada ulasan game online dengan akurasi 83%, sedangkan Naive Bayes hanya mencapai 75,5%. Hasil ini mendorong penelitian ini untuk menggunakan SVM untuk klasifikasi sentimen pada ulasan game online (Safrudin dkk., 2024).

3. Analisis Sentimen Ulasan Aplikasi ChatGPT Menggunakan *Support Vector Machine*.

Metode *Support Vector Machine* (SVM) digunakan dalam penelitian ini oleh (Salsabila & Muntahanah, 2024) untuk menganalisis ulasan pengguna aplikasi ChatGPT di Google Play Store. Data yang dianalisis terdiri dari 8.369 ulasan yang dikumpulkan antara tanggal 27 Juli 2023 dan tanggal 30 Juni 2024. Meskipun model SVM sangat baik untuk menghasilkan sentimen positif, penelitian ini menemukan bahwa mereka kurang akurat dalam mengklasifikasikan ulasan negatif.

4. Analisis Sentimen Aplikasi Threads Menggunakan *Support Vector Machine*.

Tujuan peneliti (Nufairi dkk., 2024) ini adalah untuk menentukan persepsi pengguna terhadap aplikasi media sosial Threads dan juga untuk mengukur akurasi algoritma *Support Vector Machine* (SVM) dalam analisis. Data yang digunakan dari Google Play Store berjumlah 1.429 ulasan, dengan 1.103 ulasan positif dan 326 ulasan negatif. Algoritma SVM berhasil mengklasifikasikan sentimen dengan tingkat akurasi 88% dengan menggunakan rasio pembagian data 80:20.

5. Analisis Sentimen Provider by.U Menggunakan *Naïve Bayes Classifier* dan *K-Nearest Neighbor*.

Dengan menggunakan perbandingan metode *Naïve Bayes Classifier* (NBC) dan *K-Nearest Neighbor* (KNN), penelitian ini bertujuan untuk mengklasifikasikan sentimen dari 5.608 sampel ulasan pengguna layanan provider by.U. Hasil penelitian menunjukkan bahwa *Naïve Bayes Classifier* jauh lebih baik. NBC memiliki akurasi tertinggi sebesar 95 persen pada pembagian data 90:10, sedangkan KNN hanya memiliki 89 persen (Putri, R.C. 2024).

Berikut perbandingan hasil penelitian dari kelima penelitian di atas, dapat dilihat pada Tabel dibawah ini:

Tabel 2.1 Tabel Penelitian sejenis

Pembanding	Penelitian 1	Penelitian 2	Penelitian 3	Penelitian 4	Penelitian 5
Objek	Dana	Genshin Impact	ChatGPT	Threads	By.U
Algoritma	<i>Naïve Bayes & K-Nearest Neighbor</i>	<i>Naïve Bayes & Support Vector Machine</i>	<i>Support Vector Machine</i>	<i>Support Vector Machine</i>	<i>Naïve Bayes & K-Nearest Neighbor</i>
Sumber & Jumlah Data	Google Play Store (1.995 ulasan)	Google Play Store (2.000 ulasan)	Google Play Store (8.369 ulasan)	Google Play Store (1.429 ulasan)	Google Play Store (5.608 ulasan)
Hasil Tes Akurasi	KNN 87,07% Naïve bayes 80,95%	SVM 83% Naive Bayes 75,5%	SVM 91%	SVM 88%	KNN 89% Naïve Bayes 95%

2.2 Penelitian Dengan Model Serupa

Terdapat sejumlah penelitian yang melakukan analisis komparatif untuk mengevaluasi efektivitas berbagai algoritma machine learning dalam tugas klasifikasi. Secara khusus, tiga penelitian berikut menyajikan perbandingan kinerja antara algoritma Naïve Bayes, *Support Vector Machine* (SVM), dan Random Forest.

1. Komparasi Algoritma Random Forest, Naïve Bayes, dan SVM pada Sentimen Kebijakan PPN 12%.

Penelitian yang dilakukan oleh (Purnomo dkk., 2025) mengevaluasi dan membandingkan kinerja tiga algoritma pembelajaran mesin: Random Forest, Naïve Bayes, dan Support Vector Machine (SVM). Tujuan dari algoritma ini adalah untuk mengategorikan persepsi publik di Twitter tentang rencana kenaikan Pajak Pertambahan Nilai (PPN) di Indonesia menjadi 12% pada tahun 2025. Penelitian ini bertujuan untuk menentukan algoritma mana yang paling akurat untuk menentukan opini masyarakat positif, netral, dan negatif dengan menggunakan 2747 tweet yang dikumpulkan dari April 2024 hingga Januari 2025. Hasil menunjukkan bahwa SVM adalah yang terbaik dengan akurasi 76,25%, diikuti oleh Random Forest (72,35%) dan Naïve Bayes (60,66%). Ini menunjukkan bahwa SVM lebih efektif dalam menangani data opini publik yang rumit di media sosial.

2. Komparasi Algoritma Naive Bayes, Random Forest dan SVM untuk Memprediksi Niat Pembelanja Online

Penelitian yang dilakukan oleh (Agustyaningrum dkk., 2020) membandingkan tiga algoritma pembelajaran mesin: *Naive Bayes*, *Random Forest*, dan *Support Vector Machine* (SVM). Studi ini mengevaluasi setiap algoritma menggunakan aplikasi WEKA dengan metrik seperti F1-Score, Akurasi, Kappa Statistic, dan Mean Absolute Error (MAE). Hasilnya menunjukkan bahwa Random Forest secara konsisten unggul dalam sebagian besar kriteria, dengan nilai F1-Score tertinggi sebesar 89,9% dan akurasi tertinggi sebesar 89,9%. Oleh karena itu, penelitian menunjukkan bahwa Random Forest adalah algoritma yang paling cocok untuk mengklasifikasikan niat belanja konsumen secara online, dan itu disarankan untuk digunakan.

3. Comparison of Naïve Bayes Classifier, Support Vector Machine, Random Forest Algorithms for Public Sentiment Analysis of KIP-K Program on Twitter

Penelitian yang dilakukan oleh (Ali & Hendrastuty, 2024) menganalisis sentimen publik terhadap program Kartu Indonesia Pintar Kuliah (KIP-K) dengan membandingkan kinerja tiga algoritma: Naive Bayes, Support Vector Machine (SVM), dan Random Forest. Perbincangan hangat di Twitter tentang program tersebut—termasuk kontroversi tentang penyalahgunaan dana oleh beberapa oknum—memicu penelitian ini. Dataset yang digunakan mencakup 6.842 tweet yang dikumpulkan dari tahun 2023 hingga 2024. Data awal sangat tidak seimbang (5.382 sentimen negatif dan 257 sentimen positif), jadi teknik optimasi SMOTE digunakan untuk menyeimbangkan jumlah data di kedua kelas. Hasil pengujian menunjukkan bahwa algoritma Random Forest adalah yang terbaik untuk tugas klasifikasi ini setelah optimalisasi, dengan akurasi 100%, diikuti oleh SVM dengan 99%, dan Naive Bayes dengan 91%. Penelitian ini menyimpulkan bahwa Random Forest adalah metode paling efektif untuk tugas klasifikasi ini, dan hasilnya dapat membantu pengelola program KIP-K untuk lebih memahami persepsi publik.

Berikut perbandingan hasil penelitian dari kelima penelitian di atas, dapat dilihat pada Tabel dibawah ini:

Tabel 2.2 Tabel Penelitian Dengan Model Serupa

Pembanding	Penelitian 1	Penelitian 2	Penelitian 3
Objek	Tweet demgan topik PPN 12%	Dataset “Online Shoppers Purchasing Intention Dataset”	Tweet demgan topik program KIP-K
Algoritma	<i>Naïve Bayes</i> <i>Support Vector Machine</i> <i>Random Forest</i>	<i>Naïve Bayes</i> <i>Support Vector Machine</i> <i>Random Forest</i>	<i>Naïve Bayes</i> <i>Support Vector Machine</i> <i>Random Forest</i>
Hasil Tes Akurasi	<i>Naïve Bayes</i> 60.66% <i>Support Vector Machine</i> 76,25% <i>Random Forest</i> 72,35%	<i>Naïve Bayes</i> 80,9% <i>Support Vector Machine</i> 88,02% <i>Random Forest</i> 89.9%	<i>Naïve Bayes</i> 91% <i>Support Vector Machine</i> 99% <i>Random Forest</i> 100%

2.3 Aplikasi Pendidikan Anak

Aplikasi pendidikan anak berbahasa Indonesia di Play Store merupakan perangkat lunak yang dirancang khusus untuk perangkat seluler berbasis Android, yang ditujukan bagi anak-anak pada jenjang usia dini (PAUD) hingga awal sekolah dasar. Aplikasi sejenis ini berfungsi sebagai media pembelajaran digital yang menyajikan materi-materi edukatif dasar seperti pengenalan abjad, angka, warna, bentuk, nama hewan, hingga huruf hijaiyah dengan menggunakan Bahasa Indonesia sebagai bahasa pengantar utama baik dalam instruksi, narasi, maupun audio. Melalui pendekatan yang interaktif seperti permainan, lagu, dan visual yang menarik, aplikasi jenis bertujuan untuk menstimulasi minat belajar anak, mengembangkan kemampuan berhitung, kemampuan membaca, kemampuan menulis, kemampuan berpikir kreatif dan kemampuan pemecahan masalah (Putri, D.E. & Mandala, 2021), serta memanfaatkan ketertarikan alami anak terhadap gawai untuk kegiatan yang lebih positif dan mendidik.

2.4 Analisis Sentimen

Opinion Mining, juga dikenal sebagai analisis sentimen, adalah bidang studi yang mempelajari pendapat, sentimen, evaluasi, sikap, dan emosi individu terhadap hal-hal tertentu, seperti barang, layanan, atau topik tertentu (Putri, R. C. 2024). Analisis sentimen biasanya adalah proses yang dilakukan secara otomatis untuk mengolah dan mengekstrak informasi

emosional dari data tekstual. Ini digunakan untuk memahami persepsi seseorang dan menentukan apakah pendapat mereka positif, negatif ataupun netral. Analisis sentimen berbeda dengan deteksi topik karena analisis sentimen berkonsentrasi pada menemukan dan mengategorikan polaritas pendapat yang ditemukan dalam bahasa alami. Hasilnya dapat menjadi referensi penting dalam proses pengambilan keputusan tentang pengembangan produk maupun layanan.

2.5 Playstore

Google Play Store, yang secara resmi dikenal sebagai Google Play, adalah platform distribusi digital yang dikembangkan dan dikelola oleh Google LLC. Ini berfungsi sebagai tempat utama dan sistem distribusi untuk program perangkat lunak, permainan video, dan materi digital lainnya yang menargetkan perangkat dengan sistem operasi Android (Nurian, 2023). Selain itu, Google Play Store berfungsi sebagai antarmuka pemrograman aplikasi (API) dan layanan backend yang terintegrasi dengan sistem operasi Android. Ini memungkinkan pengguna mengakses berbagai aplikasi dan konten digital yang dibuat dengan menggunakan Android.

Google Play Store pertama kali diluncurkan pada 22 Oktober 2008 sebagai pengganti Android Market, sebuah platform yang dimaksudkan untuk menyimpan aplikasi dan permainan untuk sistem operasi Android yang baru dirilis. Untuk memperluas ekosistem Google, ia meluncurkan aplikasi berbayar pada tahun 2009, dan kemudian meluncurkan Google eBook pada tahun 2010 dan Google Music pada tahun 2011. Puncak transformasi terjadi pada 6 Maret 2012, ketika Google mengambil tindakan strategis untuk menggabungkan Android Market dan layanan konten digital lainnya ke dalam satu entitas yang lebih besar yang disebut Google Play. Pergeseran dari sekadar toko aplikasi menjadi sebuah pusat konten digital yang luas. Google Play terus meningkatkan kinerjanya melalui berbagai peningkatan penting, seperti penerapan Material Design pada tahun 2014, pemisahan penjualan perangkat keras ke Google Store pada tahun 2015, dan peluncuran sistem keamanan proaktif Google Play Protect pada tahun 2017. Semua peningkatan ini membuat Google Play menjadi pemimpin dalam distribusi konten dan keamanan bagi miliaran pengguna Android di seluruh dunia

2.6 Text Mining

Text mining, juga dikenal sebagai *text analytics*, adalah sebuah proses komputasi interdisipliner yang mengekstraksi pola, informasi, dan wawasan yang signifikan dari data teks

yang tidak terstruktur. Tujuan utama proses ini adalah untuk mengubah kumpulan teks bebas, seperti email, artikel berita, unggahan media sosial, atau ulasan pelanggan sebagai pengguna produk maupun layanan, menjadi data numerik yang terstruktur yang dapat di analisa lebih lanjut. Text mining menggunakan teknik dari berbagai bidang ilmu, terutama *Natural Language Processing* (NLP), *Machine learning*, statistik, dan linguistik komputasi, untuk mengidentifikasi topik, menganalisis sentimen, mengekstraksi entitas, dan menemukan hubungan tersembunyi di dalam data tekstual yang masif dan bisa menjadi tindakan awal dalam proses manajemen resiko, deteksi kecurangan, intelijen bisnis, dan analisa media sosial(Ferreira-Mello dkk., 2019).

2.7 Random Forest

Random Forest adalah metode supervised learning tipe ensemble yang membangun banyak pohon keputusan secara acak pada berbagai sub-sampel data selama proses pelatihan. Untuk tugas regresi, hasil rata-rata dari semua pohon digunakan untuk menentukan hasil akhir; untuk tugas klasifikasi, pilihan mayoritas dari semua pohon menentukan prediksi akhir. Metode ini menghasilkan model yang memiliki performa akurat dan stabil (Khairunnisa dkk., 2024).

2.8 Support Vector Machine (SVM)

Boser, Guyon, dan Vapnik mengembangkan Support Vector Machine (SVM). SVM pertama kali ditunjukkan di Annual Workshop on Computational Learning Theory pada tahun 1992. untuk klasifikasi dan regresi, Support Vector Machine (SVM) adalah salah satu algoritma supervised machine learning. Tujuan utama SVM adalah menemukan hyperplane terbaik yang dapat memisahkan data ke dalam berbagai kelas dengan margin maksimum. Model ini berfungsi dengan baik dalam ruang berdimensi besar dan efektif dalam situasi di mana jumlah data lebih besar daripada jumlah fitur. Selain itu, SVM dapat menggunakan trik kernel untuk menangani data yang tidak dapat dipisahkan secara linear dengan mengubahnya ke dalam ruang fitur yang lebih besar(Lowell dkk., 2025).

2.9 Naïve Bayes

Naive Bayes adalah algoritma klasifikasi yang didasarkan pada Teorema Bayes, yang menjelaskan bahwa data dapat berdiri sendiri atau tidak saling memengaruhi. Secara sederhana(Megira & Kusriani, 2024), model ini menghitung kemungkinan data masuk ke dalam kelas tertentu berdasarkan fitur-fiturnya.

Meskipun asumsi independensinya seringkali tidak sepenuhnya benar dalam kehidupan nyata, Naive Bayes sangat cepat, efektif, dan cocok untuk masalah klasifikasi teks seperti mendeteksi spam atau analisis sentimen, serta untuk masalah diagnostik medis. Sangat disukai sebagai dasar atau model awal untuk banyak proyek pembelajaran mesin karena cepatnya dalam proses pelatihan dan kemampuan untuk menangani data besar.

2.10 RoBERTa

RoBERTa, yang disebut sebagai *Robustly Optimized BERT Pre-training Approach*, adalah model bahasa yang dibuat oleh kecerdasan buatan Facebook sebagai penyempurnaan dari *BERT* (Pookduang dkk., 2025). Meskipun memiliki struktur yang sama, *RoBERTa* dioptimalkan dengan mengubah strategi pre-training untuk meningkatkan kinerjanya secara signifikan. Salah satu perubahan utama yang dilakukan adalah sebagai berikut:

1. melatih model dengan data yang lebih besar dan ukuran batch yang lebih besar.
2. menghilangkan tugas prediksi kalimat berikutnya (Next Sentence Prediction/NSP) yang dianggap kurang efektif.
3. menggunakan penutup dinamis, yang berarti pola penutup kata diubah setiap kali pelatihan, bukan BERT yang statis.

Hasil dari optimisasi ini adalah sebuah model yang lebih kuat dan berhasil yang mencapai kinerja cangguh (standar) pada berbagai tugas *Natural Language Processing* (NLP), seperti *GLUE*, *SQuAD*, dan *RACE*. Secara khusus, RoBERTa menunjukkan bahwa meningkatkan metode pelatihan *BERT* tidak memerlukan mengubah arsitektur dasarnya.

Berdasarkan penelitian yang disajikan (Salsabila & Muntahanah, 2024), model *w11wo/indonesian-roberta-base-sentiment-classifier* terbukti unggul secara komparatif dan kinerjanya tervalidasi secara menyeluruh, sehingga dapat menjadi referensi yang relevan untuk pelabelan sentimen. Dalam penelitian ini, model IndoRoBERTa ini menunjukkan kinerja terbaik dibandingkan varian IndoBERT lainnya. Dengan akurasi tes sebesar 96,2% dan skor F1-sebesar 95,2%, keandalannya diuji melalui evaluasi menyeluruh dengan berbagai metrik. Laporan klasifikasi menunjukkan kinerja yang sangat baik untuk semua kelas sentimen, sementara analisis confusion matrix mengkonfirmasi prediksi yang sangat akurat untuk setiap kelas—dengan tingkat akurasi 95 persen untuk kelas negatif, 95 persen untuk kelas netral, dan 97 persen untuk kelas positif. Selain itu, nilai sensitivitas (0.9569), spesifisitas (0.9784), MCC (0.9340), dan Indeks Kappa Cohen (0.9339) yang sangat tinggi menunjukkan bahwa model ini menunjukkan keseimbangan yang sangat baik dalam pengenalan dan pengenalan setiap kelas.

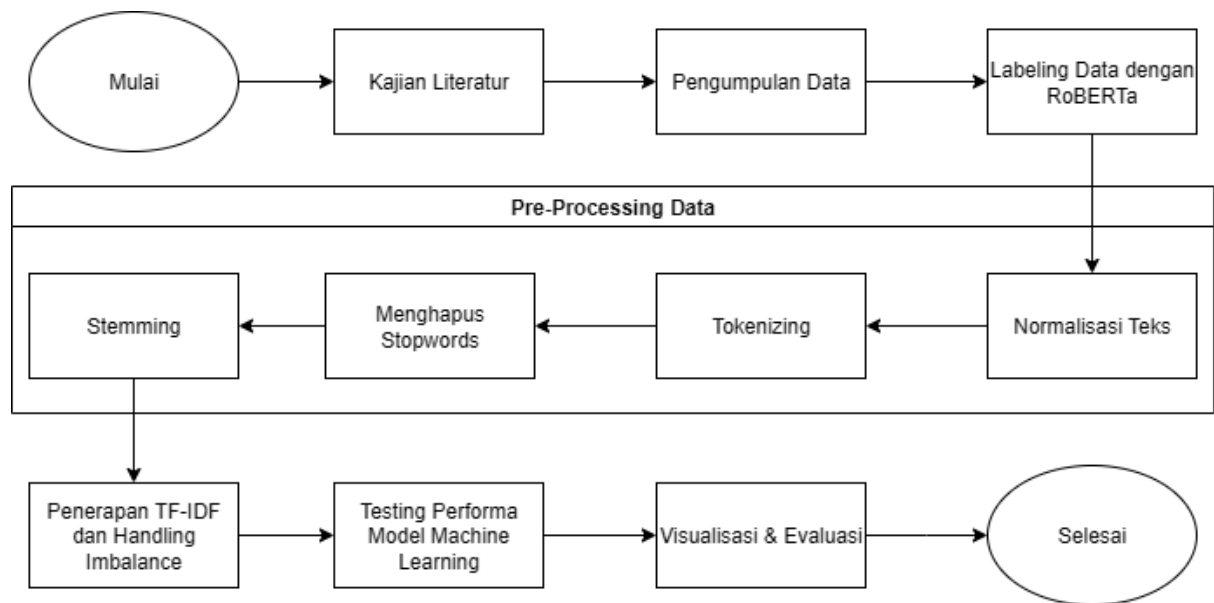
Hasil evaluasi kurva ROC juga menunjukkan kemampuan diskriminatif model yang tinggi; untuk kelas negatif, skor Area Under the Curve (AUC) adalah 0.96, dan untuk kelas netral dan positif masing-masing skor 0.97. Pada akhirnya, setelah diterapkan pada sampel komentar media sosial riil, model menunjukkan kemampuan prediksi yang andal, dengan rata-rata nilai prediksi 97.70%. Semua bukti ini secara kolektif menunjukkan bahwa model ini adalah standar yang kuat dan relevan untuk tugas pelabelan sentimen dalam Bahasa Indonesia.

BAB III

METODE PENELITIAN

3.1 Alur Penelitian

Tahapan penelitian merupakan alur yang dilalui oleh penulis untuk menyelesaikan penelitian yang bertujuan agar pengembangan penelitian ini menjadi lebih terarah, alur dapat dilihat pada gambar dibawah:

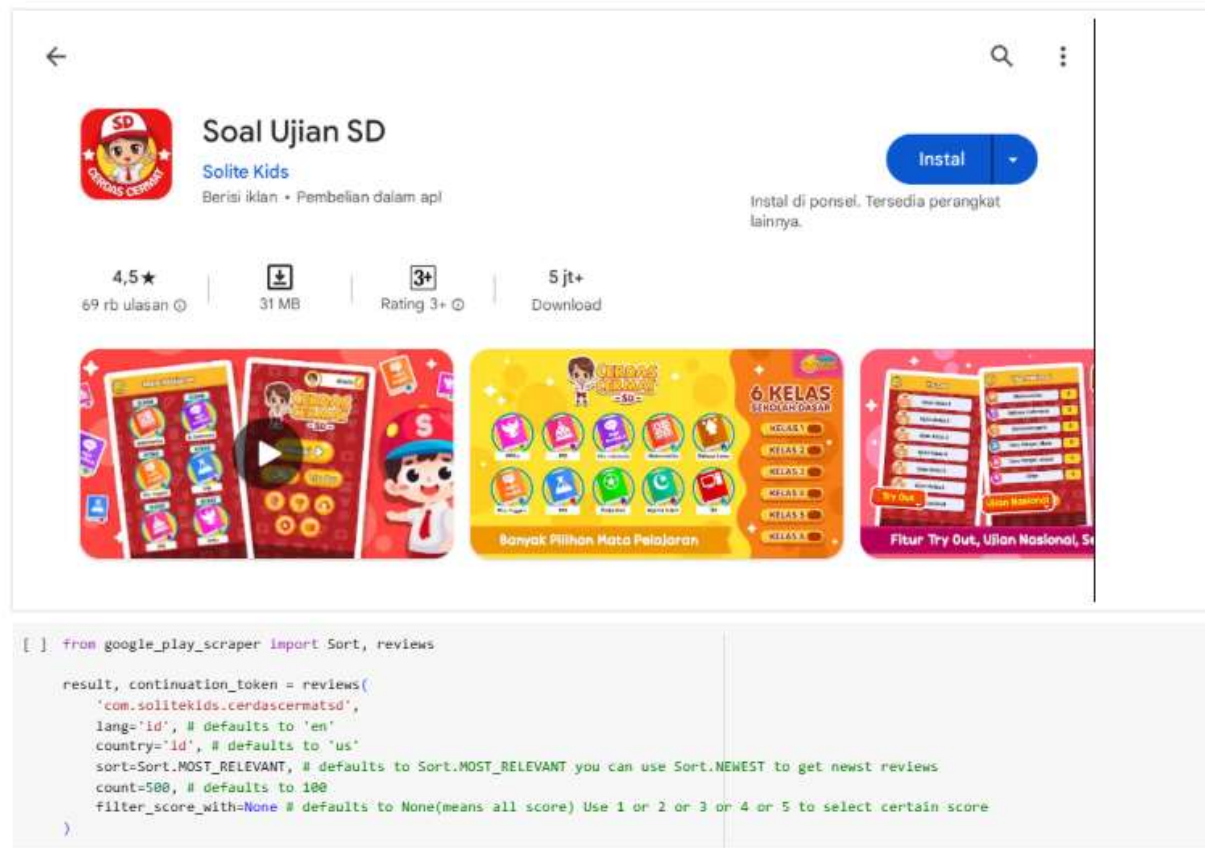


Gambar 3.1 Flowchart Alur Penelitian

3.2 Pengumpulan Data

Data yang digunakan bersumber dari ulasan pada *google play store* sampel 20 Aplikasi pendidikan berbahasa indonesia untuk anak-anak menggunakan metode scrapping data menggunakan library “google-play-scraper” pada python. Adapun untuk jumlah ulasan yang dikumpulkan berjumlah sekitar 500 ulasan paling relevan berbahasa indonesia yang menjadikan total data ulasan terkumpul berjumlah 9990 komentar.

✓ SCRAP APLIKASI 1 : Soal Ujian SD



Gambar 3.2 Capture Scrapping Data Dari Playstore

```
1 from google_play_scraper import Sort, reviews
2 result, continuation_token = reviews(
3     'com.solitekids.cerdascermatsd',
4     lang='id',
5     country='id',
6     sort=Sort.MOST_RELEVANT,
7     count=500,
8     filter_score_with=None
9 )
10 #Pembentukan Data Frame Hasil Scrap
11 df_busu = pd.DataFrame(np.array(result), columns=['review'])
12 df_busu = df_busu.join(pd.DataFrame(df_busu.pop('review').tolist()))
13 df_busu.head()
14
15 #Sortir Data Yang Diperlukan
16 new_df = df_busu[['userName', 'score', 'at', 'content']]
17 sorted_df = new_df.sort_values(by='at', ascending=False)
18 sorted_df.head()
```

```

19 my_df = sorted_df[['userName', 'score','at', 'content']]
20
21 #Export Data Menjadi File CSV
22 my_df.to_csv("scrapped_data_app_1.csv", index = False)

```

Gambar 3.3 Sintaks Scrapping Data

Pada sintaks tersebut digunakan untuk mengumpulkan data komentar ulasan aplikasi dengan sampel 1 aplikasi berjudul “Soal Ujian SD” dengan ketentuan data yang di ambil berupa : Nama pengulas, Skor rating bintang, waktu ulas dan isi komentar ulasan.

3.3 Variabel

Data yang telah didapatkan dengan metode scrapping sebelumnya di sortir kembali dan didapatkan variabel yang diperlukan untuk kebutuhan analisa berupa:

1. Nama Pengguna.
2. Rating Bintang.
3. Waktu Ulasan.
4. Isi Ulasan

3.4 Labeling Menggunakan RoBERTa

Hasil sortiran data diolah kembali untuk diberikan label penilaian positif, negatif maupun netral dengan memanfaatkan library pelabelan “w1lwo/indonesian-roberta-base-sentiment-classifier”(Wongso, 2023) sehingga progres labeling diharapkan bisa lebih efisien dan akurat mengingat data yang perlu di labeli berjumlah 9990 dan akan memakan banyak waktu untuk bisa melabeli satu persatu

```

1 #Mounting file CSV
2
3 from google.colab import drive
4 drive.mount('/content/drive')
5
6 #Install Library
7 from huggingface_hub import login
8 login(token="hf_ubXFcmatyLnHGzYGfOtZrirIWTESdtwPQ")
9 pip install pandas torch transformers

```

```

10
11 import pandas as pd
12 from transformers import pipeline
13 from tqdm.auto import tqdm
14
15 import logging
16 logging.getLogger("transformers").setLevel(logging.ERROR)
17
18 #Lokasi Pembacaan Data dan Lokasi Output Data
19
20 file_input_path = '/content/drive/MyDrive/csv/combined_csv.csv'
21 file_output_path = '/content/drive/MyDrive/reviews_with_sentiment_final.csv'
22 print(f"Membaca file dari: {file_input_path}...")
23 try:
24
25 df = pd.read_csv(file_input_path)
26 except FileNotFoundError:
27
28 print(f"Error: File tidak ditemukan di path '{file_input_path}'")
29 else:
30
31 # Membersihkan data
32 df.dropna(subset=['content'], inplace=True)
33 df['content'] = df['content'].astype(str)
34 print("Memuat model Pelabelan Sentimen (RoBERTa)...")
35 sentiment_analyzer = pipeline(

```

Gambar 3.4 Sintaks Labeling Menggunakan RoBERTa

Tujuan dari sintaks kode di atas adalah untuk melakukan klasifikasi sentimen secara otomatis pada dataset ulasan berbahasa Indonesia yang berasal dari file CSV. Tahap pra-pemrosesan dilakukan untuk membersihkan data teks, dan kemudian digunakan model pemrosesan bahasa alami (NLP) berbasis RoBERTa, yang dilatih khusus untuk sentimen Bahasa Indonesia ([w11wo/indonesian-roberta-base-sentiment-classifier](#)). Tujuannya adalah untuk melakukan klasifikasi sentimen secara otomatis pada sebuah dataset ulasan berbahasa Indonesia yang berasal dari file CSV, dimulai dengan tahap pra-pemrosesan untuk membersihkan data teks, kemudian setiap ulasan dianalisis untuk diberi label sentimen (positif, negatif, atau netral), dan hasilnya disimpan kembali ke dalam sebuah file CSV baru. Selanjutnya, setiap ulasan dievaluasi untuk diberi label sentimen (positif, negatif, atau netral), dan hasilnya disimpan kembali ke dalam file CSV baru. Ini menghasilkan dataset yang dipercaya dengan atribut sentimen untuk analisis tambahan.

3.5 Pre-processing Data

Pre-processing dilakukan untuk menghindari data yang berlebihan atau cacat, Ada beberapa tahapan preprocessing dalam penelitian ini.yang akan dilakukan dengan metode normalisasi teks berikut:

a. Folding Case & Symbol Removal

Proses ini berguna untuk menyetarakan semua huruf menggunakan huruf kecil dan juga membersihkan kalimat ulasan dari simbol-simbol yang tidak diperlukan sehingga tidak terjadi error dalam analisa kata yang sama namun berbeda dalam penulisan huruf kapital.

```

1 df['content'] = df['content'].str.lower()
2 df['content'] = df['content'].str.replace('@\S+', ' ', case=False)
3 df['content'] = df['content'].str.replace('#\S+', ' ', case=False)
4 df['content'] = df['content'].str.replace("\'\'w+", ' ', case=False)
5 df['content'] = df['content'].str.replace("[^\w\s]", ' ', case=False)
6 df['content'] = df['content'].str.replace("\s(2)", ' ', case=False)
7 df['content'] = df['content'].str.replace('https\S+', ' ', case=False)

```

Gambar 3.5 Sintaks folding case dan symbol removal

Tabel 3.1 Perbandingan sebelum dan sesudah folding case dan symbol removal

Sebelum	Sesudah
Bagus banget tapi sayang nya kenapa cuma ada versi androidnya aja, klo dibuatkan film nya mungkin lebih lucu ya krna sayang yg cuma punya memory dikit ga bsa donlot semuanya	bagus banget tapi sayang nya kenapa cuma ada versi androidnya aja, klo dibuatkan film nya mungkin lebih lucu ya krna sayang yg cuma punya memory dikit ga bsa donlot semuanya
Coba	coba
kadang suaranya ilang	kadang suaranya ilang
assalamualaikum namaku adalah annisa sebelumnya nilai ujian agamaku turun yaitu 68 tapi sekarang nilaiku 89 apl ini sangat bagus	assalamualaikum namaku adalah annisa sebelumnya nilai ujian agamaku turun yaitu 68 tapi sekarang nilaiku 89 apl ini sangat bagus
di coba dlu ya	di coba dlu ya
assallamuallaikum....ini komentar dati saya.sangat mambantu dalam belajar,terus bisa jadi bimbingan belajar.kalo kita siang siang bosan bisa main aja game ini bisa sekalian buat belajar,tapi ya jangan(kecanduan) ngelupain buku,bukulah bimbingan belajar(bimbel) yang	Assallamuallaikum ini komentar dati saya.sangat mambantu dalam belajar terus bisa jadi bimbingan belajar kalo kita siang siang bosan bisa main aja game ini bisa sekalian buat belajar,tapi ya jangan(kecanduan) ngelupain buku,bukulah bimbingan belajar(bimbel) yang

paling penting...ini saja komentar saya.wassallamuallaikum	paling penting ini saja komentar saya wassallamuallaikum
segitu doang,...anak cepat bosan	segitu doang anak cepat bosan
Permainan nya bagus tapi iklan nya yang terus terus an aku ga mau ada iklan nya mohon perbaiki yah permainan ini yang menga dakan mang kanya saya kasih bintang 4 nya oh ita nama saya zelda s. Saya pake hp mama saya asalamualaikum dadah 🤞	permainan nya bagus tapi iklan nya yang terus terus an aku ga mau ada iklan nya mohon perbaiki yah permainan ini yang menga dakan mang kanya saya kasih bintang 4 nya oh ita nama saya zelda saya pake hp mama saya asalamualaikum dadah
Terima kasih ya yang udah membuat gamekarena game ini aku dan adikku saat bermain aku dan adikku jadi lebih tahu agama Islam terima kasih wassalamualaikum warahmatullahi wabarakatuh aku Islam aku Priska Maharani Wulansari	terima kasih ya yang udah membuat gamekarena game ini aku dan adikku saat bermain aku dan adikku jadi lebih tahu agama islam terima kasih wassalamualaikum warahmatullahi wabarakatuh aku islam aku priska maharani wulansari
Assalamu alaikum aku suka bgt apk ini karna aplikasinya bgs bgt bisa bljr solat.ayoo buruan donlowd pasti enggak nyesel donlownya ini bgd bgt apk mantap bnr	assalamu alaikum aku suka bgt apk ini karna aplikasinya bgs bgt bisa bljr solat ayoo buruan donlowd pasti enggak nyesel donlownya ini bgd bgt apk mantap bnr

b. Tokenizing

Tokenizing berguna untuk menguraikan teks menjadi bagian kecil, seperti kata atau frasa. Langkah ini dilakukan dengan tujuan menyederhanakan data agar siap untuk tahap pemrosesan lanjutan, seperti analisis statistik atau pembuatan model bahasa.

```

1 from nltk.tokenize import word_tokenize
2 from nltk.tokenize import RegexpTokenizer
3 regexp = RegexpTokenizer('\w+')
4 df['content_token']=df['content'].apply(regexp.tokenize)
5 df.head(10)

```

Gambar 3.6 Sintaks Tokenizing

Tabel 3.2 Perbandingan sebelum dan sesudah Tokenizing

Sebelum	Sesudah
bagus banget tapi sayang nya kenapa cuma ada versi androidnya aja, klo dibuatkan film nya mungkin lebih lucu ya krna sayang yg cuma punya memory dikit ga bsa donlot semuanya	bagus,banget,tapi,sayang,nya,kenapa,cuma,ad a,versi,androidnya,aja,klo,dibuatkan,film,nya, mungkin,lebih,lucu,ya,krna,sayang,yg,cuma,p unya,memory,dikit,ga,bsa,donlot,semuanya

coba	coba
kadang suaranya ilang	kadang,suaranya,ilang
assalamualaikum namaku adalah annisa sebelumnya nilai ujian agamaku turun yaitu 68 tapi sekarang nilaiku 89 apl ini sangat bagus	assalamualaikum,namaku,adalah,annisa,sebel umnya,nilai,ujian,agamaku,turun,yaitu,68,tapi, sekarang,nilaiku,89,apl,ini,sangat,bagus
di coba dlu ya	di,coba,dlu,ya
Assallamuallaikum ini komentar dari saya.sangat mambantu dalam belajar terus bisa jadi bimbingan belajar kalo kita siang siang bosan bisa main aja game ini bisa sekalian buat belajar,tapi ya jangan(kecanduan) ngelupain buku,bukulah bimbingan belajar(bimbel) yang paling penting ini saja komentar saya wassallamuallaikum	assallamuallaikum,ini,komentar,dari,saya,sang at,mambantu,dalam,belajar,terus,bisa,jadi,bim bingan,belajar,kalo,kita,siang,siang,bosan,bisa ,main,aja,game,ini,bisa,sekalian,buat,belajar,ta pi,ya,jangan,kecanduan,ngelupain,buku,bukul ah,bimbingan,belajar,bimbel,yang,paling,penti ng,ini,saja,komentar,saya,wassallamuallaikum
segitu doang anak cepat bosan	segitu,doang,anak,cepat,bosan
permainan nya bagus tapi iklan nya yang terus terus an aku ga mau ada iklan nya mohon perbaiki yah permainan ini yang menga dakan mang kanya saya kasih bintang 4 nya oh ita nama saya zelda saya pake hp mama saya assalamualaikum dadah	permainan,nya,bagus,tapi,iklan,nya,yang,terus ,terus,an,aku,ga,mau,ada,iklan,nya,mohon,per baiki,yah,permainan,ini,yang,menga,dakan,ma ng,kanya,saya,kasih,bintang,4,nya,oh,ita,nama ,saya,zelda,s,saya,pake,hp,mama,saya,assalamu alaikum,dadah
terima kasih ya yang udah membuat gamekarena game ini aku dan adikku saat bermain aku dan adikku jadi lebih tahu agama islam terima kasih wassalamualaikum warahmatullahi wabarakatuh aku islam aku priska maharani wulansari	terima,kasih,ya,yang,udah,membuat,gamekare na,game,ini,aku,dan,adikku,saat,bermain,aku, dan,adikku,jadi,lebih,tahu,agama,islam,terima, kasih,wassalamualaikum,warahmatullahi,wab arakatuh,aku,islam,aku,priska,maharani,wulan sari
assalamu alaikum aku suka bgt apk ini karna aplikasinya bgs bgt bisa bljr solat ayoo buruan donlowd pasti enggak nyesel donlownya ini bgd bgt apk mantap bnr	assalamu,alaikum,aku,suka,bgt,apk,ini,karna,a plikasinya,bgs,bgt,bisa,bljr,solat,ayoo,buruan, donlowd,pasti,enggak,nyesel,donlownya,ini,bg d,bgt,apk,mantap,bnr

c. Stopwords Removal

Proses ini berfungsi untuk menghilangkan kata kata yang tidak memiliki makna maupun informasi yang tidak signifikan, tipe kata-kata seperti ini cenderung bisa diabaikan dan dihapus, kata yang dipertahankan merupakan kata-kata yang memiliki informasi relevan dan signifikan, dalam hal ini ditambahkan kamus kustom untuk mencakup beberapa kata yang ditemukan pada pengetesan sebelumnya dirasa lolos dari filter kamus NLTK.

```

1 nltk.download('stopwords')
2 from nltk.corpus import stopwords
3
4 stopwords = nltk.corpus.stopwords.words("indonesian")
5 my_stopwords = ['gak', 'nggak', 'ga', 'sih', 'deh', 'biar', 'aja', 'yg',
6 'dgn', 'kalo', 'dll', 'nya', 'klo', 'kalo', 'ya', 'assalamualaikum',
7 'assalamualaikum', 'assallamuallaikum', 'ngak']
8 stopwords.extend(my_stopwords)
9
10 df['content_token'] = df['content_token'].apply(lambda x: [item for
11 item in x if item not in stopwords])
12 df.head(10)

```

Gambar 3.7 Sintaks Stopwords Removal

Tabel 3.3 Perbandingan Proses Stopwords Removal

Sebelum	Sesudah
bagus,banget,tapi,sayang,nya,kenapa,cuma,ada,versi,androidnya,aja,klo,dibuatkan,film,nya,mungkin,lebih,lucu,ya,krna,sayang,yg,cuma,punya,memory,dikit,ga,bsa,donlot,semuanya	bagus,banget,sayang,versi,androidnya,dibuatkan,film,lucu,krna,sayang,memory,dikit,bsa,donlot
coba	coba
kadang,suaranya,ilang	kadang,suaranya,ilang
assalamualaikum,namaku,adalah,annisa,sebelumnya,nilai,ujian,agamaku,turun,yaitu,68,tapi,sekarang,nilaiku,89,apl,ini,sangat,bagus	namaku,annisa,nilai,ujian,agamaku,turun,68,nilaiku,89,apl,bagus
di,coba,dlu,ya	coba,dlu
assallamuallaikum,ini,komentar,dati,saya,sangat,mambantu,dalam,belajar,terus,bisa,jadi,bimbingan,belajar,kalo,kita,siang,siang,bosen,bisa	komentar,dati,mambantu,belajar,bimbingan,belajar,siang,siang,bosen,main,game,belajar,kec

,main,aja,game,ini,bisa,sekalian,buat,belajar,ta pi,ya,jangan,kecanduan,ngelupain,buku,bukul ah,bimbingan,belajar,bimbel,yang,paling,penti ng,ini,saja,komentar,saya,wassallamuallaikum	anduan,ngelupain,buku,bukulah,bimbingan,be lajar,bimbel,komentar,wassallamuallaikum
segitu,doang,anak,cepat,bosan	segitu,doang,anak,cepat,bosan
permainan,nya,bagus,tapi,iklan,nya,yang,terus ,terus,an,aku,ga,mau,ada,iklan,nya,mohon,per baiki,yah,permainan,ini,yang,menga,dakan,ma ng,kanya,saya,kasih,bintang,4,nya,oh,ita,nama ,saya,zelda,s,saya,pake,hp,mama,saya,asalamu alaikum,dadah	permainan,bagus,iklan,an,iklan,mohon,perbai ki,yah,permainan,menga,dakan,mang,kanya,k asih,bintang,4,oh,ita,nama,zelda,s,pake,hp,ma ma,asalamualaikum,dadah
terima,kasih,ya,yang,udah,membuat,gamekare na,game,ini,aku,dan,adikku,saat,bermain,aku, dan,adikku,jadi,lebih,tahu,agama,islam,terima, kasih,wassalamualaikum,warahmatullahi,wab arakatuh,aku,islam,aku,priska,maharani,wulan sari	terima,kasih,udah,gamekarena,game,adikku,b ermain,adikku,agama,islam,terima,kasih,wass alamualaikum,warahmatullahi,wabarakatuh,isl am,priska,maharani,wulansari
assalamu,alaikum,aku,suka,bgt,apk,ini,karna,a plikasinya,bgs,bgt,bisa,bljr,solat,ayoo,buruan, donlowd,pasti,engak,nyesel,donlownya,ini,bg d,bgt,apk,mantap,bnr	assalamu,alaikum,suka,bgt,apk,karna,aplikasi nya,bgs,bgt,bljr,solat,ayoo,buruan,donlowd,en gak,nyesel,donlownya,bgd,bgt,apk,mantap,bnr

d. Stemming

Mengubah berbagai variasi kata ke bentuk dasarnya atau kata akarnya adalah kegunaan utama stemming dalam analisis sentimen; ini membuat informasi sentimen lebih kuat dan terkonsolidasi pada satu kata dasar, dan jumlah kata unik yang ada dalam data berkurang. Ini membantu model berkonsentrasi pada makna kata daripada bentuk gramatikalnya, meningkatkan kemampuan model untuk mengidentifikasi pola sentimen yang luas pada kesempatan ini penulis menggunakan library sastrawi untuk proses stemming.

```

1 !pip install Sastrawi
2 from Sastrawi.Stemmer.StemmerFactory import StemmerFactory
3
4 factory = StemmerFactory()
5 stemmer = factory.create_stemmer()
6
7 df['stemmed'] = df['content_token'].apply(lambda x: [stemmer.stem(y) for
y in x])

```

Gambar 3.8 Sintaks Stemming

Tabel 3.4 Perbandingan sebelum dan sesudah Stemming

Sebelum	Sesudah
bagus,banget,sayang,versi,androidnya,dibuatkan,film,lucu,krna,sayang,memory,dikit,bsa,donlot	bagus,banget,sayang,versi,androidnya,buat,film,lucu,krna,sayang,memory,dikit,bsa,donlot
coba	coba
kadang,suaranya,ilang	kadang,suara,ilang
namaku,annisa,nilai,ujian,agamaku,turun,68,nilaiku,89,apl,bagus	nama,annisa,nilai,uji,agama,turun,68,nilai,89,apl,bagus
coba,dlu	coba,dlu
komentar,dati,mambantu,belajar,bimbingan,belajar,siang,siang,bosen,main,game,belajar,kecanduan,ngelupain,buku,bukulah,bimbingan,belajar,bimbel,komentar,wassallamuallaikum	komentar,dati,mambantu,ajar,bimbing,ajar,siang,siang,bosen,main,game,ajar,candu,ngelupain,buku,buku,bimbing,ajar,bimbel,komentar,wassallamuallaikum
segitu,doang,anak,cepat,bosan	segitu,doang,anak,cepat,bosan
permainan,bagus,iklan,an,iklan,mohon,perbaiki,yah,permainan,menga,dakan,mang,kanya,kasih,bintang,4,oh,ita,nama,zelda,s,pake,hp,mama,asalamualaikum,dadah	main,bagus,iklan,an,iklan,mohon,bai,yah,main,menga,dakan,mang,kanya,kasih,bintang,4,oh,ita,nama,zelda,s,pake,hp,mama,asalamualaikum,dadah
terima,kasih,udah,gamekarena,game,adikku,bermain,adikku,agama,islam,terima,kasih,wassalamualaikum,warahmatullahi,wabarakatuh,islam,priska,maharani,wulansari	terima,kasih,udah,gamekarena,game,adik,main,adik,agama,islam,terima,kasih,wassalamualaikum,warahmatullahi,wabarakatuh,islam,priska,maharani,wulansari
assalamu,alaikum,suka,bgt,apk,karna,aplikasinya,bgs,bgt,bljr,solat,ayoo,buruan,donlowd,enggak,nyesel,donlownya,bgd,bgt,apk,mantap,bnr	assalamu,alaikum,suka,bgt,apk,karna,aplikasi,bgs,bgt,bljr,solat,ayoo,buru,donlowd,enggak,nyesel,donlownya,bgd,bgt,apk,mantap,bnr

3.6 Penerapan TF-IDF

Penelitian ini menggunakan metode pembobotan TF-IDF (Term Frequency-Inverse Document Frequency) untuk mengubah data teks menjadi format numerik yang dapat diolah oleh algoritma pembelajaran mesin. TF-IDF adalah ukuran statistik yang mengevaluasi relevansi kata (term) dalam suatu dokumen terhadap kumpulan dokumen secara keseluruhan. Frekuensi kata dalam satu dokumen disebut Frekuensi Kata (TF). Frekuensi Dokumen Inverse (IDF) juga digunakan untuk menghitung frekuensi kata yang jarang muncul di seluruh korpus dan sebaliknya. Dengan mengalikan nilai TF dan IDF, sebuah skor dibuat yang menunjukkan pentingnya setiap kata. Kata-kata yang sering muncul dalam satu dokumen tetapi langka di dokumen lain akan memiliki skor TF-IDF yang tinggi, yang dapat digunakan sebagai fitur yang efektif untuk mengidentifikasi karakteristik khusus dokumen dalam proses klasifikasi.

```
1 df['text_string']
2
3 X = df['text_string']
4 y = df['label']
5
6 tfidf = TfidfVectorizer()
7 X_final = tfidf.fit_transform(X)
```

Gambar 3.9 Sintaks Penerapan TF-IDF

3.7 Handling Data Imbalance

Dalam dataset penelitian ini, distribusi kelas menyebabkan ketidakseimbangan data, atau ketidakseimbangan data. Dalam ketidakseimbangan data, jumlah sampel dari satu kelas lebih dominan daripada sampel dari kelas lainnya. Dalam situasi seperti ini, model klasifikasi mungkin bias terhadap kelas mayoritas dan gagal menangani kelas minoritas. Untuk memecahkan masalah tersebut, penelitian ini menggunakan metode oversampling SMOTE (Teknik Oversampling Minoritas Sintetis) untuk membuat sampel-sampel baru untuk kelas minoritas. Proses ini dilakukan dengan memilih sampel minoritas, menentukan tetangga terdekat (disebut k-nearest neighbors), dan kemudian membuat data sintetis baru di sepanjang garis yang menghubungkan sampel dengan tetangga yang dipilih. Sehingga model dapat mempelajari pola dari kelas minoritas secara lebih efektif dan menghasilkan performa klasifikasi yang lebih andal, penerapan SMOTE bertujuan untuk menyeimbangkan distribusi kelas pada data latih.

```
#Data sebelum SMOTE
print(y.value_counts())

label
positive      7796
negative      1603
neutral        591
Name: count, dtype: int64

smote = SMOTE()
x_sm,y_sm = smote.fit_resample(X_final,y)
#Data setelah SMOTE
label
positive      7796
negative      7796
neutral      7796
Name: count, dtype: int64
```

Gambar 3.10 Sintaks SMOTE Untuk *Handling Data Imbalance*

BAB IV

HASIL DAN PEMBAHASAN

4.1 Pengetesan Performa

Analisis komparatif dilakukan untuk mengevaluasi dan membandingkan kinerja tiga algoritma pembelajaran mesin: *Random Forest*, *Support Vector Machine* (SVM), dan *Naive Bayes*. Tujuan utama dari perbandingan ini adalah untuk menemukan model klasifikasi mana yang dapat melakukan analisis sentimen pada ulasan aplikasi pendidikan anak berbahasa Indonesia. Penelitian ini bertujuan untuk memberikan justifikasi berbasis data mengenai algoritma mana yang paling andal untuk memahami opini pengguna dalam situasi ini dengan menguji setiap model pada dataset khusus ini.

Setiap algoritma yang dipilih memiliki pendekatan dasar yang berbeda untuk menganalisis data teks. *Random Forest* diuji sebagai model kelompok yang diharapkan dapat menangani variasi kalimat yang kompleks dalam ulasan. Kemampuan *Support Vector Machine* (SVM) untuk menemukan batas keputusan yang jelas antara sentimen positif, negatif, dan netral dinilai. Namun, *Naive Bayes* dimasukkan sebagai model probabilistik yang dikenal cepat dan efektif untuk tugas klasifikasi teks, meskipun model ini cenderung mengarah ke independensi antar kata.

Dalam evaluasi model, *Weighted Average* digunakan untuk menunjukkan pengukuran performa agregat yang paling representatif dan adil. Ini terutama berlaku untuk set data pengujian di mana mungkin ada ketidakseimbangan kelas. Metrik ini menghitung skor precision, recall, dan F1 rata-rata, dengan bobot setiap kelas proporsional dengan jumlah sampel sebenarnya (support). Metode ini memastikan bahwa kinerja model pada kelas mayoritas yang lebih dominan memberikan kontribusi yang lebih besar terhadap skor akhir. Ini menghasilkan evaluasi keseluruhan yang menunjukkan kemampuan prediksi model untuk distribusi data yang sebenarnya. Untuk memberikan validasi model yang lebih luas dan dapat diandalkan dan untuk menghindari penilaian macro average, yang memperlakukan setiap kelas secara sama.

Perbandingan langsung ini akan melihat bagaimana setiap pendekatan berpengaruh terhadap metrik evaluasi seperti akurasi, presisi, recall, dan skor F1. Hasil evaluasi ini diharapkan dapat memberikan rekomendasi model terbaik serta informasi tentang kekuatan dan kelemahan masing-masing algoritma dalam mengklasifikasikan sentimen pada ulasan aplikasi pendidikan anak yang menggunakan bahasa Indonesia.

4.2 Testing performa dengan berbagai rasio data uji dan latih

Berikut terlampir hasil pengujian performa untuk tiga model klasifikasi yang telah dilatih Random Forest, Support Vector Machine (SVM), dan Naive Bayes disajikan di sini. Untuk melakukan evaluasi, skema pembagian data digunakan untuk membagi data menjadi beberapa skenario rasio data uji dan data latih. Untuk mengevaluasi kinerja setiap model, metrik standar seperti akurasi, presisi, recall, dan skor F1 digunakan. Hasilnya disajikan secara rinci untuk perbandingan.

```
1 #Split Data dengan rasio 90 : 10
2 X_train , X_test , y_train , y_test = train_test_split(x_sm , y_sm ,
test_size=0.1,random_state=3)
```

Gambar 4.1 Sintaks Splitting Data Rasio

Dalam sintaks di atas digunakan untuk membagi rasio data uji dan data latih dengan cara merubah variabel test size dari 0.1 untuk rasio 90 data latih dan 10 data uji hingga ke 0.9 untuk rasio 10 data latih dan 90 data uji.

```
1 random_forest_classifier = RandomForestClassifier()
2 random_forest_classifier.fit(X_train, y_train)
3
4 random_forest_classifier_prediction=
random_forest_classifier.predict(X_test)
5
6 accuracy = accuracy_score(y_test, random_forest_classifier_prediction)
7 precision = precision_score(y_test, random_forest_classifier_prediction,
average='weighted')
8 recall = recall_score(y_test, random_forest_classifier_prediction,
average='weighted')
9 f1 = f1_score(y_test, random_forest_classifier_prediction,
average='weighted')
```

Gambar 4.2 Sintaks Testing Model Random Forest

Model klasifikasi Random Forest digunakan untuk membuat prediksi dari set data yang terpisah. Pertama, model dilatih (fit) untuk mempelajari pola-pola yang ada pada data menggunakan data latih (X_train, y_train). Kemudian, kemampuan generalisasi model diuji

dengan menggunakannya untuk membuat prediksi (predict) pada data uji independen dan belum pernah terlihat sebelumnya (X_{test}). Selanjutnya, kinerja model diukur secara kuantitatif dengan menggunakan metrik standar untuk akurasi, presisi, recall, dan skor F1 yang disajikan dalam Laporan Klasifikasi. Ini memberikan analisis menyeluruh tentang efektivitas prediktif model pada setiap kelas target.

```
1  svm = SVC()
2  svm.fit(X_train, y_train)
3
4  svm_prediction = svm.predict(X_test)
5
6  accuracy = accuracy_score(y_test, svm_prediction)
7  precision = precision_score(y_test, svm_prediction, average='weighted')
8  recall = recall_score(y_test, svm_prediction, average='weighted')
9  f1 = f1_score(y_test, svm_prediction, average='weighted')
10
11 print(f"Accuracy: {accuracy}")
12 print(f"Precision: {precision}")
13 print(f"Recall: {recall}")
```

Gambar 4.3 Sintaks Testing Model SVM

Pada tahap ini, digunakan model klasifikasi Support Vector Machine (SVMmodel klasifikasi Support Vector Machine (SVM) dan melatihnya menggunakan data latih (X_{train} , y_{train}). Setelah model terlatih, kode ini menggunakannya untuk membuat prediksi pada data uji (X_{test}) yang belum pernah dilihat sebelumnya. Akhirnya, kode tersebut mengevaluasi kinerja model dengan menghitung dan menampilkan metrik-metrik penting seperti accuracy (akurasi), precision (presisi), dan recall (daya ingat) dengan membandingkan hasil prediksi model dengan label data uji yang sebenarnya (y_{test}).

```
1 nb = MultinomialNB()
2 nb.fit(X_train, y_train)
3
4 nb_prediction = nb.predict(X_test)
5
6 accuracy = accuracy_score(y_test, nb_prediction)
7 precision = precision_score(y_test, nb_prediction, average='weighted')
8 recall = recall_score(y_test, nb_prediction, average='weighted')
9 f1 = f1_score(y_test, nb_prediction, average='weighted')
10
11 print(f"Accuracy: {accuracy}")
12 print(f"Precision: {precision}")
```

Gambar 4.4 Sintaks Testing Model Naive Bayes

Bagian ini membahas model Multinomial Naive Bayes (MNB), lalu melatih model tersebut menggunakan data latih (`X_train` dan `y_train`). Setelah dilatih, model itu digunakan untuk membuat prediksi pada data uji (`X_test`). Akhirnya, performa model dievaluasi dengan menghitung dan mencetak nilai accuracy (akurasi) dan precision (presisi) dengan cara membandingkan hasil prediksi (`nb_prediction`) dengan label data uji yang sesungguhnya (`y_test`).

4.3 Hasil Tes Performa :

```
#Output 90:10
```

```
Random Forest
```

```
Accuracy: 0.926891834117144
```

```
Precision: 0.9295663172773097
```

```
Recall: 0.926891834117144
```

```
F1 Score: 0.9266334357692448
```

```
=====
```

```
Laporan Klasifikasi (Classification Report):
```

	precision	recall	f1-score	support
negative	0.90	0.96	0.92	812
neutral	0.92	0.96	0.94	732
positive	0.97	0.87	0.92	795
accuracy			0.93	2339
macro avg	0.93	0.93	0.93	2339
weighted avg	0.93	0.93	0.93	2339

```
SVM
```

```
Accuracy: 0.920478837109876
```

```
Precision: 0.9252546223909276
```

```
Recall: 0.920478837109876
```

```
F1 Score: 0.9203529195698191
```

```
=====
```

```
Laporan Klasifikasi (Classification Report):
```

	precision	recall	f1-score	support
negative	0.95	0.85	0.90	812
neutral	0.86	0.99	0.92	732
positive	0.96	0.92	0.94	795
accuracy			0.92	2339
macro avg	0.92	0.92	0.92	2339
weighted avg	0.93	0.92	0.92	2339

Naive Bayes

Accuracy: 0.8225737494655836

Precision: 0.8240660045346344

Recall: 0.8225737494655836

F1 Score: 0.8229167160575217

=====

Laporan Klasifikasi (Classification Report):

	precision	recall	f1-score	support
negative	0.87	0.83	0.85	812
neutral	0.81	0.80	0.80	732
positive	0.79	0.83	0.81	795
accuracy			0.82	2339
macro avg	0.82	0.82	0.82	2339
weighted avg	0.82	0.82	0.82	2339

Gambar 4.5 Output Rasio 90:10

#Output 80:20

Random Forest

Accuracy: 0.9221889696451475

Precision: 0.9241642435039678

Recall: 0.9221889696451475

F1 Score: 0.9216114744867704

=====

Laporan Klasifikasi (Classification Report):

	precision	recall	f1-score	support
negative	0.89	0.95	0.92	1585
neutral	0.93	0.97	0.95	1508
positive	0.96	0.85	0.90	1585
accuracy			0.92	4678
macro avg	0.92	0.92	0.92	4678
weighted avg	0.92	0.92	0.92	4678

SVM

Accuracy: 0.9187687045746046

Precision: 0.923005619388091

Recall: 0.9187687045746046

F1 Score: 0.918312776535617

=====

Laporan Klasifikasi (Classification Report):

	precision	recall	f1-score	support
negative	0.95	0.84	0.89	1585
neutral	0.86	0.99	0.92	1508
positive	0.95	0.93	0.94	1585
accuracy			0.92	4678
macro avg	0.92	0.92	0.92	4678
weighted avg	0.92	0.92	0.92	4678

Naive Bayes

Accuracy: 0.8172295852928602

Precision: 0.8183703657526683

Recall: 0.8172295852928602

F1 Score: 0.8173468315248991

=====

Laporan Klasifikasi (Classification Report):

	precision	recall	f1-score	support
negative	0.85	0.83	0.84	1585
neutral	0.82	0.79	0.80	1508
positive	0.78	0.84	0.81	1585
accuracy			0.82	4678
macro avg	0.82	0.82	0.82	4678
weighted avg	0.82	0.82	0.82	4678

Gambar 4.6 Output Rasio 80:20

#OUTPUT 70:30

RANDOM FOREST

Accuracy: 0.9169160609947271
 Precision: 0.9191827609297362
 Recall: 0.9169160609947271
 F1 Score: 0.9163638167361022

=====

Laporan Klasifikasi (Classification Report):

	precision	recall	f1-score	support
negative	0.88	0.94	0.91	2397
neutral	0.92	0.97	0.94	2287
positive	0.96	0.84	0.90	2333
accuracy			0.92	7017
macro avg	0.92	0.92	0.92	7017
weighted avg	0.92	0.92	0.92	7017

SVM

Accuracy: 0.9126407296565484
 Precision: 0.9171189592129718
 Recall: 0.9126407296565484
 F1 Score: 0.9120157930727084

=====

Laporan Klasifikasi (Classification Report):

	precision	recall	f1-score	support
negative	0.95	0.83	0.88	2397
neutral	0.86	0.98	0.92	2287
positive	0.95	0.93	0.94	2333
accuracy			0.91	7017
macro avg	0.92	0.91	0.91	7017
weighted avg	0.92	0.91	0.91	7017

NAIVE BAYES

Accuracy: 0.8058999572466866

Precision: 0.8085054112604037

Recall: 0.8058999572466866

F1 Score: 0.8062068770798769

=====

Laporan Klasifikasi (Classification Report):

	precision	recall	f1-score	support
negative	0.86	0.80	0.83	2397
neutral	0.81	0.78	0.79	2287
positive	0.76	0.84	0.80	2333
accuracy			0.81	7017
macro avg	0.81	0.81	0.81	7017
weighted avg	0.81	0.81	0.81	7017

Gambar 4.7 Output Rasio 70:30

#OUTPUT 60:40

RANDOM FOREST

Accuracy: 0.9078666096622489

Precision: 0.9107184701678668

Recall: 0.9078666096622489

F1 Score: 0.9073383131029196

=====

Laporan Klasifikasi (Classification Report):

	precision	recall	f1-score	support
negative	0.87	0.93	0.90	3159
neutral	0.91	0.96	0.93	3070
positive	0.95	0.83	0.89	3127
accuracy			0.91	9356
macro avg	0.91	0.91	0.91	9356
weighted avg	0.91	0.91	0.91	9356

SVM

Accuracy: 0.9050876442924327

Precision: 0.9108108188688953

Recall: 0.9050876442924327

F1 Score: 0.9039565225745729

=====

Laporan Klasifikasi (Classification Report):

	precision	recall	f1-score	support
negative	0.95	0.80	0.87	3159
neutral	0.85	0.98	0.91	3070
positive	0.93	0.93	0.93	3127
accuracy			0.91	9356
macro avg	0.91	0.91	0.90	9356
weighted avg	0.91	0.91	0.90	9356

NAIVE BAYES

Accuracy: 0.806434373663959

Precision: 0.8093987519322567

Recall: 0.806434373663959

F1 Score: 0.806649379953796

=====

Laporan Klasifikasi (Classification Report):

	precision	recall	f1-score	support
negative	0.86	0.80	0.83	3159
neutral	0.81	0.78	0.79	3070
positive	0.76	0.85	0.80	3127
accuracy			0.81	9356
macro avg	0.81	0.81	0.81	9356
weighted avg	0.81	0.81	0.81	9356

Gambar 4.8 Output Rasio 60:40

```
#OUTPUT 50:50
```

```
RANDOM FOREST
```

```
Accuracy: 0.8983239268000685
```

```
Precision: 0.901132590468868
```

```
Recall: 0.8983239268000685
```

```
F1 Score: 0.8978378362800361
```

```
=====
```

```
Laporan Klasifikasi (Classification Report):
```

	precision	recall	f1-score	support
negative	0.86	0.92	0.89	3955
neutral	0.90	0.95	0.92	3850
positive	0.94	0.83	0.88	3889
accuracy			0.90	11694
macro avg	0.90	0.90	0.90	11694
weighted avg	0.90	0.90	0.90	11694

```
SVM
```

```
Accuracy: 0.8932786044125193
```

```
Precision: 0.8991789254497596
```

```
Recall: 0.8932786044125193
```

```
F1 Score: 0.8917492754139978
```

```
=====
```

```
Laporan Klasifikasi (Classification Report):
```

	precision	recall	f1-score	support
negative	0.94	0.78	0.85	3955
neutral	0.84	0.97	0.90	3850
positive	0.91	0.93	0.92	3889
accuracy			0.89	11694
macro avg	0.90	0.89	0.89	11694
weighted avg	0.90	0.89	0.89	11694

NAIVE BAYES

Accuracy: 0.7961347699675047

Precision: 0.80047175694314

Recall: 0.7961347699675047

F1 Score: 0.7963218551879387

=====

Laporan Klasifikasi (Classification Report):

	precision	recall	f1-score	support
negative	0.86	0.77	0.81	3955
neutral	0.80	0.77	0.78	3850
positive	0.75	0.85	0.80	3889
accuracy			0.80	11694
macro avg	0.80	0.80	0.80	11694

Gambar 4.9 Output Rasio 50:50

#OUTPUT 40:60

RANDOM FOREST

Accuracy: 0.8857692581771539

Precision: 0.8885108998751371

Recall: 0.8857692581771539

F1 Score: 0.8854897471791793

=====

Laporan Klasifikasi (Classification Report):

	precision	recall	f1-score	support
negative	0.85	0.90	0.87	4711
neutral	0.89	0.93	0.91	4654
positive	0.93	0.82	0.87	4668
accuracy			0.89	14033
macro avg	0.89	0.89	0.89	14033
weighted avg	0.89	0.89	0.89	14033

SVM

Accuracy: 0.8804959737760992

Precision: 0.8857892892291311

Recall: 0.8804959737760992

F1 Score: 0.8786120856096464

=====

Laporan Klasifikasi (Classification Report):

	precision	recall	f1-score	support
negative	0.93	0.76	0.83	4711
neutral	0.83	0.96	0.89	4654
positive	0.90	0.93	0.91	4668
accuracy			0.88	14033
macro avg	0.89	0.88	0.88	14033
weighted avg	0.89	0.88	0.88	14033

NAIVE BAYES

Accuracy: 0.7857906363571582

Precision: 0.7913307821256907

Recall: 0.7857906363571582

F1 Score: 0.7857016966564261

=====

Laporan Klasifikasi (Classification Report):

	precision	recall	f1-score	support
negative	0.84	0.75	0.79	4711
neutral	0.80	0.75	0.77	4654
positive	0.73	0.86	0.79	4668
accuracy			0.79	14033
macro avg	0.79	0.79	0.79	14033
weighted avg	0.79	0.79	0.79	14033

Gambar 4.10 Output Rasio 40:60

```
#OUTPUT 30:70
```

```
RANDOM FOREST
```

```
Accuracy: 0.8689836305888101
```

```
Precision: 0.871680469480341
```

```
Recall: 0.8689836305888101
```

```
F1 Score: 0.8687775825792761
```

```
=====
```

```
Laporan Klasifikasi (Classification Report):
```

	precision	recall	f1-score	support
negative	0.84	0.87	0.85	5494
neutral	0.86	0.92	0.89	5433
positive	0.92	0.82	0.87	5445
accuracy			0.87	16372
macro avg	0.87	0.87	0.87	16372
weighted avg	0.87	0.87	0.87	16372

```
SVM
```

```
Accuracy: 0.8628756413388713
```

```
Precision: 0.8702487088057069
```

```
Recall: 0.8628756413388713
```

```
F1 Score: 0.8600790060162696
```

```
=====
```

```
Laporan Klasifikasi (Classification Report):
```

	precision	recall	f1-score	support
negative	0.92	0.71	0.81	5494
neutral	0.81	0.94	0.87	5433
positive	0.88	0.94	0.91	5445
accuracy			0.86	16372
macro avg	0.87	0.86	0.86	16372
weighted avg	0.87	0.86	0.86	16372

NAIVE BAYES

Accuracy: 0.7733935988272661

Precision: 0.7819532783196133

Recall: 0.7733935988272661

F1 Score: 0.7729315143559038

=====

Laporan Klasifikasi (Classification Report):

	precision	recall	f1-score	support
negative	0.83	0.73	0.78	5494
neutral	0.80	0.72	0.76	5433
positive	0.71	0.87	0.78	5445
accuracy			0.77	16372
macro avg	0.78	0.77	0.77	16372
weighted avg	0.78	0.77	0.77	16372

Gambar 4.11 Output Rasio 30:70

#OUTPUT 20:80

RRANDOM FOREST

Accuracy: 0.8371546149323927

Precision: 0.8443075957651528

Recall: 0.8371546149323927

F1 Score: 0.836975033059186

=====

Laporan Klasifikasi (Classification Report):

	precision	recall	f1-score	support
negative	0.85	0.80	0.82	6242
neutral	0.78	0.92	0.84	6221
positive	0.91	0.79	0.85	6248
accuracy			0.84	18711
macro avg	0.84	0.84	0.84	18711
weighted avg	0.84	0.84	0.84	18711

SVM

Accuracy: 0.8349633905189461
 Precision: 0.8444241645812554
 Recall: 0.8349633905189461
 F1 Score: 0.8306398195344672

=====

Laporan Klasifikasi (Classification Report):

	precision	recall	f1-score	support
negative	0.91	0.66	0.76	6242
neutral	0.77	0.91	0.84	6221
positive	0.85	0.93	0.89	6248
accuracy			0.83	18711
macro avg	0.84	0.84	0.83	18711
weighted avg	0.84	0.83	0.83	18711

NAIVE BAYES

Accuracy: 0.7624926513815403
 Precision: 0.771029321949112
 Recall: 0.7624926513815403
 F1 Score: 0.7612401123544029

=====

Laporan Klasifikasi (Classification Report):

	precision	recall	f1-score	support
negative	0.81	0.73	0.77	6242
neutral	0.80	0.68	0.74	6221
positive	0.70	0.88	0.78	6248
accuracy			0.76	18711
macro avg	0.77	0.76	0.76	18711
weighted avg	0.77	0.76	0.76	18711

Gambar 4.12 Output rasio 20:80

#OUTPUT 10:90

RANDOM FOREST

Accuracy: 0.7876009501187649

Precision: 0.7900114361428823

Recall: 0.7876009501187649

F1 Score: 0.7880427933204843

=====

Laporan Klasifikasi (Classification Report):

	precision	recall	f1-score	support
negative	0.75	0.77	0.76	7024
neutral	0.77	0.81	0.79	7036
positive	0.85	0.78	0.81	6990
accuracy			0.79	21050
macro avg	0.79	0.79	0.79	21050
weighted avg	0.79	0.79	0.79	21050

SVM

Accuracy: 0.7816627078384798

Precision: 0.7942185294846854

Recall: 0.7816627078384798

F1 Score: 0.7732555630316005

=====

Laporan Klasifikasi (Classification Report):

	precision	recall	f1-score	support
negative	0.87	0.56	0.68	7024
neutral	0.74	0.84	0.78	7036
positive	0.78	0.94	0.85	6990
accuracy			0.78	21050
macro avg	0.79	0.78	0.77	21050
weighted avg	0.79	0.78	0.77	21050

NAIVE BAYES

Accuracy: 0.6504513064133016

Precision: 0.708994659548936

Recall: 0.6504513064133016

F1 Score: 0.6412775016305057

=====

Laporan Klasifikasi (Classification Report):

	precision	recall	f1-score	support
negative	0.76	0.59	0.67	7024
neutral	0.82	0.44	0.57	7036
positive	0.54	0.92	0.68	6990
accuracy			0.65	21050
macro avg	0.71	0.65	0.64	21050
weighted avg	0.71	0.65	0.64	21050

Gambar 4.13 Output rasio 10:90

Berikut terlampir tabel ringkasan, hasil pengetesan performansi 3 model machine learning sebelum :

Tabel 4.1 Tabel Ringkasan Hasil Pengetesan Performa

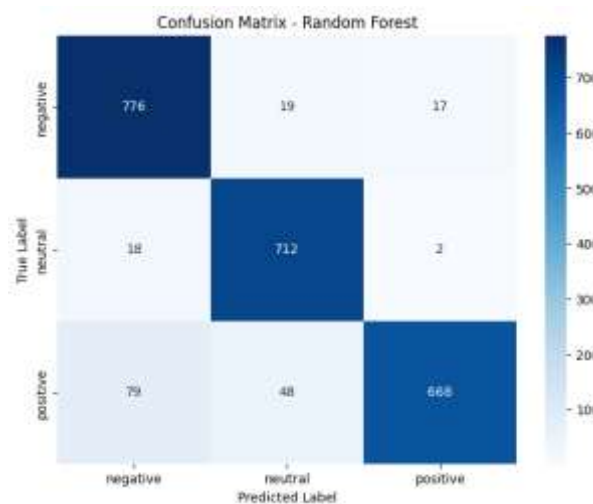
Rasio Data (Train:Test)	Model	Akurasi	Presisi	Recall	F1 Score
90 : 10	Random Forest	92.69%	92.96%	92.69%	92.66%
	SVM	92.05%	92.53%	92.05%	92.03%
	Naive Bayes	82.26%	82.41%	82.26%	82.29%
80:20	Random Forest	92.22%	92.42%	92.22%	92.16%
	SVM	91.88%	92.30%	91.88%	91.83%
	Naive Bayes	81.72%	81.84%	81.72%	81.73%
70:30	Random Forest	91.69%	91.92%	91.69%	91.64%
	SVM	91.26%	91.71%	91.26%	91.20%
	Naive Bayes	80.59%	80.85%	80.59%	80.62%
60:40	Random Forest	90.79%	91.07%	90.79%	90.73%
	SVM	90.51%	91.08%	90.51%	90.40%
	Naive Bayes	80.64%	80.94%	80.64%	80.66%
50:50	Random Forest	89.83%	90.11%	89.83%	89.78%
	SVM	89.33%	89.92%	89.33%	89.17%
	Naive Bayes	79.61%	80.05%	79.61%	79.63%
40 : 60	Random Forest	88.58%	88.85%	88.58%	88.55%
	SVM	88.05%	88.58%	88.05%	87.86%
	Naive Bayes	78.58%	79.13%	78.58%	78.57%
30 : 70	Random Forest	86.90%	87.17%	86.90%	86.88%
	SVM	86.29%	87.02%	86.29%	86.01%
	Naive Bayes	77.34%	78.20%	77.34%	77.29%
20 : 80	Random Forest	83.71%	84.43%	83.71%	83.70%
	SVM	83.50%	84.44%	83.50%	83.06%
	Naive Bayes	76.25%	77.10%	76.25%	76.12%
10 : 90	Random Forest	78.76%	79.00%	78.76%	78.80%
	SVM	78.17%	79.42%	78.17%	77.33%
	Naive Bayes	65.05%	70.90%	65.05%	64.13%

Serangkaian hasil pengujian terhadap tiga model klasifikasi Random Forest, SVM, dan Naive Bayes—dilakukan dengan menggunakan sembilan skenario rasio pembagian data (mulai dari 90:10 hingga 10:90). Ada beberapa kesimpulan utama yang dapat diambil dari hasil ini. Di seluruh pengujian, model Hutan Random dan SVM secara konsisten menunjukkan kinerja yang jauh lebih baik dibandingkan Naive Bayes. Keduanya mampu mencapai skor akurasi, presisi, dan F1 di atas 90% pada rasio data pelatihan yang besar, seperti 90:10, 80:20, dan 70:30, dengan Random Forest seringkali menunjukkan keunggulan tipis sebagai model berkinerja terbaik. Puncak Random Forest tertinggi tercatat pada rasio 90:10 dengan akurasi 0.9268, sementara puncak SVM tercatat pada rasio 80:20 dengan akurasi 0.9187.

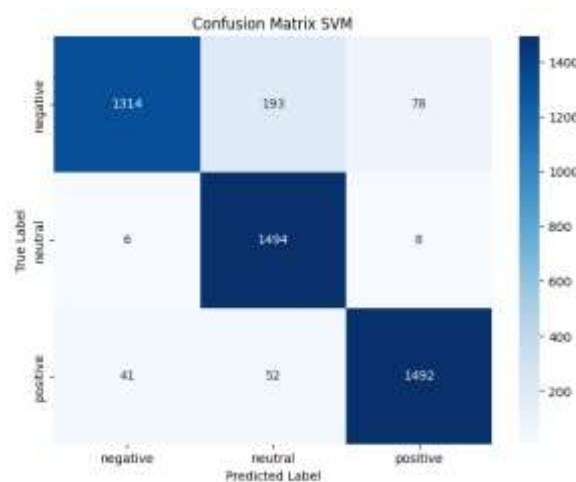
Selain itu, ditemukan bahwa ada korelasi yang kuat antara jumlah data pelatihan dan kinerja ketiga model. Semua model menurun secara sistematis seiring dengan berkurangnya

jumlah data pelatihan. Ketika rasio data pelatihan berada di bawah 60%, penurunan ini menjadi sangat signifikan. Ini menunjukkan bahwa jumlah data pelatihan yang memadai sangat penting untuk mencapai daya prediksi yang andal. Meskipun model Naive Bayes adalah yang paling ringan secara komputasional, ia selalu menghasilkan hasil yang paling tidak akurat dan menunjukkan sensitivitas yang lebih tinggi terhadap pengurangan data pembelajaran. Hasilnya menurun drastis hingga 0,6504 pada rasio 10:90.

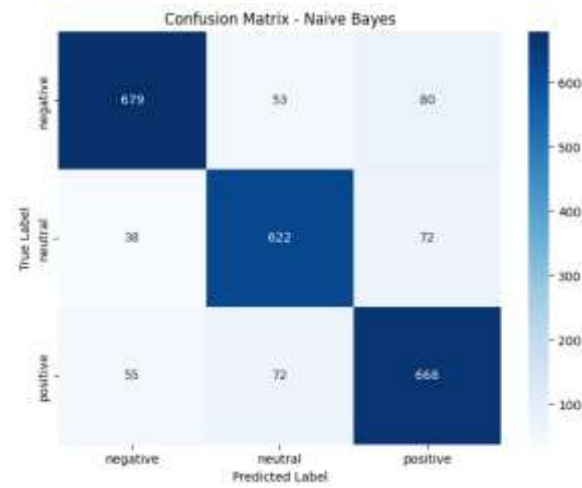
Secara keseluruhan, temuan ini menunjukkan bahwa Random Forest adalah model yang paling cocok untuk dataset yang digunakan. Selain itu, terbukti bahwa pembagian data dengan rasio 90:10 atau 80:20 memberikan keseimbangan terbaik antara data pelatihan dan pengujian, yang menghasilkan model klasifikasi yang paling akurat dan andal sesuai dengan hasil *confusion matrix* untuk pengujian rasio 90:10 :



Gambar 4.14 Confusion Matrix Random Forest 90:10

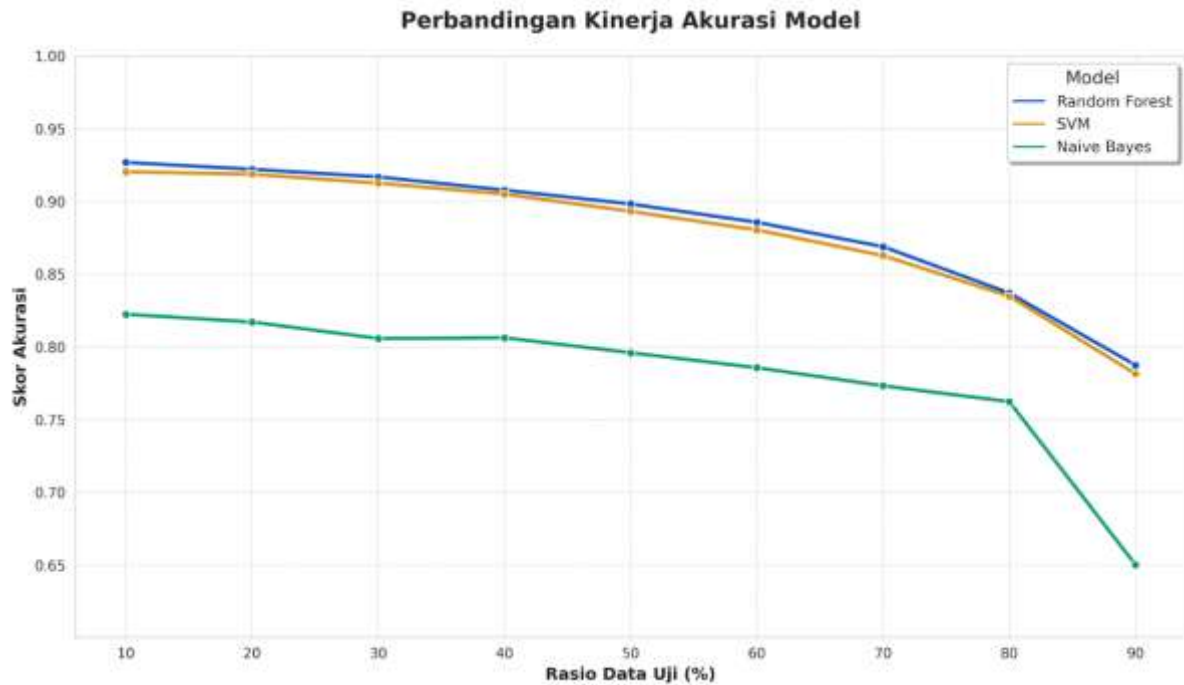


Gambar 4.15 Confusion Matrix SVM 90:10

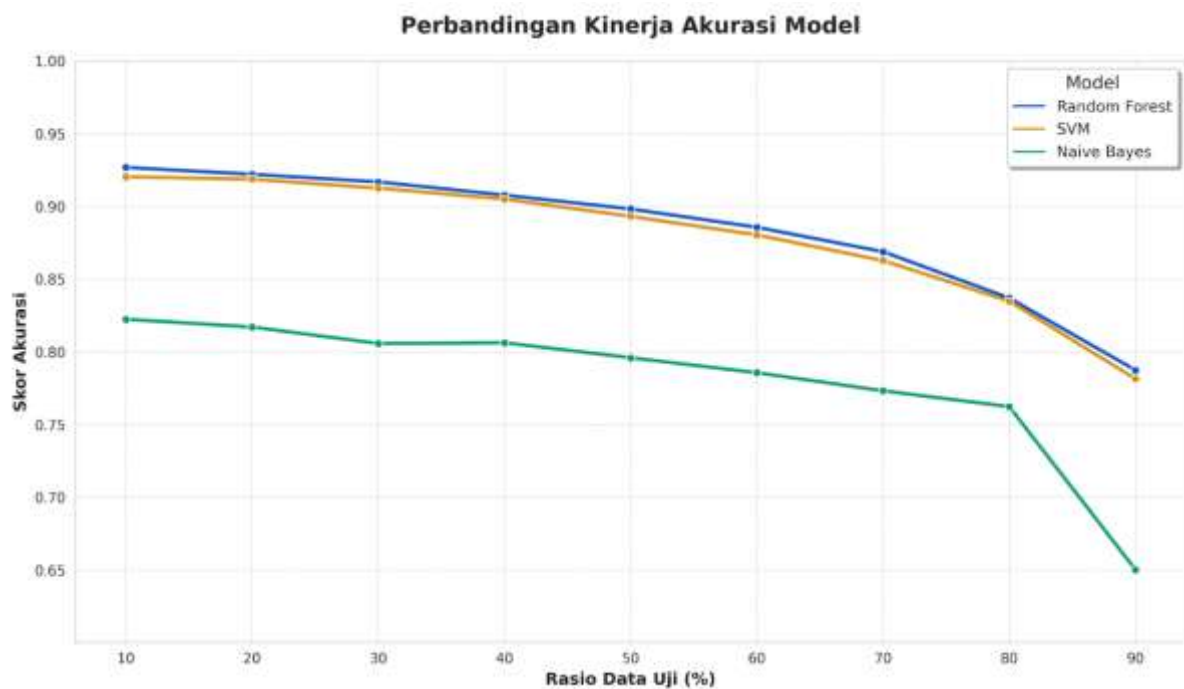


Gambar 4.16 Confusion Matrix Naive Bayes 90:10

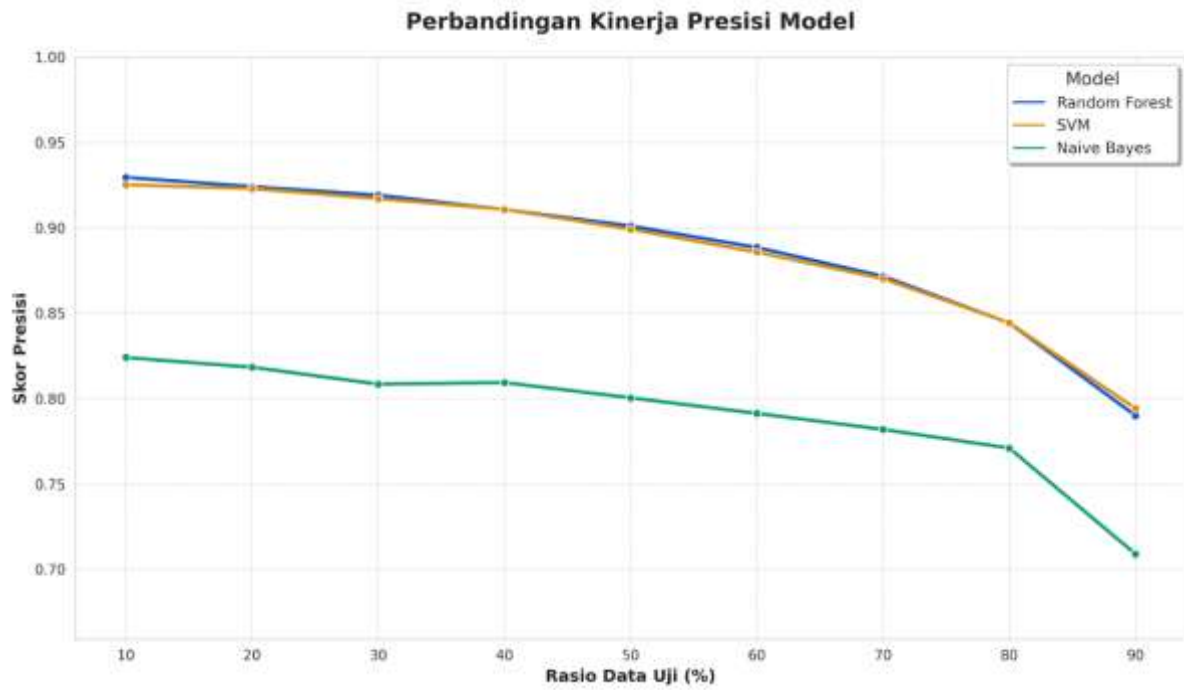
Berikut terlampir untuk visualisasi grafik performa dari parameter akurasi, presisi, recall dan skor F1.



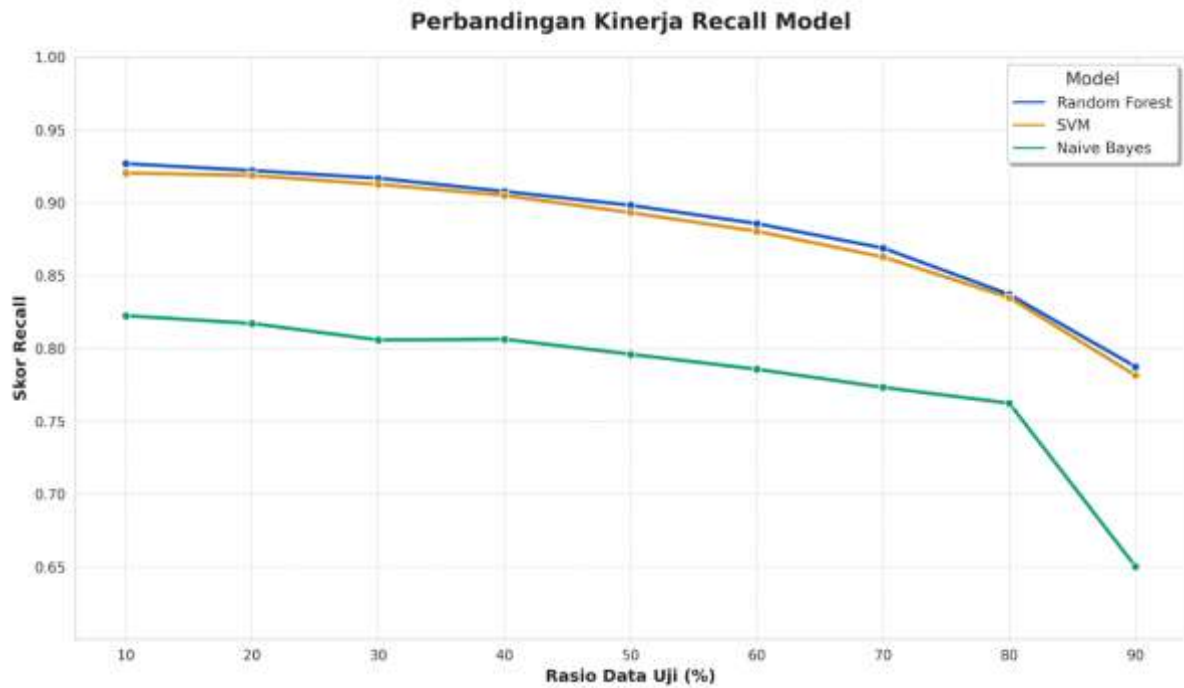
Gambar 4.17 Gambar Grafik Performa Akurasi Model



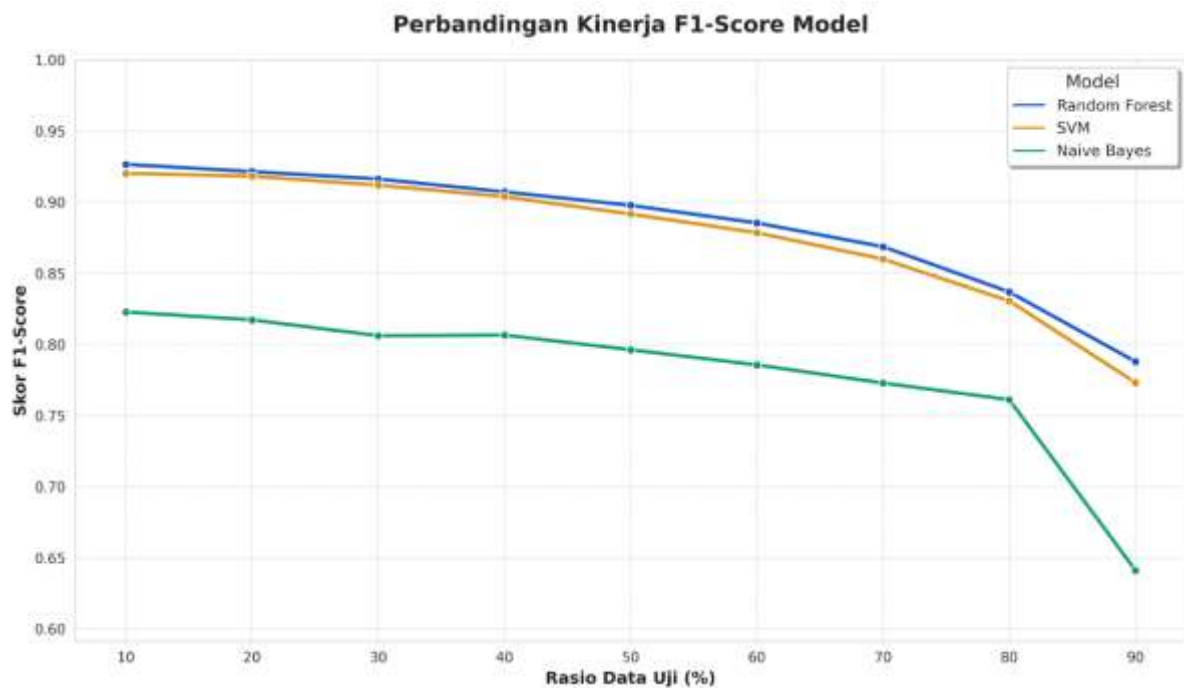
Gambar 4.18 Gambar Grafik Performa Akurasi Model



Gambar 4.19 Gambar Grafik Performa Presisi Model



Gambar 4.20 Gambar Grafik Performa *Recall* Model



Gambar 4.21 Gambar Grafik Performa Skor F1 Model

BAB V

KESIMPULAN DAN SARAN

5.1 Kesimpulan

Hasil Penelitian Model Kinerja Terbaik: Dari tiga model yang diuji, Random Forest secara konsisten menunjukkan kinerja terbaik; SVM (*Support Vector Machine*) mengikuti dengan kinerja yang sangat dekat. Kedua telah terbukti sangat efektif dalam menyelesaikan tugas klasifikasi pada kumpulan data ini, terutama dalam kasus di mana proporsi data yang dilatih sangat besar.

Untuk mengevaluasi kinerja model klasifikasi, penggunaan Weighted average diperlukan untuk memberikan pengukuran yang lebih representatif dan adil. Ini terutama berlaku untuk mengevaluasi kemungkinan adanya ketidakseimbangan kelas (*class imbalance*) pada *dataset* pengujian. Metode ini menghitung skor *precision*, *recall*, dan *F1* rata-rata untuk setiap kelas, dengan bobot setiap skor secara proporsional berdasarkan jumlah sampel (*support*) yang ada dalam kelas tersebut. Metode ini secara efektif memberikan prioritas pada performa model di kelas mayoritas tanpa mengabaikan kontribusi kelas minoritas, sehingga menghasilkan skor total yang lebih akurat yang menunjukkan kemampuan model untuk memprediksi distribusi data yang sebenarnya. Weighted average menawarkan penilaian yang lebih menyeluruh dan dapat diandalkan, berbeda dengan macro average, yang memperlakukan setiap kelas secara setara dan dapat memberikan gambaran yang terlalu optimis ketika model berkinerja buruk pada kelas minoritas. Karena itu, menggunakannya dalam laporan klasifikasi sangat penting untuk memastikan model yang objektif, terutama setelah menggunakan teknik penyeimbangan seperti SMOTE pada data latih karena set data uji mempertahankan distribusi kelas asli.

Model dengan Kinerja Terendah: Di setiap skenario pengujian, model Naive Bayes secara konsisten memberikan hasil yang paling rendah dari segi akurasi, presisi, recall, dan skor F1, Model ini bekerja dengan memperlakukan setiap nilai TF-IDF sebagai entitas probabilistik yang terisolasi. Ini tidak sesuai dengan karakteristik bahasa alami, di mana dependensi antar kata sangat penting untuk menentukan sentimen. Kelemahan utama adalah ketidakmampuan untuk memahami relasi semantik ini. Namun, efek yang paling signifikan berasal dari intervensi teknik oversampling SMOTE. Teknik ini secara artifisial menghasilkan data dengan tingkat korelasi fitur lokal yang tinggi dengan menghasilkan sampel sintetis yang didasarkan pada interpolasi antara tetangga terdekat. Prasyarat matematis dari algoritma Naive Bayes secara langsung dilanggar dengan pengenalan data yang fiturnya tidak independen.

Akibatnya, validitas statistik model terganggu, yang mengakibatkan ketidakmampuan untuk membuat batas keputusan yang efektif dan akurat. Pada akhirnya, ini menghasilkan skor performa yang jauh lebih rendah dibandingkan dengan model lain yang tidak terikat oleh asumsi serupa.

Pengaruh Rasio Data: Jumlah data latih sangat memengaruhi kinerja semua model. Rasio data latih yang lebih tinggi, seperti 90:10 dan 80:20, menghasilkan kinerja yang jauh lebih baik. Sebaliknya, ketika rasio data latih lebih rendah, seperti 20:80 atau 10:90, kinerja semua model menurun secara signifikan.

Studi ini merekomendasikan penggunaan model Random Forest dengan pembagian data latih sebesar 80% hingga 90% dari seluruh dataset untuk mendapatkan hasil klasifikasi yang paling akurat dan andal.

Terkait analisis potensi overfitting pada model Random Forest dan SVM, kedua model memiliki kemampuan untuk di generalisasi dalam konteks penelitian ini. Mekanisme ensemble di Random Forest, yang mengagregasi prediksi dari beberapa pohon keputusan yang dilatih pada subset data dan fitur yang acak, secara inheren mengurangi risiko overfitting dengan mengurangi varians model. Dengan cara yang sama, kompleksitas model di minimalisasi dengan memilih *Linear Support Vector Classifier* (LinearSVC) untuk implementasi SVM. Dengan menggunakan hyperplane pemisah linear, model dipaksa untuk menemukan pola yang lebih umum daripada membuat batas keputusan yang terlalu kompleks yang hanya menyesuaikan diri dengan data latihan. Hasil eksperimen mendukung kesimpulan ini: kedua model menunjukkan penurunan performa yang wajar dan konsisten di berbagai rasio pembagian data. Alih-alih hanya menghafal data latih, stabilitas performa ini menunjukkan bahwa model berhasil mempelajari pola yang mendasari data dan mampu mengaplikasikannya pada data yang belum pernah terlihat sebelumnya.

Adapun dari hasil penelitian dapat disimpulkan pada proses labeling menggunakan model *RoBERTa* sebelum dilakukan proses *handling imbalance data*, bahwa berdasarkan sampling dari 9990 data ulasan pengguna aplikasi edukasi anak berbahasa Indonesia cenderung bermuatan positif

Proses penyesuaian yang berpusat pada tahap *preprocessing* teks diperlukan untuk menyesuaikan dan menguji validitas kerangka kerja penelitian ini pada berbagai domain linguistik. Metodologi evaluasi komparatif seperti model Naive Bayes, SVM, dan Random Forest dapat digunakan secara langsung, tetapi dengan sedikit perubahan pada bagian leksikal bahasa target. Secara khusus, daftar *stopwords* Bahasa Inggris yang relevan harus digunakan

sebagai pengganti kamus *stopwords* Bahasa Indonesia yang digunakan dalam penelitian ini. Dengan cara yang sama, algoritma stemming saat ini harus diganti dengan yang menggunakan kaidah morfologi Bahasa Inggris, seperti algoritma *Porter* atau *Snowball*. Karena representasi fitur melalui TF-IDF sangat bergantung pada keberhasilan normalisasi teks, proses adaptasi ini sangat penting. Jika sumber daya linguistik ini tidak digunakan, model tidak akan dapat mengenali kata-kata dasar yang sama dan akan terdistorsi oleh kata-kata umum yang tidak berinformasi. Pada akhirnya, ini akan menghasilkan representasi fitur yang tidak akurat dan kesimpulan bahwa kinerja model tidak sah.

Implementasi model klasifikasi sentimen yang telah tervalidasi, seperti Naïve Bayes, Random Forest dan SVM analisis ini, memberikan nilai praktis yang signifikan bagi siklus pengembangan aplikasi karena dapat mengubah data kualitatif dari umpan balik pengguna menjadi metrik kuantitatif yang dapat ditindaklanjuti. Hasil emosi ini dapat dimasukkan ke dalam berbagai fungsi operasional secara strategis. Pertama, dalam iterasi produk dan peningkatan pengalaman pengguna, analisis sentimen dapat secara otomatis mengidentifikasi keluhan pengguna dari ulasan dengan sentimen negatif. Ini memungkinkan tim pengembang untuk memprioritaskan perbaikan bug dan penyempurnaan fitur yang paling mendesak. Kedua, dalam optimalisasi dukungan pelanggan, sentimen dapat digunakan sebagai mekanisme triase otomatis untuk menyebarluaskan tiket pengguna yang sangat tidak puas, sehingga meningkatkan pengalaman pengguna. Terakhir, pada tingkat intelijen bisnis, agregasi data sentimen dapat digunakan untuk analisis kompetitif, misalnya untuk mengetahui seberapa baik persepsi publik terhadap aplikasi pesaing dan seberapa baik kampanye pemasaran atau peluncuran fitur baru. Oleh karena itu, model sentimen yang akurat menghubungkan masukan pengguna dengan pengambilan keputusan berbasis data. Ini memungkinkan pengembang menggunakan metode yang lebih cepat dan sesuai dengan kebutuhan pasar.

pengguna, seperti memberikan rating, dan topik yang paling sering menjadi subjek umpan balik, seperti permintaan fitur dan keberadaan iklan. Oleh karena itu, dari sudut pandang netral, informasi ini memberikan gambaran praktis tentang apa aplikasi itu dan bagaimana pengguna berinteraksi dengannya secara teknis dan fungsional. Mereka juga memetakan titik-titik interaksi utama dan topik diskusi paling umum tentangnya.

5.3 Saran

Dari Penelitian ini dapat dijadikan sebagai titik tolak untuk penelitian lanjutan yang berfokus pada tiga garis utama. Untuk melihat potensi peningkatan dalam menangkap dependensi kontekstual yang kompleks, pertama-tama dapat di evaluasi implementasi model deep learning seperti arsitektur Transformer atau LSTM. Kedua, studi lanjutan dapat membandingkan secara sistematis berbagai strategi penanganan ketidakseimbangan kelas selain SMOTE (seperti pengambilan sampel rendah atau pembelajaran yang sensitif terhadap biaya) untuk menemukan yang paling kuat dan kompatibel dengan berbagai algoritma. Terakhir, menggunakan metode Explainable AI (XAI), seperti SHAP atau LIME, pada model berkinerja tinggi dapat memberikan wawasan interpretatif yang lebih dalam tentang faktor pendorong keputusan model, lebih dari sekadar metrik akurasi prediktif.

DAFTAR PUSTAKA

- Agustyaningrum, C. I., Gata, W., Nurfalah, R., Radiyah, U., & Maulidah, M. (2020). Komparasi Algoritma Naive Bayes, Random Forest dan SVM untuk Memprediksi Niat Pembelanja Online. *Jurnal Informatika*, 20(2), 164–173. <https://doi.org/10.30873/JI.V20I2.2402>
- Ali, H., & Hendrastuty, N. (2024). Comparison of Naïve Bayes Classifier, Support Vector Machine, Random Forest Algorithms for Public Sentiment Analysis of KIP-K Program on Twitter. *Jurnal Teknik Informatika (Jutif)*, 5(6), 1701–1712. <https://doi.org/10.52436/1.JUTIF.2024.5.6.4030>
- Ramadhan, G. R., & Sugianto, C. A. (2024). Analisis Sentimen Ulasan Aplikasi Dana di Google Play Store Menggunakan Algoritma Naïve Bayes. *JATI (Jurnal Mahasiswa Teknik Informatika)*, 8(5), 9849–9857. <https://doi.org/10.36040/jati.v8i5.10732>
- Ferreira-Mello, R., André, M., Pinheiro, A., Costa, E., & Romero, C. (2019). Text mining in education. *WIREs Data Mining and Knowledge Discovery*, 9(6). <https://doi.org/10.1002/widm.1332>
- Pookduang, P., Klangbunrueang, R., Chansanam, W., & Lunrasri, T. (2025). Advancing Sentiment Analysis: Evaluating RoBERTa against Traditional and Deep Learning Models. *Engineering, Technology & Applied Science Research*, 15(1), 20167–20174. <https://doi.org/10.48084/etasr.9703>
- Cahyono, N., & Praneswara, A. O. (2023). Analisis Sentimen Ulasan Aplikasi TikTok Shop Seller Center di Google Playstore Menggunakan Algoritma Naive Bayes. *Indonesian Journal of Computer Science*, 12(6). <https://doi.org/10.33022/ijcs.v12i6.3473>
- Khairunnisa, A., Notodiputro, K. A., & Sartono, B. (2024). A Comparative Study of Random Forest and Double Random Forest Models from View Points of Their Interpretability. *Scientific Journal of Informatics*, 11(1), 207–218. <https://doi.org/10.15294/sji.v11i1.48721>
- Lowell, A., Lowell, A., Candra, K., Indra, E., & Informasi, S. (2025). Perbandingan Metode Support Vector Machine (SVM) dan Naive Bayes pada Analisis Sentimen Ulasan Aplikasi OVO. *Jurnal Media Informatika*. <https://doi.org/10.55338/jumin.v6i2.5134>
- Megira, S., & Kusriani, K. (2024). Perbandingan Kinerja Naive Bayes dan Support Vector Machine untuk Prediksi Herregistrasi. *ejournal.uniled.ac.id*, 8(3). <https://doi.org/10.54342/qjc4j706>

- Ningrum, M. P., & Mutia, R. (2025). Sentiment Analysis of Twitter Reviews on Google Play Store Using a Combination of Convolutional Neural Network and Long Short-Term Memory Algorithms. *Research Journal of Engineering, Data Technology and Computer*, 2025, 2(2), 107–115. <https://doi.org/10.57152/predatecs.v2i2.1625>
- Nufairi, F., Pratiwi, N., & Herlando, F. (2024). Analisis Sentimen pada Ulasan Aplikasi Threads di Google Play Store Menggunakan Algoritma Support Vector Machine. *JIPPI (Jurnal Ilmiah Penelitian dan Pembelajaran Informatika)*, 9(1), 339–348. <https://doi.org/10.29100/jipi.v9i1.4929>
- Nurian, A. (2023). Analisis Sentimen Ulasan Pengguna Aplikasi Google Play Menggunakan Naïve Bayes. *Jurnal Informatika dan Teknik Elektro Terapan*, 11(3s1), 2830–7062. <https://doi.org/10.23960/JITET.V11I3S1.3348>
- Purnomo, D., Firgiawan, W., & Nur, N. (2025). Komparasi Algoritma Random Forest, Naïve Bayes, dan SVM pada Sentimen Kebijakan PPN 12%. *Jurnal Tekno Kompak*, 19(2), 155–167. <https://doi.org/10.33365/jtk.v19i2.122>
- Putri, R. C. (2024). Analisis sentimen provider by. u dengan metode naïve bayes classifier dan k-nearest neighbor (KNN). <https://repository.uinjkt.ac.id/dspace/handle/123456789/76502>
- Rivaldi, R. C., & Wismarini, T. D. (2024). Analisis Sentimen Pada Ulasan Produk Dengan Metode Natural Language Processing (NLP). *Elkom: Jurnal Elektronika dan Komputer*, 17(1), 120–128. <https://doi.org/10.51903/elkom.v17i1.1680>
- Safrudin, M., Martanto, M., & Hayati, U. (2024). Perbandingan Kinerja Naïve Bayes dan Support Vector Machine untuk Klasifikasi Sentimen Ulasan Game Genshin Impact. *JATI (Jurnal Mahasiswa Teknik Informatika)*, 8(3), 3182–3188. <https://doi.org/10.36040/jati.v8i3.8415>
- Salsabila, A., & Muntahanah, M. (2024). Analisis Sentimen Ulasan Aplikasi ChatGPT Pada Google Play Menggunakan Metode Support Vector Machine. *Progresif: Jurnal Ilmiah Komputer*, 20(2), 714. <https://doi.org/10.35889/progresif.v20i2.2074>
- Putri, D. E., & Mandala, E. P. (2021). Pemanfaatan Aplikasi Pendidikan Anak sebagai Penunjang Pembelajaran. *jmi-upiypk.org*, 02(28). <https://doi.org/https://doi.org/10.35134/jmi.v28i2.69>
- Vitianingsih, A. V. (2016). *Game Edukasi Sebagai Media Pembelajaran Pendidikan Anak Usia Dini*. <http://repository.unitomo.ac.id/id/eprint/75>

Wongso, W. (2023). *indonesian-roberta-base-sentiment-classifier* (Revision e402e46).
Hugging Face. <https://doi.org/10.57967/hf/0644>

LAMPIRAN