

Analisis sentimen aplikasi satu sehat menggunakan metode naive bayes dan particle swarm optimization

by 09021381924131 Reza Qumaini

Submission date: 29-Jul-2024 03:21 PM (UTC+0700)

Submission ID: 2424263754

File name: ENGGUNAKAN_METODE_NAIVE_BAYES_DAN_PSO_UPDATE_-_Reza_Qumaini.docx (1.22M)

Word count: 11169

Character count: 67119

*ANALISIS SENTIMEN APLIKASI SATUSEHAT MENGGUNAKAN
METODE NAIVE BAYES DAN PARTICLE SWARM OPTIMIZATION*

Diajukan sebagai syarat untuk menyelesaikan
pendidikan program strata-1 pada jurusan teknik informatika



Oleh :

Reza Qumaini
NIM : 09021381924131

JURUSAN TEKNIK INFORMATIKA FAKULTAS ILMU
KOMPUTER UNIVERSITAS SERIWIJAYA

2024

I-1

BAB I

PENDAHULUAN

1.1. Pendahuluan

Pendahuluan Pada bab yang berfungsi sebagai pengantar, pokok-pokok pikiran utama yang menjadi landasan bagi realisasi tesis dibahas. hal-hal utama yang menjadi pokok pemikiran meliputi landasan, masalah penelitian, perincian masalah/isu penelitian, target penelitian, dan manfaat penelitian..

1.2. Latar Belakang

Dalam era digital yang semakin pesat perkembangannya, teknologi informasi memiliki peran yang sangat penting dalam berbagai sektor, termasuk kesehatan. Aplikasi Satu Sehat adalah salah satu aplikasi kesehatan yang menyediakan informasi dan layanan kesehatan kepada penggunanya. Dalam pengembangan aplikasi semacam ini, analisis sentimen menjadi sangat penting untuk memahami respons pengguna terhadap aplikasi tersebut. Analisis sentimen adalah proses yang melibatkan teknik-teknik pemrosesan bahasa alami dan pembelajaran mesin untuk mengidentifikasi dan mengkategorikan opini yang diekspresikan dalam teks. Penelitian ini menggunakan metode Naive Bayes dan Particle Swarm Optimization (PSO) untuk meningkatkan akurasi klasifikasi sentimen pengguna terhadap aplikasi Satu Sehat. Dengan metode ini, penelitian ini diharapkan dapat memberikan kontribusi signifikan dalam pengembangan aplikasi yang lebih responsif terhadap kebutuhan pengguna.

Menangani informasi menjadi data yang bermanfaat membutuhkan teknik dan perhitungan yang berbeda dalam bidang perancangan informatika. Teknik yang sering digunakan dalam analisis sentimen dan pemrosesan bahasa alami termasuk algoritma optimasi seperti Particle

Swarm Optimization (PSO) dan algoritma klasifikasi seperti Naive Bayes. Dengan pendekatan probabilistiknya, Naive Bayes membuat klasifikasi teks menjadi efisien dan efektif. Sementara itu, PSO membantu melacak batas-batas ideal untuk mengerjakan presentasi model pengaturan. Dalam ulasan ini, gabungan dari kedua strategi tersebut diterapkan untuk mengkarakterisasi perasaan klien terhadap aplikasi Satu Sehat. Kebutuhan akan akurasi yang tinggi pada hasil klasifikasi dan karakteristik data yang digunakan mempengaruhi pemilihan metode ini.

Penelitian Puspita dan Widodo pada tahun 2021 menunjukkan bahwa metode Naive Bayes memiliki tingkat akurasi sebesar 89,14 persen. Meskipun masih lebih rendah dari metode Decision Tree yang mencapai akurasi 96,13 persen, hal ini menunjukkan bahwa Naive Bayes memiliki tingkat akurasi yang cukup baik. Algoritma Naive Bayes Classifier memiliki tingkat akurasi sebesar 62%, menurut penelitian lain oleh Karim (2021). Hal ini mungkin disebabkan karena penelitian tersebut hanya menggunakan 250 data media sosial Instagram, sehingga kurang akurat dibandingkan dengan penelitian yang menggunakan lebih banyak data.

Menurut penelitian Sari (2019), akurasi algoritma Naive Bayes dapat ditingkatkan secara signifikan dengan menggunakan algoritma Particle Swarm Optimization. Algoritma Naive Bayes sendiri menghasilkan akurasi 74,34%, mewakili peningkatan akurasi sebesar 8,11 persen, sedangkan gabungan metode Naive Bayes dan Particle Swarm Optimization mencapai akurasi 82,45%.

Salah satu kesimpulan yang dapat diambil dari penelitian sebelumnya adalah bahwa Naive Bayes merupakan salah satu algoritma yang paling efisien untuk analisis sentimen. Bagaimanapun, ada beberapa faktor yang dapat mempengaruhi ketepatan perhitungan, misalnya, berapa banyak data yang dicoba dan metode optimasi yang digunakan. Oleh karena itu, algoritma Naive Bayes dalam analisis sentimen dioptimasi dalam penelitian ini dengan menggunakan algoritma Particle Swarm Optimization

dengan harapan dapat meningkatkan nilai akurasi yang tinggi sehingga dapat digunakan sebagai tolak ukur kepuasan masyarakat terhadap suatu pelayanan.

1.3. Rumusan Masalah

Berdasarkan penjelasan dari identifikasi masalah penelitian ini, maka permasalahan yang diharap dalam menyelesaikan penelitian ini adalah :

1. Bagaimana menerapkan Naïve Bayes dan *Particle Swarm Optimization* pada Analisis Sentimen?
2. Berapa tingkat akurasi algoritma Naïve Bayes dengan menggunakan optimisasi *Particle Swarm Optimization* dalam mengklasifikasikan analisis sentimen.
3. Bagaimana pengaruh metode *Particle Swarm Optimizaton* terhadap kinerja algoritma Naïve Bayes dalam mengklasifikasikan sentimen aplikasi Satu Sehat?

1.4. Tujuan Penelitian

Tujuan pada penelitian ini adalah:

1. Menerapkan algoritma Naïve Bayes dan *Particle Swarm Optimization* dalam klasifikasi teks pada analisis sentimen.
2. Mengukur dan mengetahui nilai akurasi pada model jika diterapkan metode *Particle Swarm Optimization* pada algoritma Naïve Bayes.
3. Mengevaluasi dan memahami pengaruh metode *Particle Swarm Optimization* terhadap kinerja algoritma *Naïve Bayes* dalam mengklasifikasikan sentimen Aplikasi Satu Sehat.

4. Manfaat Penelitian

Manfaat dari penelitian yang akan dilakukan ini antara lain:

1. Dapat memahami dan mengaplikasikan metode Naïve Bayes pada pengklasifikasian teks pada analisis sentimen
2. Dapat memahami kegunaan metode *Particle Swarm Optimization*

sebagai optimasi dari metode naive bayes dalam analisis sentimen

3. Memberikan fasilitas kepada masyarakat untuk mengetahui sentimen dari suatu kalimat teks. Sehingga dapat dilakukan analisis reputasi pada suatu pelayanan

4. Batasan Masalah

Pembatasan masalah yang menjadi ruang lingkup penelitian diperlukan mengingat latar belakang dan identifikasi masalah yang telah disebutkan di atas dan menghindari penyimpangan dari ruang lingkup masalah. Oleh karena itu, berikut ini akan dibahas prosedur-prosedur yang akan dilakukan:

1. Data yang digunakan sebanyak 1000 data yang menggunakan bahasa Indonesia.
2. Data akan di klasifikasikan ke dalam kelas positif dan Negatif

5. Sistematika Penulisan

Sistematika penulisan skripsi ini adalah sebagai berikut :

BAB I. PENDAHULUAN

Latar belakang, rumusan masalah, tujuan dan manfaat penelitian, batasan masalah/ruang lingkup, metodologi penelitian, dan sistematika penulisan semuanya tercakup dalam bab ini.

BAB II. KAJIAN LITERATUR

Landasan teori penelitian, seperti definisi analisis sentimen, text mining, klasifikasi teks, algoritma naive bayes, dan algoritma particle swarm optimization, akan dibahas dalam bab ini.

BAB III. METODOLOGI PENELITIAN

Tahapan-tahapan penelitian ini akan dibahas secara rinci dalam bab ini. Kerangka kerja digunakan untuk memberikan penjelasan rinci tentang setiap rencana tahap penelitian.

BAB IV. PENGEMBANGAN PERANGKAT LUNAK

Tahapan pengembangan perangkat lunak penelitian ini akan dibahas dalam bab ini. Rencana untuk setiap tahap penelitian.

BAB V. HASIL DAN ANALISIS PENELITIAN

Bab ini akan memberikan penjelasan mengenai temuan-temuan studi serta analisis terhadap temuan-temuan tersebut.

BAB VI. KESIMPULAN DAN SARAN

Temuan-temuan penelitian akan dirangkum dalam bab ini, bersama dengan saran untuk penelitian di masa depan.

BAB II

KAJIAN LITERATUR

2.1. Pendahuluan

Bab ini menjelaskan konsep-konsep yang menjadi dasar penelitian ini. Konsep-konsep tersebut mencakup analisis sentimen, Aplikasi SatuSehat, platform media Play Store, klasifikasi teks, serta algoritma Naive Bayes dan Particle Swarm Optimization (PSO). Selain itu, bab ini juga membahas penelitian terkait yang relevan dengan penelitian yang akan dilakukan. Terakhir, disajikan kesimpulan yang dapat ditarik dari penelitian-penelitian tersebut.

2.2. Analisis Sentimen

Analisis sentimen adalah proses menganalisis data, opini, dan persepsi yang berasal dari sebuah teks yang digunakan oleh seseorang. Dari sudut pandang aplikasi Satu Sehat, analisis yang disajikan di sini dapat membantu menentukan siapa yang bertanggung jawab dan siapa yang diuntungkan dari kualitas aplikasi tersebut. Di zaman yang semakin terkomputerisasi, aplikasi kesehatan menjadi salah satu tren yang diminati oleh masyarakat. Aplikasi ini memberikan informasi dan layanan terkait kesehatan kepada penggunanya, sehingga dipercaya bahwa setiap orang dapat memperoleh data kesehatan yang tepat dan dapat diandalkan.

Metode Naive Bayes digunakan untuk mengklasifikasikan sentimen positif dan negatif dari teks yang dihasilkan oleh pengguna. Penelitian yang dilakukan oleh Puspita dan Widodo (2021) menunjukkan bahwa metode Naive Bayes memiliki tingkat akurasi sebesar 89,14%. Namun, penelitian lain yang dilakukan oleh Karim (2021) menunjukkan tingkat akurasi yang lebih rendah, yaitu 62%, kemungkinan disebabkan oleh minimnya data yang digunakan.

Selain itu, algoritma Particle Swarm Optimization telah terbukti dapat meningkatkan akurasi dari kinerja algoritma Naive Bayes. Penelitian yang dilakukan oleh Sari (2019) menunjukkan bahwa dengan menggunakan metode Naive Bayes dan Particle Swarm Optimization, akurasi analisis sentimen dapat mencapai 82,45%, meningkat sebesar 8,11% dibandingkan dengan penggunaan Naive Bayes saja.

Berdasarkan beberapa penelitian terdahulu, Naive Bayes merupakan algoritma yang efektif untuk analisis sentimen. Namun, faktor seperti jumlah data yang diuji dan metode optimisasi yang digunakan dapat mempengaruhi tingkat akurasi. Oleh karena itu, penelitian ini menggunakan algoritma Particle Swarm Optimization untuk mengoptimalkan algoritma Naive Bayes dalam analisis sentimen aplikasi Satu Sehat. Dengan harapan, dapat meningkatkan nilai akurasi yang tinggi sehingga menjadi tolak ukur tingkat kepuasan masyarakat terhadap suatu layanan yang diberikan.

2.3. Text Preprocessing

Text preprocessing adalah tahap untuk mempersiapkan teks menjadi data yang siap diolah pada tahap berikutnya. Input awal dari proses ini berupa dokumen. Text preprocessing dalam penelitian ini melibatkan beberapa langkah, yaitu: proses pemecahan kalimat, proses case folding, proses tokenisasi kata, proses penyaringan, dan proses stemming

2.3.1. Pemecah Kalimat

Memecah dokumen menjadi kalimat-kalimat merupakan langkah awal dalam tahap text preprocessing. Proses pemecahan kalimat ini adalah memecah string teks dokumen yang panjang menjadi kumpulan kalimat-kalimat. Dalam penelitian ini, pemecahan dokumen menjadi kalimat-kalimat dilakukan menggunakan fungsi split(), dengan tanda titik ".", tanda

tanya "?" dan tanda seru "!" sebagai delimiter untuk memotong string dokumen.

Tabel II-1. Contoh Pemecahan Kalimat

Kalimat	Hasil Pemecahan Kalimat
Saat sangat dibutuhkan ga bsa login. Sekarang apa" harus pakai sertifikat vaksin tapi susah untuk unduh dan login. Padahal ini aplikasi pemerintah, buatan pemerintah, dan syarat dimana mana, taoui kenapa gni sih	Saat sangat dibutuhkan ga bsa login Sekarang apa harus pakai sertifikat vaksin tapi susah untuk unduh dan login Padahal ini aplikasi pemerintah, buatan pemerinta dan syarat dimana mana taoui kenapa gni sih

2.3.2. Case Folding

Case folding adalah tahapan proses mengubah semua huruf dalam teks dokumen menjadi huruf kecil, serta menghilangkan karakter selain a-z.

Tabel II-2. Contoh *Case Folding*

Kalimat	Hasil <i>Case Folding</i>
Satu sehat memang sederhana dan menyehatkan masyarakat Indonesia...Go Indonesia sehat	satu sehat memang sederhana dan menyehatkan masyarakat indonesia go indonesia sehat

2.3.3. Tokenizing

Tokenizing adalah proses pemotongan string *input* berdasarkan tiap kata yang menyusunnya. Pemecahan kalimat menjadi kata-kata tunggal dilakukan dengan men-scan kalimat dengan pemisah (*delimiter*) *white space* (spasi, *tab*, dan *newline*).

Tabel II-3. Contoh *Tokenizing*

Kalimat	Hasil <i>Tokenizing</i>
Mau download sertifikat vaksin aja gak bisa, gimana sih developer nya di update malah gajelas mending yg masih pedulilindungi	[mau], [download], [sertifikat], [vaksin], [aja], [gak], [bisa], [gimana], [sih], [developer], [nya], [di], [update], [malah gajelas], [mending], [yg], [masih], [pedulilindungi]

2.3.4. Filtering

Filtering adalah proses eliminasi atau penghilangan stopwords. Stopwords merupakan kata-kata yang sering muncul dalam dokumen tetapi tidak memiliki makna deskriptif dan tidak terkait dengan topik tertentu. Dalam konteks bahasa Indonesia, stopwords dapat dianggap sebagai kata-kata yang tidak penting, seperti "di", "oleh", "pada", "sebuah", "karena", dan lain-lain.

Tabel II-4. Contoh *Filtering*

Kalimat	Hasil <i>Filtering</i>
buat pengembang, terus semangat dalam membuat aplikasi bagi Indonesia yang lebih sehat. Salam satu sehat. Indonesia sehat , Indonesia kuat.	pengembang semangat membuat aplikasi Indonesia sehat. Salam satu sehat. Indonesia sehat , Indonesia kuat.

2.3.5. Stemming

Stemming adalah proses mencari akar (root) kata dari setiap token kata dengan mengembalikan kata berimbuhan ke bentuk dasarnya (stem). Dalam penelitian ini, digunakan Porter stemming untuk bahasa Indonesia. Proses stemming bahasa Indonesia menggunakan algoritma Porter Stemmer yang terdiri dari lima tahap aturan. Aturan-aturan tersebut adalah:

1. Penanganan partikel infleksional seperti "lah", "tah", dan "kah".
Misalnya: duduklah, makanlah.
2. Penanganan kata ganti infleksional seperti "ku", "mu", dan "nya".
Contohnya: sepedaku, bukunya.
3. Pengolahan awalan derivasi pertama seperti "meng-" dan variasinya, "peng-" dan variasinya, "di-", "ter-", dan "ke-". Misalnya: membakar, mengukur, kekasih.
4. Penanganan awalan derivasi kedua seperti "ber-" dan variasinya, serta "per-" dan variasinya. Contohnya: berlari, belajar, pergi.
5. Penanganan sufiks derivasi seperti "-kan", "-an", dan "-i". Misalnya: mendapatkan, membuat janji, menemukannya.

Meskipun terdapat lima tahap aturan, kompleksitas struktur morfologis bahasa Indonesia membuat aturan-aturan ini tidak cukup untuk menangani semua kasus dalam proses stemming bahasa Indonesia. Salah satu tantangan adalah membedakan kata-kata yang mengandung afiks, termasuk awalan dan akhiran, dari kata dasar yang terdiri dari lebih dari dua suku kata.

Tabel II-5. Contoh *Stemming*

Kalimat	Hasil <i>Stemming</i>
Ingin berkata kasar sm yg buat aplikasi, dulu msh pedulilindungi sertifikat vaksin bisa ditampilkan, begitu update saturehat malah hilang permintaan selalu ditolak. Bukannya malah bagus tp jadi busuukkkkk ini apk	Ingin kata kasar sm yg buat aplikasi, dulu msh pedulilindungi sertifikat vaksin bisa tampil, begitu update saturehat malah hilang minta selalu ditolak. bukan malah bagus tp jadi busuukkkkk ini apk

2.4. Penelitian terdahulu

Penelitian terdahulu menjadi acuan bagi penulis dalam melakukan penelitian sehingga dapat memperkaya teori yang digunakan dalam mengkaji penelitian yang dilakukan. Dari penelitian terdahulu, penulis tidak menemukan penelitian dengan judul yang sama seperti penelitian ini. Namun, penulis mengangkat beberapa penelitian sebagai referensi untuk

memperkaya bahan kajian dalam penelitian ini. Bagian ini memuat uraian secara sistematis tentang hasil penelitian terdahulu yang relevan dengan masalah yang dikaji. Hasil-hasil penelitian terdahulu antara lain:

Puji Astuti dan Nuzuliarini Nuris (2022) melakukan analisis sentimen terhadap aplikasi PeduliLindungi dengan data yang diperoleh dari Google Play. Metode yang digunakan adalah K-Nearest Neighbor (K-NN) dan hasil yang diperoleh adalah akurasi sebesar 81,74% dan nilai AUC sebesar 0,861, yang termasuk dalam kategori Good Classification.

Astrid Noviriandini, Hermanto, dan Yudhistira (2022) melakukan analisis sentimen terhadap aplikasi PeduliLindungi dengan data yang diperoleh dari Google Play menggunakan metode Naive Bayes berbasis Particle Swarm Optimization. Hasil yang diperoleh adalah akurasi sebesar 93,0% dan nilai AUC sebesar 0,977, menunjukkan bahwa metode Naive Bayes berbasis PSO memiliki akurasi yang lebih tinggi sehingga dapat memberikan solusi terhadap masalah analisis sentimen.

Riky Iskandar Syah, Hoiriyah, dan Miftahul Walid (2023) melakukan penelitian dengan judul Analisis Sentimen Pengguna Media Sosial terhadap Aplikasi M-Health PeduliLindungi menggunakan metode lexicon-based dan Naive Bayes. Kedua metode tersebut digabungkan di mana lexicon-based digunakan untuk pelabelan dan Naive Bayes untuk klasifikasi. Hasil penelitian menunjukkan bahwa pendapat masyarakat cenderung negatif dengan persentase 6% negatif, 62% netral, dan 32% positif di media sosial TikTok; 7% negatif, 56% netral, dan 37% positif di YouTube; dan 100% positif di Twitter. Hasil pengujian model menunjukkan bahwa data dari TikTok memperoleh akurasi 91%, presisi 94%, recall 82%, dan f1-score 86%; data dari YouTube memperoleh akurasi 90%, presisi 93%, recall 81%, dan f1-score 85%; sedangkan data dari Twitter memperoleh akurasi 70%, presisi 23%, recall 33%, dan f1-score 28%. Hasil ini menunjukkan bahwa penggabungan metode lexicon-based dan Naive Bayes memiliki akurasi

yang sangat tinggi pada media sosial TikTok dan YouTube, namun kurang mampu pada media sosial Twitter.

2.5. TF-IDF

Metode TF-IDF adalah metode untuk memberikan bobot pada hubungan antara sebuah kata (Term) dengan dokumen. Metode ini menggabungkan dua konsep: frekuensi kemunculan kata dalam dokumen tertentu (Term Frequency) dan kebalikan dari frekuensi dokumen yang mengandung kata tersebut (Inverse Document Frequency) (Nurjannah & Astuti, 2013). Metode ini terkenal karena efisien, mudah digunakan, dan menghasilkan hasil yang akurat. TF-IDF digunakan untuk mengukur pentingnya sebuah kata dalam sebuah dokumen atau kelompok dokumen.

Dalam metode ini, setiap kalimat dalam dokumen tunggal dianggap sebagai sebuah dokumen. Frekuensi kemunculan kata dalam dokumen menunjukkan seberapa umum kata tersebut (Melita, dkk., 2018). Term Frequency (TF) mengukur seberapa sering sebuah kata muncul dalam dokumen. Karena panjang dokumen dapat bervariasi, ada kemungkinan bahwa sebuah kata akan muncul lebih banyak dalam dokumen yang lebih panjang daripada yang lebih pendek (Balya, 2019). Inverse Document Frequency (IDF) mengukur pentingnya sebuah kata. Ketika menghitung TF, semua kata dianggap memiliki tingkat kepentingan yang sama, namun kita menyadari bahwa ada kata-kata seperti "adalah", "dari", dan "bahwa" yang memiliki sedikit kepentingan (Balya, 2019). Rumus TF-IDF digunakan untuk menghitung bobot sebuah dokumen dengan menggunakan persamaan II-1 dan II-2.

$$TF_{t,d} = \begin{cases} f_{t,d} & f_{t,d} > 0 \\ 1 & f_{t,d} = 0 \end{cases} \quad (II-1)$$

$$IDF_t = \log \left(\frac{N}{df_t} \right) \quad (II-2)$$

$$TFIDF_{t,d} = TF_{t,d} \times IDF_t \quad (II-3)$$

Keterangan :

T = Suatu kata

D = Suatu dokumen

$TF_{t,d}$ = frekuensi term t pada dokumen d

IDF_t = Pendistribusian term t pada seluruh dokumen

$f_{t,d}$ = Jumlah term t yang ada pada dokumen dN = jumlah dokumen

df_t = Jumlah dokumen yang mengandung term

2.6. Confusion Matrix

Confusion matrix adalah metode yang digunakan untuk mengukur performa dari suatu metode klasifikasi. Confusion matrix menyajikan informasi yang membandingkan hasil klasifikasi yang dilakukan oleh sistem untuk mengevaluasi tingkat keakuratannya. Confusion matrix juga digunakan sebagai cara visualisasi hasil pembelajaran sistem, dimana visualisasi ini mencakup dua kategori atau lebih (Rahman, dkk., 2017). Contoh di bawah ini adalah tabel hasil confusion matrix untuk prediksi dua kelas.

Tabel II-6. Confusion Matrix

Klasifikasi Benar	Klasifikasi	
	Positif	Negatif
Positif Benar	True Positif (TP)	False Negatif (FN)
Negatif Benar	False Positif (FP)	True Negatif (TN)

Untuk menghitung akurasi berdasarkan confusion matrix dengan menggunakan tabel di atas, kita dapat menggunakan persamaan berikut

Keterangan:

- a. True Positive (TP): merupakan jumlah data dengan kelas positif yang diklasifikasikan positif.

b. True Negative (TN): merupakan jumlah data dengan kelas negatif yang diklasifikasikan negatif.

c. False Positive (FP): merupakan jumlah data dengan kelas positif yang diklasifikasikan negatif.

d. False Negative (FN): merupakan jumlah data dengan kelas negatif yang diklasifikasikan positif.

Dalam perhitungan akurasi, presisi, dan recall berdasarkan confusion matrix, kita dapat menggunakan persamaan sebagai berikut :

$$Akurasi = \frac{(TP + TN)}{TP + TN + FP + FN} \times 100\% \quad (II-4)$$

$$Presisi = \frac{(TP + TN)}{TP + FP} \times 100\% \quad (II-5)$$

$$Recall = \frac{TP}{TP + FN} \times 100\% \quad (II-6)$$

2.7. Naive Bayes

Teknik klasifikasi yang digunakan dalam analisis sentimen dan penggalian teks adalah metode Naive Bayes. Karena keakuratan dan efisiensi komputasinya, metode ini memiliki potensi yang signifikan dalam klasifikasi. Beberapa pendekatan, termasuk Unigram Nave Bayes, Multinomial Nave Bayes, Complement Nave Bayes, dan Klasifikasi Entropi Maksimum, sering digunakan dalam teknik klasifikasi, terutama dalam konteks Twitter. Teknik karakterisasi Innocent Bayes berarti untuk area kekuatan yang serius untuk menghasilkan berdasarkan keadaan yang ada. Persamaan dan algoritma Bayes digunakan untuk menghitung probabilitas dalam Naive Bayes (Muhamad et al., 2017).

$$P(H|X) = \frac{P(H|X)P(H)}{P(X)} \quad (II-7)$$

Keterangan:

$X =$	=	Data dengan kelas yang belum diketahui;
$H =$	=	Hipotesis data X merupakan suatu kelas spesifik;
$P(H X) =$	=	Probabilitas hipotesis H berdasarkan kondisi X ;
$P(X H) =$	=	Probabilitas hipotesis X berdasarkan kondisi H ;
$P(H) =$	=	Probabilitas hipotesis H ;
$P(X) =$	=	Probabilitas hipotesis X .

Pengikisan data, di mana kata kunci digunakan untuk mengumpulkan data dalam jangka waktu yang telah ditentukan, adalah tahap pertama dari penelitian ini. Setelah informasi dikumpulkan, siklus penamaan diselesaikan untuk menentukan perasaan dari informasi tersebut. Tahap berikutnya adalah preprocessing, di mana informasi dipilih dan diubah menjadi informasi yang terorganisir. Pada tahap ini, prosedur pembersihan dilakukan untuk mengurangi noise dan prosedur penghilangan stopword dilakukan untuk menghilangkan kata-kata yang tidak memiliki arti, seperti “saya”, “dan”, dan “atau”. Interaksi tokenisasi digunakan untuk membedakan kata-kata dan memisahkan kalimat menjadi beberapa istilah berdasarkan spasi dan aksentuasi. Stemming, di mana kata berimbuhan diubah menjadi kata dasar, adalah tahap akhir dari prapemrosesan (Prabowo & Wiguna, 2021).

Particle Swarm Optimization (PSO)

Algoritma Particle Swarm Optimization (PSO) adalah metode komputasi evolusioner yang diperkenalkan pertama kali oleh Kennedy dan Eberhart pada tahun 1995. Algoritma ini terinspirasi dari perilaku sosial binatang, seperti kelompok burung dalam satu kawanan (swarm). Populasi dalam algoritma ini didasarkan pada pencarian algoritma dan dimulai dengan populasi awal yang acak, yang disebut partikel. Melalui pembaruan generasi, solusi optimal dapat ditemukan. Partikel-partikel tersebut bergerak melalui ruang pencarian dengan kecepatan dinamis yang

disesuaikan berdasarkan perilaku historisnya. Dalam proses pencarian, partikel-partikel cenderung bergerak ke area pencarian yang lebih baik.

Algoritma Particle Swarm Optimization (PSO) memiliki perbedaan dengan teknik komputasi evolusioner lainnya karena setiap partikel dalam algoritma ini terkait dengan kecepatan (velocity). Meskipun terdapat kesamaan dengan algoritma genetika, seperti memulai dengan populasi awal yang acak dalam bentuk matriks, algoritma PSO tidak menggunakan operator evolusi seperti crossover dan mutasi yang terdapat dalam algoritma genetika. Partikel dalam PSO dapat dianggap sebagai baris dalam matriks, sedangkan dalam algoritma genetika, baris tersebut disebut sebagai kromosom yang terdiri dari nilai-nilai variabel.

Dalam implementasi algoritma Particle Swarm Optimization (PSO), terdapat beberapa komponen umum yang digunakan dalam teknik optimasi, seperti kelompok (swarm), partikel, Pbest, Gbest, kecepatan (velocity), dan faktor inersia (inertia weight) (Wati, 2011).

2.8.1. Komponen Particle Swarm Optimization (PSO)

Komponen algoritma PSO adalah sebagai berikut (Tuegeh, 2009):

Pengikisan data, di mana kata kunci digunakan untuk mengumpulkan data dalam jangka waktu yang telah ditentukan, adalah tahap pertama dari penelitian ini. Setelah informasi dikumpulkan, siklus penamaan diselesaikan untuk menentukan perasaan dari informasi tersebut. Tahap berikutnya adalah preprocessing, di mana informasi dipilih dan diubah menjadi informasi yang terorganisir. Pada tahap ini, prosedur pembersihan dilakukan untuk mengurangi noise dan prosedur penghilangan stopword dilakukan untuk menghilangkan kata-kata yang tidak memiliki arti, seperti “saya”, “dan”, dan “atau”. Interaksi tokenisasi digunakan untuk membedakan kata-kata dan memisahkan kalimat menjadi beberapa istilah berdasarkan spasi dan aksentuasi. Stemming, di mana kata berimbuhan

diubah menjadi kata dasar, adalah tahap akhir dari prapemrosesan (Prabowo & Wiguna, 2021).

Tahapan-Tahapan Algoritma PSO

Algoritma PSO melibatkan beberapa tahap, termasuk pembangkitan partikel, pembangkitan posisi dan kecepatan partikel, pembaruan kecepatan, dan pembaruan posisi. Partikel berpindah dari satu posisi (iterasi) ke posisi

lainnya berdasarkan pembaruan kecepatan. Tahap pertama adalah pembangkitan partikel, di mana nilai-nilai acak dibangkitkan sesuai dengan permasalahan yang sedang dihadapi. Nilai-nilai acak ini akan menjadi kandidat solusi dalam algoritma PSO dan akan diperbarui seiring dengan kemajuan iterasi.

Tahap kedua adalah pembangkitan posisi X_k^i dan kecepatan v_k^i dari kumpulan partikel. Nilai-nilai ini dibangkitkan secara acak menggunakan batas atas (X_{max}) dan batas bawah (X_{min}) seperti yang ditunjukkan pada persamaan berikut.

$$x_k^i = x_{min} + rand(x_{max} - x_{min}) \quad (II-8)$$

$$v_k^i = x_{min} + rand(x_{max} - x_{min}) \quad (II-9)$$

Langkah berikutnya adalah memperbarui kecepatan (velocity) untuk setiap partikel pada waktu k+1 menggunakan nilai cost dari posisi partikel pada waktu k. Dalam penelitian ini, nilai cost digunakan untuk meminimalkan jarak antara solusi yang dihasilkan dengan nilai cf (confidence factor) dari pakar. Nilai cf pakar merupakan nilai yang dianggap paling optimal dan diperoleh dari hasil wawancara dengan pakar. Tujuan dari penelitian ini adalah untuk mendapatkan solusi yang memiliki jarak yang mendekati 0, meskipun tidak menutup kemungkinan bahwa solusi yang optimal memiliki nilai 0 ketika kecepatan dan posisi diperbarui.

Menggunakan Rumus menghitung perubahan untuk posisi setiap partikel dapat menggunakan fungsi *sigmoid* dengan menggunakan kecepatan

partikel adalah parameternya

$$\text{Sigmoid}(v_i^d) = \frac{1}{1 + e^{-v_i^d}} \quad \text{II-10}$$

$$\text{if sigmoid}(v_i^d) > r3 \text{ then } x_i^d = 1 \text{ else } x_i^d = 0 \quad \text{II-11}$$

Keterangan :

v_i^d = kecepatan partikel

x_i^d = Posisi partikel

$r3$ = Nilai acak yang di gunakan sebagai pencarian solusi

Solusi yang dimaksud di sini adalah nilai partikel atau kandidat solusi yang dihasilkan secara acak, dengan rentang nilai antara 0 hingga 1. Nilai cost dihitung menggunakan persamaan berikut ini:

$$\text{Nilai cost} = (\sum n \text{ Nilai bobot } \times \text{ nilai tiap partikel}) - \text{nilai bobot} \quad \text{(II-12)}$$

Dari nilai cost, kita dapat menentukan partikel yang memiliki nilai global terbaik (global best) dalam kelompok saat ini, serta posisi terbaik dari setiap partikel pada semua waktu sebelumnya. Perumusan pembaruan kecepatan juga melibatkan beberapa parameter acak. Rumus pembaruan kecepatan dapat dilihat dalam persamaan berikut ini (Valle, 2008).

$$v_{k+1}^i = w * v_k^i + c1 * rand * p_{best}^i - x_k^i + c2 * rand * g_{best} - v_k^i \quad \text{(II-13)}$$

Keterangan:

v_k^i = kecepatan partikel i pada iterasi k

w = inertia (fungsi pemberat) $c1$ = self confidence

$c2$ = swarm confidence

rand = nilai acak antara 0 dan 1

v_k^i = posisi partikel i pada iterasi k

p_{best}^i = posisi terbaik dari partikel i

g_{best} = nilai p_{best} terbaik dari *swarm*

Pada setiap iterasi, nilai fungsi pemberat (*inertia*) di-update melalui persamaan berikut (Bansal, 2011).

$$w = w - \frac{w - w_{min}}{iter\ max} \times iter \quad (II-14)$$

Keterangan :

ω_{max} = nilai pemberat (*inertia*) awal

ω_{min} = nilai pemberat (*inertia*) akhir

$iter_{max}$ = jumlah iterasi maksimum

$iter$ = jumlah iterasi terakhir

Langkah terakhir adalah memperbarui posisi setiap partikel. Dengan adanya perubahan kecepatan, posisi partikel juga akan berubah pada setiap iterasi. Pembaruan posisi dapat dihitung menggunakan persamaan berikut ini (Bansal, 2011).

$$x_{k+1}^i = x_k^i + v_{k+1}^i \quad (II-15)$$

Keterangan:

x_{k+1}^i = Posisi partikel saat ini

x_k^i = Posisi partikel sebelumnya

v_{k+1}^i = Kecepatan partikel saat ini

Tiga langkah yang telah dijelaskan akan diulang secara berulang

hinggakriteria kekonvergenan terpenuhi. Kriteria kekonvergenan sangat pentingdalam menghindari peningkatan evaluasi setelah solusi optimal ditemukan. Namun, kriteria kekonvergenan tidak selalu diperlukan secara mutlak. Penetapan jumlah iterasi maksimal juga dapat digunakan sebagai kondisi berhenti dari algoritma. Dalam penelitian ini, kondisi berhenti yang digunakan adalah mencapai jumlah iterasi maksimal, sehingga algoritma akan terus berjalan sampai jumlah iterasi yang ditentukan selesai.

2.8. Rational Unified Process (RUP)

Rational Unified Process adalah proses peningkatan produk yang dilakukan secara bertahap dan berulang. Tujuannya adalah untuk memberikan peningkatan kegunaan secara bertahap pada setiap tahap. Rational Unified Process (RUP), yang dikembangkan oleh Rational Software dan kemudian diakuisisi oleh IBM, merupakan salah satu implementasi terkenal dari Proses Terpadu. Sebelum memulai pengembangan perangkat lunak, komponen penting adalah *Rational Unified Process* (RUP).

Metode RUP bersifat berulang dan menekankan pada kasus penggunaan dan arsitektur yang baik. Beberapa kelemahan potensial dalam pengembangan perangkat lunak dapat diatasi dengan strategi ini. RUP mampu mengatasi kendala-kendala berikut ini:

1. RUP dapat mengakomodasi perubahan kebutuhan, seperti penambahan fungsi atau fitur baru yang diminta oleh pelanggan.
2. Integrasi tidak hanya menjadi proses besar yang dilakukan di akhir proyek, tetapi dapat dilakukan secara bertahap.
3. Risiko dapat diidentifikasi lebih awal pada tahap integrasi awal.
4. RUP memungkinkan perubahan taktis pada produk, misalnya pengembangan dengan waktu yang singkat menghasilkan produk dengan fitur terbatas, dan sebaliknya.
5. RUP mendukung penggunaan kembali (reuse) dalam pengembangan

perangkat lunak.

6. Dengan pendekatan iteratif, cacat produk dapat ditemukan dan diperbaiki dalam beberapa iterasi awal.
7. Pendekatan RUP melibatkan kerja tim dengan anggota proyek, yang lebih baik daripada pendekatan waterfall yang bekerja secara serangkaian. Hal ini menghindari kesalahpahaman antara analis dan programmer.

Dengan demikian, Rational Unified Process (RUP) merupakan pendekatan yang efektif dalam pengembangan perangkat lunak yang dapat mengatasi beberapa tantangan dan memberikan manfaat dalam hal fleksibilitas, integrasi, dan kolaborasi tim.

2.9.1. Tahapan-Tahapan Rational Unified Process (RUP)

Rational Unified Process (RUP) memiliki empat fase atau rentang waktu proses yang membentuk serangkaian target yang harus dicapai secara iteratif. Fase-fase tersebut, menurut Rosa dan Shalahuddin (2018), adalah Inception, Elaboration, Construction, dan Transition.

1. Fase Inception: Fase ini merupakan tahap awal dalam perencanaan pembuatan perangkat lunak baru. Pada tahap ini, dilakukan pemodelan proses bisnis yang diperlukan dan memahami ruang lingkup proyek. Selain itu, juga dibangun kasus bisnis yang diperlukan.
2. Fase Elaboration: Tahap ini berfokus pada perencanaan arsitektur perangkat lunak yang akan dibangun. Resiko-resiko yang mungkin timbul dari arsitektur yang dirancang juga akan diidentifikasi. Pada tahap ini, dilakukan analisis perangkat lunak yang meliputi desain arsitektur sistem, implementasi sistem, pembuatan prototipe, dan pengembangan diagram usecase, sequence, activity, dan class.
3. Fase Construction: Tahap ini merupakan salah satu tahap penting dalam pembuatan perangkat lunak, yaitu tahap pembangunan atau pengkodean. Fokus pada tahap ini adalah pengembangan komponen perangkat lunak

dan implementasi fitur-fitur yang akan disematkan pada perangkat lunak.

4. **Fase Transition:** Tahap ini merupakan tahap akhir dari RUP. Fokus pada tahap ini adalah instalasi perangkat lunak yang telah selesai dibangun. Pelatihan pengguna dan pengujian perangkat lunak juga dilakukan pada tahap ini untuk memastikan apakah perangkat lunak memenuhi harapan. Hasil uji coba akan menghasilkan kesimpulan tentang aplikasi yang telah dibangun.

Dengan demikian, RUP memberikan kerangka kerja yang terstruktur untuk mengelola dan mengembangkan perangkat lunak secara iteratif, dengan memperhatikan aspek perencanaan, desain, pengkodean, dan pengujian.

BAB IV PENGEMBANGAN PRANGKAT LUNAK

4.1. Pendahuluan

Bab ini menyajikan proses pemrograman dan Rational Unified Method untuk mengklasifikasi data mengenai aplikasi satu sehat menggunakan naive bayes dan naive bayes dengan pemilihan fitur particle swarm optimization

4.2. Fase Insepsi

4.2.1. Pemodelan Bisnis

Google Play Store merupakan Platform untuk mendownload berbagai aplikasi yang di dalamnya terdapat fitur komentar yang dapat di gunakan oleh pengguna untuk memberikan sentimen, emosi, pengalaman dan berbagi pendapat mengenai aplikasi yang telah mereka install di perangkat mereka. Oleh karna itu, komentar tersebut harus di kategorikan positif atau negatif sehingga nanti kita dapat mengukur tingkat kepuasan masyarakat terhadap layanan yang di tujukan.

Saat mengklasifikasikan komentar terhadap aplikasi SatuSehat, di perlukan metode Klasifikasi yang berfungsi dengan baik agar klasifikasi dapat di lakukan dengan tepat. Perangkat lunak ini di buat dan desktop dan ditulis dengan bahasa pemrograman python. Program ini dibuat untuk membandingkan kinerja dari metode klasifikasi yang di lakukan oleh naive bayes dan naive bayes dengan seleksi fitur Particle swarm optimization. Hasil metode akan di bandingkan untuk menentukan pendekatan terbaik untuk mengkategorikan sentimen aplikasi satu sehat.

4.2.2. Kebutuhan Fungsional dan non-fungsional

Tabel IV-1 menunjukkan dua jenis kebutuhan perangkat lunak: kebutuhan fungsional dan kebutuhan non-fungsional

Tabel IV-1. Kebutuhan Fungsional dan Non-Fungsional

Kebutuhan Fungsional	Kebutuhan Non-Fungsional
Data dapat diproses oleh perangkat lunak.	Sistem dapat menampilkan pesan kesalahan

Sistem dapat melakukan klasifikasi menggunakan metode Naïve Bayes tanpa seleksi fitur.	Sistem memiliki tampilan antarmuka yang mudah dipahami oleh pengguna.
Sistem dapat melakukan klasifikasi menggunakan metode Naïve Bayes dengan <i>Particle Swarm Optimization Feature Selection</i> .	Sistem dapat diakses menggunakan perangkat laptop atau PC.

4.2.3. Fitur proses data

Program ini memiliki kemampuan pemrosesan data untuk pra-pengolahan data yang dimasukkan. Preprocessing dilakukan untuk menyeragamkan data yang masuk dan mempermudah kategorisasi. *cleaning*, *case folding*, *normalize*, *tokenizing*, *stopword removal*, dan *stemming* adalah contoh prosedur pra-pemrosesan.

4.2.4. Fitur Klasifikasi dengan naive bayes

Fungsi ini digunakan untuk melakukan klasifikasi menggunakan metode *Naive Bayes*. Parameter input adalah data pra-pemrosesan berbobot TF-IDF, dan data teks diubah menjadi vektor.

4.2.5. Fitur Klasifikasi dengan Naive Bayes dan Particle Swarm Optimization

Fungsi ini digunakan untuk melakukan klasifikasi dengan menggunakan metode *Naive Bayes* dan *Particle Swarm Optimization* (PSO). Parameter masukan berupa data yang telah diproses dengan metode pre-processing yang dipilih, Fitur-fitur yang relevan akan dipilih menggunakan algoritma PSO untuk meningkatkan kinerja model *Naive Bayes*.

4.3. Analisis Dan Perancangan

Fase ini menjelaskan analisis kebutuhan dan desain program berdasar kebutuhan fungsional dan non-fungsional, seperti data, metode pra-pengolahan, proses klasifikasi, dan temuan klasifikasi

4.3.1. Analisis Kebutuhan Perangkat Lunak

Berdasarkan pemodelan bisnis yang disebutkan di atas, perangkat lunak yang akan dibuat harus memiliki fitur-fitur berikut.

1. Mampu melakukan pra-pemrosesan data
2. Dapat mengkategorikan data menggunakan metode yang sudah ditentukan
3. Dapat mengklasifikasikan data menggunakan teknik naive bayes dengan particle swarm optimization feature selection

4.3.2. Analisis Data

Data teks berupa komentar yang di kumpulkan dari *Crawling* data Google Play Store, data komentar yang akan digunakan untuk membangun program. Sebanyak 1000 komentar terdiri atas 100 0komentar positif dan 1000 komentar negative. Data ditampilkan pada Tabel IV-2.

Tabel IV-2. Data Hasil *Crawling*

No.	Komentar
1	Kagak jelas gblk mau ganti email sama no telpon GK bisa mau daftar LG di tolak gaje bgtt sumpah pdhal mau liat sertifikat vaksin semakin di persulit
2	Sangat Membantu.
3	Bagus
4	Nomer saya nggak ganti ganti Mau masuk aja ribet banget "Maaf nomer telfon tidak terdaftar" Yang tolong ini saya atau aplikasinya atau servernya Coba lewat email malah nomer verifikasi kedaluarsa padahal limit menitnya masih sisa 1 menit (PAYAH)
5	Apk nya bagus sangat membantu
6	Aplikasi busuk, sudah vaksin ga muncul sertifikatnya
7	Saya sudah vaksin dosis 2, tetapi sertifikat belum muncul padahal sudah lama sekali vaksin.

8	Mantap sekali
9	Good
10	Oke lah
11	Apk gk guna . Buang2 uang rakyat ajah
12	kenapa saya mau masukan email yg terdaftar kok gak bisa iya? dulu peduli lindungi bisa kok masukan email kenapa sekarang gak bisa
13	Kak akun saya belum terdaftar padahal saya sudah vaksin booster 3
14	Saya mau download sertifikat vaksin, tapi kenapa coba lagi. Solusinya gimana ini?
15	ribet bgt, susah login, lambat download sertifikat vaksin gagal terus jlk lah pokoknya
16	Sangat membantu Dan mudah
17	Sertifikat vaksin gua mana min, bodoh kali
18	Min kok gabisa login? Padahal email bner tapi katanya tidak terdaftar -_- padahal di akun itu udah vaksin ketiga gmn tuh
19	Aplikasi apaan seeh ini.. kebijakan mengharuskan tapi dalam layanan menghaduhkan.. aplikasi elit layanan syulit.. pengelola kementerian tapi layanan recehan.. anggaran besar layanan musti sabarr.. download sertifikat aja bikin naik darah.
20	Integrasi Mobile JKN sama SATUsehat Mobile belum ada! Belum bisa lihat Resume Medis, belum bisa Ambil Antrian Online! Tingkatkan jangan Stuck disini aja, untuk daftar Obat atau Vitamin semua penyakit atau Virus belum ada..... Yuk bisa yuk jangan Tidur DeveloperNya, sayang aplikasinya harusnya ada perkembangan lebih mengejutkan dan memanjakan para penggunanya.....
. 1000	

4.3.3. Analisis Pra Pengolahan

Data teks yang telah diproses sebelumnya sebelum kategorisasi untuk membuatnya lebih terorganisir. Tabel IV-3 menunjukkan contoh sepuluh datasampel yang digunakan dalam prosedur pra-pemrosesan

Tabel IV-3. Sampel Data Komentar

No.	Komentar
D1	Kagak jelas gblk mau ganti email sama no telpon GK bisa mau daftar LG di tolak gaje bgtt sumpah pdhal mau liat sertifikat vaksin semakin di persulit
D2	Sangat Membantu.
D3	ribet bgt, susah login, lambat download sertifikat vaksin gagal terus jlk lah pokoknya
D4	Apk nya bagus sangat membantu
D5	Saya mau download sertifikat vaksin, tapi kenapa coba lagi. Solusinya gimana ini?
D6	Sertifikat vaksin gua mana min, bodoh kali
D7	Good
D8	Kak akun saya belum terdaftar padahal saya sudah vaksin booster 3
D9	kenapa saya mau masukan email yg terdaftar kok gak bisa iya? dulu peduli lindungi bisa kok masukan email kenapa sekarang gak bisa
D10	Oke lah

Berikut ini adalah langkah-langkah dari proses pra-pemrosesan

1. Cleaning

Data dibersihkan pada tahap ini dengan menghilangkan tanda baca, angka, dan spasi, mengurangi kebisingan dalam proses klasifikasi. Tabel IV-3 menampilkan hasil *cleaning data* sampel

Tabel IV-4. Proses Cleaning

No.	Komentar
D1	Kagak jelas gblk mau ganti email sama no telpon GK bisa mau daftar LG di tolak gaje bggt sumpah pdhal mau liat sertifikat vaksin semakin di persulit
D2	Sangat Membantu
D3	ribet bgt, susah login lambat download sertifikat vaksin gagal terus jlk lah pokoknya
D4	Apk nya bagus sangat membantu
D5	Saya mau download sertifikat vaksin, tapi kenapa coba lagi. Solusinya gimana ini
D6	Sertifikat vaksin gua mana min bodoh kali
D7	Good
D8	Kak akun saya belum terdaftar padahal saya sudah vaksin booster 3
D9	kenapa saya mau masukan email yg terdaftar kok gak bisa iya dulu peduli lindungi bisa kok masukan email kenapa sekarang gak bisa
D10	Oke lah

2. Case Folding

Tahap ini adalah tahap di mana semua huruf dikonversi ke huruf kecil.

Tabel IV- 5 menampilkan hasil tahap ini.

Tabel IV-5. Hasil *Case Folding* Sampel Data

No.	Komentar
D1	Kagak jelas gblk mau ganti email sama no telpon GK bisa mau daftar LG di tolak gaje bggt sumpah pdhal mau liat sertifikat vaksin semakin di persulit
D2	Sangat Membantu
D3	ribet bgt, susah login lambat download sertifikat vaksin gagal terus jlk lah pokoknya

D4	apk nya bagus sangat membantu
D5	saya mau download sertifikat vaksin, tapi kenapa coba lagi. Solusinya gimana ini
D6	sertifikat vaksin gua mana min bodoh kali
D7	good
D8	Kak akun saya belum terdaftar padahal saya sudah vaksin booster 3
D9	kenapa saya mau masukan email yg terdaftar kok gak bisa iya dulu peduli lindungi bisa kok masukan email kenapa sekarang gak bisa
D10	oke lah

3. Normalize

Tahap ini akan mengubah kata-kata non-standar menjadi kata-kata standar.

Tabel IV-6 menampilkan hasil normalisasi.

Table IV-6. Hasil dari normalize data

No.	Komentar
D1	Kagak jelas gblk mau ganti email sama no telpon GK bisa mau daftar LG di tolak gaje bgtt sumpah pdhal mau liat sertifikat vaksin semakin di persulit
D2	Sangat Membantu
D3	ribet bgt, susah login lambat download sertifikat vaksin gagal terus jlk lah pokoknya
D4	apk bagus sangat membantu
D5	saya mau download sertifikat vaksin, tapi kenapa coba lagi. Solusinya gimana ini
D6	sertifikat vaksin gua mana min bodoh kali
D7	good

D8	Kak akun saya belum terdaftar padahal saya sudah vaksin booster 3
D9	kenapa saya mau masukan email yg terdaftar kok gak bisa iya dulu peduli lindungi bisa kok masukan email kenapa sekarang gak bisa
D10	oke lah

4. Tokenizing

Pada tahap tokenizing data akan dipisahkan menjadi menjadi kata - perkata. Tabel IV-7 menunjukkan hasil *tokenizing*

Tabel IV-7. Hasil dari tokenizing

No.	Komentar
D1	[Kagak], [jelas], [gblk], [mau],[ganti],[email], [sama], [no], [telpon], [GK], [bisa], [mau] ,[daftar], [LG],[di], [tolak] ,[gaje],[bgtt],[sumpah],[pdhal],[mau],[liat],[sertifikat],[vaksin], [semakin],[di],[persulit]
D2	[Sangat], [Membantu]
D3	[Ribet], [bgt],[susah],[login], [lambat], [download], [sertifikat], [vaksin], [gagal] [terus], [jlk], [lah], [pokoknya]
D4	[apk], [bagus], [sangat], [membantu]
D5	[saya], [mau], [download], [sertifikat] ,[vaksin], [tapi], [kenapa] ,[coba], [lagi] [solusinya], [gimana] ,[ini]
D6	[sertifikat], [vaksin], [gua] ,[mana], [min], [bodoh], [kali]
D7	[good]
D8	[Kak], [akun], [saya],[belum], [terdaftar], [padahal], [saya] ,[sudah] [vaksin], [booster], [3]
D9	[kenapa], [saya] ,[mau], [masukan] ,[email], [yg] ,[terdaftar], [kok],[gak], [bisa], [iya], [dulu], [peduli], [lindungi] ,[bisa] ,[kok] ,[masukan], [email], [kenapa] [sekarang], [gak], [bisa]
D10	[Oke], [lah]

5. Stopword Removal

Tahap ini menghapus frasa atau kata dari kamus tergantung pada stoplist. Tabel IV-8 menunjukkan hasil penghapusan stopwords dari data komentar sampel.

Table IV-8. Hasil dari stopword Removal

No.	Komentar
D1	Kagak jelas mau ganti email sama no telpon bisa mau daftar di tolak gaje sumpah mau liat sertifikat vaksin semakin di persulit
D2	sangat membantu
D3	ribet susah login lambat download sertifikat vaksin gagal terus lah pokoknya
D4	apk bagus sangat membantu
D5	saya mau download sertifikat vaksin tapi kenapa coba lagi solusinya gimana ini
D6	sertifikat vaksin gua mana bodoh kali
D7	good
D8	Kak akun saya belum terdaftar padahal saya sudah vaksin booster 3
D9	kenapa saya mau masukan email terdaftar kok gak bisa iya dulu peduli lindungi bisa kok masukan email kenapa sekarang gak bisa
D10	oke lah

6. Stemming

Langkah *stemming* akan mencari kata dasar dari setiap kata yang terbentuk selama tahap penghapusan *stopword*. Tabel IV-9 menampilkan hasil yang berasal dari data komentar sampel

Table IV-9. Hasil Dari Stemming

No.	Komentar
D1	Kagak jelas mau ganti email sama no telpon bisa mau daftar di tolak gaje sumpah mau liat sertifikat vaksin semakin di persulit
D2	sangat membantu
D3	ribet susah login lambat download sertifikat vaksin gagal terus lah pokoknya
D4	apk bagus sangat membantu
D5	saya mau download sertifikat vaksin tapi kenapa coba lagi solusinya gimana ini
D6	sertifikat vaksin gua mana bodoh kali
D7	good
D8	Kak akun saya belum terdaftar padahal saya sudah vaksin booster 3
D9	kenapa saya mau masukan email terdaftar kok gak bisa iya dulu peduli lindungi bisa kok masukan email kenapa sekarang gak bisa
D10	oke lah

4.3.4. Analisis Proses Klasifikasi Menggunakan Naive Bayes

Naive Bayes digunakan pada proses klasifikasi dalam dua tahap, yaitu pelatihan dan pengujian. Klasifikasi data yang ada pada penelitian ini dibagi menjadi dua scenario: klasifikasi hanya menggunakan metode Naive Bayes dan klasifikasi menggunakan metode Naive Bayes dengan pemilihan fitur Particle Swarm Optimization.

Teknik pelatihan digunakan untuk menghitung nilai probabilitas prior untuk setiap kelas yang ada pada data pelatihan, dan juga setiap atribut di dalam masing masing kelas akan dihitung berdasarkan frekuensi kemunculannya di dalam setiap kelas itu sendiri. Pada pelatihan ini kita akan menggunakan hukum bayes untuk menghitung probabilitas kondisional

masing-masing kelas berdasarkan atribut yang diberikan. Pada Teknik pelatihan ini kita menggunakan 4 buah datayang mewakili setiap kelas.

Table IV-10.Data yang di gunakan untuk perhitungan naive bayes

No	Komentar	Label
1	Trima kasih sudah membantu saya	POSITIF
2	bikin badmood bgttt mau unduh sertifikat eror terus pas udh keunduh ga masuk galeri diunduh lagi eror lagi, kembalin apk pedulilindungi!!!!!!	NEGATIF
3	Tolong di perbaiki untuk login nya masak saya sudah ada kode untuk masuk kan malah eror padahal cuman beberapa detik masuk malah eror	NEGATIF
4	media Informasi yang baik	POSITIF

Setelah dilakukan pra-pengolahan, langkah selanjutnya yaitu kita melakukan pembobotan menggunakan tf (*term frequency*) / jumlah kemunculan kata.

Table IV-11.Pra-Pengolahan

No.	Term	Tf(Positif)	Tf(Negatif)
1	membantu	1	0
2	baik	1	0
3	terima	1	0
4	kasih	1	0
5	engga	0	1
6	error	0	1
7	perbaiki	0	1
8	malah	0	1

9	kembaliin	0	1
10	badmood	0	1
Jumlah <i>term</i>		4	6

Diperoleh :

Count positif=4, count negative =6, dengan total 10 term yang didapat

4.3.5. Proses Klasifikasi Data Latih

1. Hitung probabilitas prior setiap kelas, disini kita memilih 2 kelas ya itu positif dan negatif.

$$P \frac{\text{Positif}}{\text{Negatif}} = \frac{x(\text{positif/negatif})}{|c|} \quad (\text{IV-1})$$

$$(P(\text{positif}) \frac{fx(\text{positif})}{|c|} = \frac{2}{4}$$

$$= 0.5$$

$$(P(\text{Negatif}) \frac{fx(\text{Negatif})}{|c|} = \frac{2}{4}$$

$$= 0.5$$

2. Menghitung probabilitas likelihood setiap term dari keseluruhan dokumen. Jumlah seluruh kata 10, 4 trem dari kelas positif, dan 6 trem dari kelas negatif.

$$P(w|\text{positif|negatif}) = \frac{(n_k(\text{positif|negatif}) + 1)}{(n_{\text{positif|negatif}} + |T|)} \quad (\text{IV-2})$$

Probabilitas kata "membantu"

$$P(\text{positif}) \frac{1 + 1}{4 + 10}$$

$$= 0.142857142857$$

$$P(\text{negatif}) \frac{0 + 1}{6 + 10} = 0.0625$$

Probabilitas kata "baik"

$$P(\text{positif}) \frac{1+1}{4+10} = 0.142857142857$$

$$P(\text{negatif}) \frac{0+1}{6+10} = 0.0625$$

Probabilitas kata "terima"

$$P(\text{positif}) \frac{1+1}{4+10} = 0.142857142857$$

$$P(\text{negatif}) \frac{0+1}{6+10} = 0.0625$$

Probabilitas kata "kasih"

$$P(\text{positif}) \frac{1+1}{4+10} = 0.142857142857$$

$$P(\text{negatif}) \frac{0+1}{6+10} = 0.0625$$

Probabilitas kata "engga"

$$P(\text{positif}) \frac{0+1}{4+10} = 0.07142857$$

$$P(\text{negatif}) \frac{1+1}{6+10} = 0.125$$

Probabilitas kata "error"

$$P(\text{positif}) \frac{0+1}{4+10} = 0.07142857$$

$$P(\text{negatif}) \frac{1+1}{6+10} = 0.125$$

Probabilitas kata "perbaiki"

$$P(\text{positif}) \frac{0+1}{4+10} = 0.07142857$$

$$P(\text{negatif}) \frac{1+1}{6+10} = 0.125$$

Probabilitas kata "malah"

$$P(\text{positif}) \frac{0 + 1}{4 + 10} = 0.07142857$$

$$P(\text{negatif}) \frac{1 + 1}{6 + 10} = 0.125$$

Probabilitas kata "kembaliin"

$$P(\text{positif}) \frac{0 + 1}{4 + 10} = 0.07142857$$

$$P(\text{negatif}) \frac{1 + 1}{6 + 10} = 0.125$$

Probabilitas kata "badmood"

$$P(\text{positif}) \frac{0 + 1}{4 + 10} = 0.07142857$$

$$P(\text{negatif}) \frac{1 + 1}{6 + 10} = 0.125$$

4.3.6. Proses Klasifikasi Data Uji

Proses Klasifikasi data uji dilakukan dengan mengalikan semua nilai peluang. Nilai yang lebih tinggi merupakan kelas baru dari data tersebut

$$P(d) = P\left(\frac{\text{positif}}{\text{negatif}}\right) \times p(w|\text{positif}|\text{negatif}) \quad (\text{IV-3})$$

Pada data uji "Trima kasih sudah membantu saya" yang termasuk ke dalam data training adalah kata "terima" dan "membantu"

$P(\text{ujilpositif})$

$= P(\text{positif}) \times P(\text{terimalpositif}) \times P(\text{membantulpositif})$

$= 0.5 \times 0.142857132857 \times 0.142857142857$

$= 0.01020408163$

$P(\text{ujilnegatif})$

$$=P(\text{negatif}) \times P(\text{terimalnegatif}) \times P(\text{membantunegatif})$$

$$=0.5 \times 0.0625$$

$$=0.001953125$$

Pada data uji” Pusing buat str eror terus” yang termasuk ke dalam data training adalah kata”eror”

$$P(\text{ujilpositif})$$

$$=P(\text{positif}) \times P(\text{erorlpositif})$$

$$=0.5 \times 0.0625$$

$$=0.03125$$

$$P(\text{ujilnegatif})$$

$$=P(\text{negatif}) \times P(\text{erorlnegatif})$$

$$=0.5 \times 0.0714285714285$$

$$=0.0357142857142$$

Kesimpulan : Nilai probabilitas tertinggi yaitu sebesar **0.0357142857142857** pada $P(\text{ujilnegatif})$ sehingga komentar tersebut diklasifikasikan ke dalam kelas “**Negatif**”

4.3.7. Perhitungan manual Particle Swarm Optimization

Langkah pertama sebelum melakukan seleksi fitur menggunakan PSO adalah inialisasi populasi partikel, pembangkitan posisi serta kecepatan awal partikel Sebanyak 10 fitur dipilih sebagai populasi partikel

Table IV-.11 fitur

NO	FITUR	FREKUENSI
1	Aplikasi	265

TABLE IV-13.KECEPATAN AWAL RANDOM B

Kecepatan Awal (B):										
P1	0.939	0.550	0.878	0.789	0.195	0.843	0.823	0.608	0.909	0.349
P2	0.222	0.457	0.657	0.286	0.438	0.085	0.718	0.604	0.850	0.973
P3	0.544	0.602	0.988	0.630	0.413	0.458	0.542	0.851	0.283	0.768
P4	0.726	0.246	0.338	0.872	0.378	0.349	0.459	0.275	0.774	0.055
P5	0.760	0.426	0.521	0.122	0.541	0.012	0.045	0.144	0.949	0.784
P6	0.937	0.272	0.371	0.017	0.815	0.501	0.179	0.511	0.542	0.846
P7	0.420	0.677	0.859	0.616	0.266	0.742	0.607	0.626	0.785	0.078
P8	0.614	0.792	0.363	0.779	0.544	0.145	0.636	0.721	0.986	0.021
P9	0.165	0.555	0.760	0.324	0.885	0.353	0.037	0.406	0.272	0.943
P10	0.924	0.496	0.005	0.398	0.430	0.926	0.219	0.005	0.897	0.442

Kemudian dilakukan perhitungan nilai Sigmoid untuk setiap partikel
 Hasil perhitungan nilai Sigmoid pada iterasi 1 ditunjukkan pada tabel berikut

Nilai Sigmoid									
0.718	0.634	0.706	0.687	0.548	0.699	0.305	0.352	0.287	0.413
0.444	0.387	0.658	0.428	0.392	0.521	0.327	0.646	0.299	0.725
0.632	0.353	0.271	0.652	0.601	0.613	0.635	0.299	0.429	0.312
0.673	0.561	0.416	0.294	0.406	0.413	0.612	0.568	0.684	0.480
0.318	0.605	0.627	0.530	0.632	0.496	0.511	0.463	0.279	0.313
0.281	0.567	0.408	0.504	0.693	0.377	0.455	0.625	0.366	0.300
0.396	0.336	0.702	0.649	0.566	0.677	0.644	0.348	0.686	0.519
0.649	0.688	0.409	0.314	0.367	0.463	0.653	0.672	0.271	0.505
0.458	0.364	0.318	0.419	0.708	0.412	0.509	0.600	0.432	0.719
0.716	0.378	0.498	0.401	0.394	0.283	0.554	0.5013	0.289	0.391

Table IV-14.Nilai Sigmoid

Kemudian dilakukan perbandingan nilai $X_i^{(1)}$ atau A dengan nilai Sigmoid pada iterasi-1. Hasil perbandingan nilai sigmoid disajikan pada tabel IV-21

Table IV-15. Posisi baru setelah update

POSISI BARU SETELAH UPDATE BERDASARKAN SIGMOID										
P1	1	1	1	1	1	1	0	0	0	0
P2	0	0	1	0	0	1	0	1	0	1
P3	1	0	0	1	1	1	1	0	0	0
P4	1	1	0	0	0	0	1	1	1	0
P5	0	1	1	1	1	0	1	0	0	0
P6	0	1	0	1	1	0	0	1	0	0
P7	0	0	1	1	1	1	1	0	1	1
P8	1	1	0	0	0	0	1	1	0	1
P9	0	0	0	0	1	0	1	1	0	1
P10	1	0	0	0	0	0	1	1	0	0

Keterangan:

0 : Tidak di pilih

1 : Terpilih

Table IV-16. Kata kata yang di pilih

F1	aplikasi	vaksin	sertifikat'	apk	'bagus	masuk
F2	sertifikat	masuk	udah	membantu		
F3	aplikasi	apk	bagus	'masuk	login	
F4	aplikasi	vaksin	login	'udah	susah	
F5	vaksin	sertifikat	apk	'bagus	login	
F6	vaksin	apk	bagus	udah		
F7	sertifikat	apk	bagus	masuk	login	'usah
F8	aplikasi	vaksin	login	'udah	membantu	
F9	bagus	login	udah	membantu		
F10	aplikasi	login	udah			

Setelah itu akan dilakukan evaluasi nilai *Fitness* Partikel, evaluasi nilai *fitness* dilakukan untuk memperoleh nilai P_{best} setiap partikel dan satu nilai G_{best} . Nilai *fitness*

Nilai *fitness* yang telah diperoleh disebut juga sebagai P_{best} . Nilai seluruh partikel dituliskan pada tabel IV-17

Table IV-17.Nilai G_{best}

Partikel	P_{best}
P1	0.71
P2	0.72
P3	0.63
P4	0.68
P5	0.63
P6	0.69
P7	0.70
P8	0.68
P9	0.71
P10	0.71

Pada tabel IV-25 terlihat bahwa P54 memiliki nilai P_{best} terbesar yaitu 5.6, sehingga nilai G_{best} dari 10 partikel tersebut adalah 5.6. Berdasarkan tabel IV-X, fitur yang terpilih pada iterasi-1 adalah fitur yang dimiliki oleh P4

Evaluasi nilai *fitness* akan terus dilakukan sampai iterasi berhenti, sehingga akan dilakukan *update* komponen partikel dengan meng-*update* posisi dan kecepatan partikel untuk iterasi berikutnya.

Sebelum melakukan *update* komponen partikel, terlebih dahulu menentukan parameter-parameter yang akan digunakan. Parameter yang digunakan dapat dilihat pada tabel IV-18

Table IV-18.Parameter PSO

C_1	1.2
C_2	0.6
ω	0.1
t_{max}	3

Update kecepatan partikel pada iterasi ke-2

Table IV-19.Update kecepatan random literasi 2

Kecepatan Baru (B) Setelah update:										
F1	-1,174	-0.17	-0.08	-0.508	-1163	-0.249	0.479	-	-	0.838
F2	-0.022	-0.045	0.694	0.282	0.590	-1438	0.731	0.361	0.0361	-0.78
F3	-0.431	-0.060	1,04	0.063	0.041	-0.255	0.054	0.378	109576	0.088
F4	-0.870	-0.314	1,3	102	0.618	-0.034	0.045	0.963	0.7300	-0.005
F5	-0.076	1,301	0.05	0.012	0.054	0.417	0.004	-0.01	0.9695	0.857
F6	-0.093	-0.29	0.287	0.001	-0.646	-0.039	0.413	-0.93	-0.054	-0.084
F7	-0.042	-0.067	0.696	-0.207	-0.827	-0.343	0.060	0.062	-0.015	-0.142
F8	-0.268	0.3495	0.359	0.531	0.518	0.2782	0.063	0.298	-0.098	-0.143
F9	-0.016	0.5475	-0.06	0.62	-0.145	-0.035	0.003	0.410	0.747	0.035
F10	-0.202	1,095	1,20	0.66	0.214	1021	0.021	0.026	1066	-0.044

Table IV-20 Nilai sigmoid literasi 2

Nilai Sigmoid Literasi 2										
F1	0.236	0.456	0.478	0.375	0.239	0.432	0.617	0.484	0.477	0.698
F2	0.494	0.488	0.333	0.570	0.643	0.193	0.675	0.410	0.490	0.312
F3	0.393	0.484	0.733	0.515	0.510	0.436	0.513	0.593	0.749	0.522
F4	0.295	0.422	0.786	0.735	0.649	0.496	0.512	0.276	0.325	0.498
F5	0.480	0.257	0.513	0.503	0.516	0.602	0.501	0.496	0.725	0.702
F6	0.476	0.427	0.571	0.505	0.344	0.490	0.601	0.281	0.486	0.478
F7	0.489	0.483	0.332	0.445	0.304	0.415	0.515	0.484	0.496	0.464
F8	0.433	0.413	0.588	0.629	0.626	0.569	0.515	0.426	0.475	0.463
F9	0.495	0.633	0.481	0.652	0.463	0.491	0.500	0.398	0.678	0.508
F10	0.449	0.749	0.768	0.660	0.553	0.735	0.505	0.493	0.743	0.488

Update Posisi Literasi 2

Table IV-21 Posisi baru setelah update berdasarkan sigmoid

Posisi Baru Setelah update berdasarkan Sigmoid:										
F1	0	0	0	0	0	0	1	0	0	1
F2	0	0	0	1	1	0	1	0	0	0
F3	0	0	1	1	1	0	1	1	1	1
F4	0	0	1	1	1	0	1	0	0	0
F5	0	0	1	1	1	1	1	0	1	1
F6	0	0	1	1	0	0	1	0	0	0
F7	0	0	0	0	0	0	1	0	0	0
F8	0	0	1	1	1	1	1	0	0	0
F9	0	1	0	1	0	0	1	0	1	1
F10	0	1	1	1	1	1	1	0	1	0

Pada tabel IV-30 terlihat bahwa terjadi eliminasi beberapa partikel yang diakibatkan oleh kecepatan dan posisi partikel yang berubah, parameter P_{best} dan G_{best} pada iterasi pertama, serta nilai c_1, c_2 dan ω . Partikel-partikel tersebut akan memasuki tahap iterasi selanjutnya

Table IV-22. Kata yang dipilih oleh fitur

F1	login	membantu					
F2	apk	bagus	login				
F3	sertifikat	apk	bagus,	login	udah	susah	membantu
F4	sertifikat	apk	bagus	login			
F5	sertifikat	apk	bagus	masuk	login	susah	membantu
F6	sertifikat	apk	login				
F7	login						
F8	sertifikat	apk	bagus	masuk	login		
F9	vaksin	apk	login	susah	membantu		
F10	vaksin	sertifikat	apk	bagus	masuk	login	susah

Table IV-23. Nilai P (Best) Pada literasi 2

Partikel	P_{best}
P1	0.47
P2	0.67
P3	0.74

P4	0.78
P5	0.72
P6	0.60
P7	0.51
P8	0.62
P9	0.67
P10	0.76

Literasi 3

Table IV-24.Update kecepatan random

Kecepatan Baru (B) Setelah update:										
F1	-1,17	1,01	2,45	4,92	-496	7,45	1,61	-6,08	-909	126
F2	-222	-4,57	3,78	-626	5,90	-1,43	7,31	1,050	2313	-787
F3	7,78	-6,02	1,03	6,30	4,13	3,57	5,42	-2,944	-472	-305
F4	-8,70	-3,14	1,30	1,02	6,18	-3,49	4,59	-9,64	-730	-555
F5	-7,60	-1,05	5,21	1,22	5,41	-1,70	4,54	-1,44	-487	-505
F6	-9,37	1,05	-5,77	1,71	3,46	1,07	4,13	-9,38	-542	-846
F7	-4,20	-677	4,89	4,98	-556	5,03	6,07	-6,26	805	888
F8	-2,68	-349	3,59	5,31	5,18	-2,53	6,36	-2,94	-986	-146
F9	-1,65	-5,45	5,91	6,28	4,46	-3,53	3,75	-4,10	-195	-739
F10	8,157	-2,54	1,20	6,64	2,14	-1,12	2,19	9,01	-205	-442

Table IV-25.Nilai Sigmoid literasi 3

Nilai Sigmoid literasi 3										
F1	0.470	0.733	0.561	0.620	0.487	0.678	0.504	0.498	0.497	0.503
F2	0.499	0.498	0.593	0.348	0.514	0.464	0.518	0.740	0.557	0.480
F3	0.519	0.498	0.525	0.501	0.509	0.588	0.501	0.426	0.384	0.424
F4	0.478	0.492	0.532	0.525	0.515	0.499	0.501	0.475	0.481	0.499
F5	0.498	0.473	0.502	0.500	0.501	0.457	0.500	0.499	0.380	0.376
F6	0.497	0.741	0.359	0.500	0.585	0.744	0.510	0.476	0.498	0.497
F7	0.498	0.498	0.620	0.622	0.486	0.623	0.501	0.498	0.691	0.708
F8	0.493	0.491	0.508	0.513	0.512	0.436	0.501	0.492	0.498	0.496
F9	0.499	0.367	0.643	0.515	0.609	0.499	0.500	0.489	0.451	0.481
F10	0.520	0.436	0.529	0.516	0.505	0.471	0.500	0.711	0.448	0.498

Table IV-26. Posisi literasi 3

Posisi Baru Setelah update berdasarkan Sigmoid:										
F1	0	1	1	1	0	1	1	0	0	1
F2	0	0	1	0	1	0	1	1	1	0
F3	1	0	1	1	1	1	1	0	0	0
F4	0	0	1	1	1	0	1	0	0	0
F5	0	0	1	1	1	0	1	0	0	0
F6	0	1	0	1	1	1	1	0	0	0
F7	0	0	1	1	0	1	1	0	1	1
F8	0	0	1	1	1	0	1	0	0	0
F9	0	0	1	1	1	0	1	0	0	0
F10	1	0	1	1	1	0	1	1	0	0

Table IV-27. Fitur Terpilih

F1	vaksin	sertifikat	apk	masuk	login	membantu
F2	sertifikat	bagus	login	udah	susah	
F3	aplikasi	sertifikat	apk	bagus	masu	login
F4	sertifikat	apk	bagus	login		
F5	sertifikat	apk	bagus	login		
F6	vaksin	apk	bagus	masuk	login	
F7	sertifikat	apk	masuk	login	susah	membantu
F8	sertifikat	apk	bagus	login		
F9	sertifikat	apk	bagus	login		
F10	aplikasi	sertifikat	apk	bagus	login	udah

Table IV-28. Pbest literasi 3

Partikel	Pbest
P1	0.73
P2	0.74
P3	0.58
P4	0.53
P5	0.50
P6	0.74

P7	0.70
P8	0.51
P9	0.74
P10	0.71

Pada table IV-37 menampilkan bahwa P4 memiliki nilai fitness tertinggi diantara partikel-partikel lainnya, berikut adalah tabel perbandingan nilai Pbest pada iterasi-1, iterasi-2 dan iterasi-3

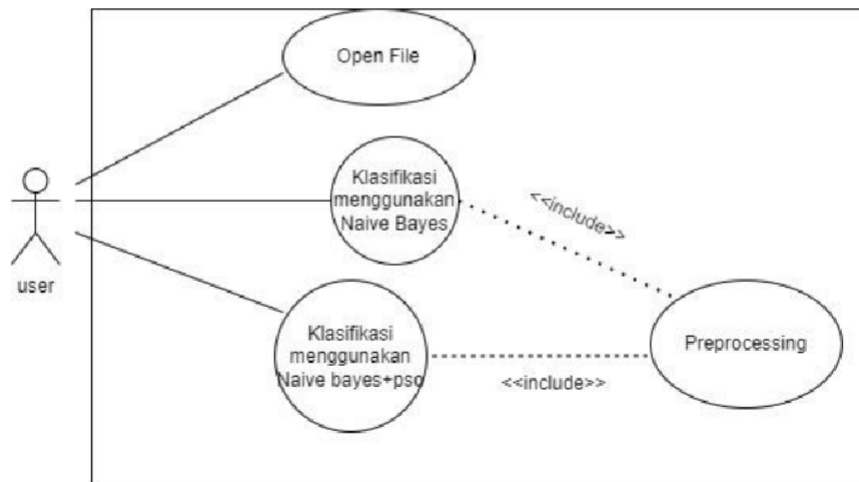
TableIV-29. P best keseluruhan literasi

Iterasi	P _{best}
1	0.72
2	0.78
3	0.74

Tabel IV-38 menunjukkan bahwa terjadi peningkatan nilai Pbest pada iterasi-2 dengan nilai **0.72** yang ditunjukkan pada partikel ke-4 atau P4, dan karena maksimal iterasi sudah terpenuhi maka didapatkan nilai G_{best} pada partikel ke 4, fitur-fitur yang ada pada P4 yaitu Sertifikat, apk, bagus, login

Fitur-fitur tersebut adalah fitur yang terpilih dan akan digunakan pada saat proses klasifikasi menggunakan metode Naïve Bayes.

4.3.8. Implementasi



Gambar IV-1. Diagram *Use Case*

Table IV- adalah Table yang menunjukkan deskripsi diagram Use Case

Table IV-30 Use case

No.	Aktor	Deskripsi
1.	User	Seseorang yang mengoperasikan dan mengelola fitur perangkat lunak

Deskripsi dari diagram use case berupa table definisi use case di tunjukan pada table IV-

Table IV-31. Table definisi Use Case

No.	Use case	Definisi
1	Open File	Prosedur ini melakukan input data berupa file dan memasukanya ke dalam program
2	Klasifikasi menggunakan naive bayes	Proses ini berfungsi untuk meng-klasifikasi data

		menggunakan metode naive bayes
3	Klasifikasi menggunakan naive bayes + PSO	Proses ini memilih fitur menggunakan pso dan kemudian mengklasifikasikan data menggunakan naive bayes

Bedasarkan table di atas skenario dari tiap use case akan di uraikan sebagai berikut

Kode	001
Nama Use Case	Open File
Aktor	User
Tujuan	Pra-pemrosesan data yang masuk sebelum melalui prosedur kategorisasi
Kondisi Awal	Belum ada input data.
Kondisi Akhir	Dataset dari file akan di tampilkan

Aktor	Sistem
1. Klik tombol Brows File	
	2. Tampilkan window pencarian file.
3. Pilih file dataset yang akan di buka	
	4. Tampilkan data yang di pilih

Kode	002
Nama Use Case	Klasifikasi menggunakan <i>Naïve Bayes</i>
Aktor	User
Tujuan	Gunakan metode <i>Naïve Bayes</i> untuk mengklasifikasikan data.
Kondisi Awal	Data berhasil di load
Kondisi Akhir	Menampilkan nilai evaluasi dari hasil klasifikasi data dengan <i>Naïve Bayes</i>

Aktor	Sistem
1. Pilih metode <i>Naïve Bayes</i>	
2. Tekan tombol Klasifikasi	
	3. Memuat Model <i>Naïve Bayes</i>
	4. Melakukan Preprocessing Data
	5. Melakukan Perhitungan TFIDF
	6. Klasifikasi dengan <i>Naïve Bayes</i>
	7. Menghitung nilai evaluasi
	8. Menampilkan nilai evaluasi
	9. Menampilkan hasil klasifikasi

Kode	003
Nama Use Case	Klasifikasi menggunakan <i>Naïve Bayes</i> dan PSO
Aktor	User
Tujuan	Melakukan klasifikasi data menggunakan Model <i>Naïve Bayes</i> dengan Seleksi Fitur PSO
Kondisi Awal	Data berhasil di load
Kondisi Akhir	Menampilkan nilai evaluasi dari hasil klasifikasi data dengan <i>Naïve Bayes</i> dan PSO

Aktor	Sistem
1. Pilih metode <i>Naïve Bayes + PSO</i>	
2. Tekan tombol Klasifikasi	
	3. Memuat Model <i>Naïve Bayes + PSO</i>
	4. Melakukan Preprocessing Data
	5. Melakukan Perhitungan TFIDF
	6. Melakukan Seleksi Fitur

	7. Melakukan Klasifikasi dengan fitur terpilih
	8. Menghitung nilai evaluasi
	9. Menampilkan nilai evaluasi
	10. Menampilkan hasil klasifikasi

Kode	004
Nama Use Case	Preprocessing
Aktor	-
Tujuan	Melakukan proses pra-pengolahan pada data
Kondisi Awal	Data berhasil di load
Kondisi Akhir	Data berhasil melewati proses pra-pengolahan dan mengembalikan text yang sudah di stemmed

Aktor	Sistem
	1. Melakukan <i>load file stopwords</i>
	2. Melakukan <i>load file normalisasi</i>
	3. Melakukan proses <i>cleaning data</i>
	4. Melakukan proses <i>case folding</i>
	5. Melakukan proses <i>normalisasi</i>
	6. Melakukan proses <i>tokenize</i>
	7. Melakukan proses <i>stopwords</i>
	8. Melakukan proses <i>stemming</i>

4.3.9. Fase Elaborasi

Tahap kedua dalam pengembangan perangkat lunak adalah identifikasi sistem. Fase akan terdiri dari desain data, desain antarmuka pengguna dan pemodelan

4.3.8.1. Perancangan Antarmuka

Bagian ini akan menampilkan dan membahas bagaimana antarmuka pengguna pada perangkat lunak di bawah ini adalah desain antar muka pengguna

The image shows a web application interface titled "Analisis sentimen aplikasi satu sehat menggunakan metode naive bayes dan particle swarm optimization". The interface includes a "Browse File" button, a "Data" input field, a "Praproses Data" button, a "PILIH METODE" section with radio buttons for "Naive Bayes" and "Naive Bayes + PSO", another "Praproses Data" button, and a "Confusion Matrix" section with sub-labels for "Akurasi", "Presisi", "Recall", and "F1 score".

Gambar IV-1. Rancangan antarmuka

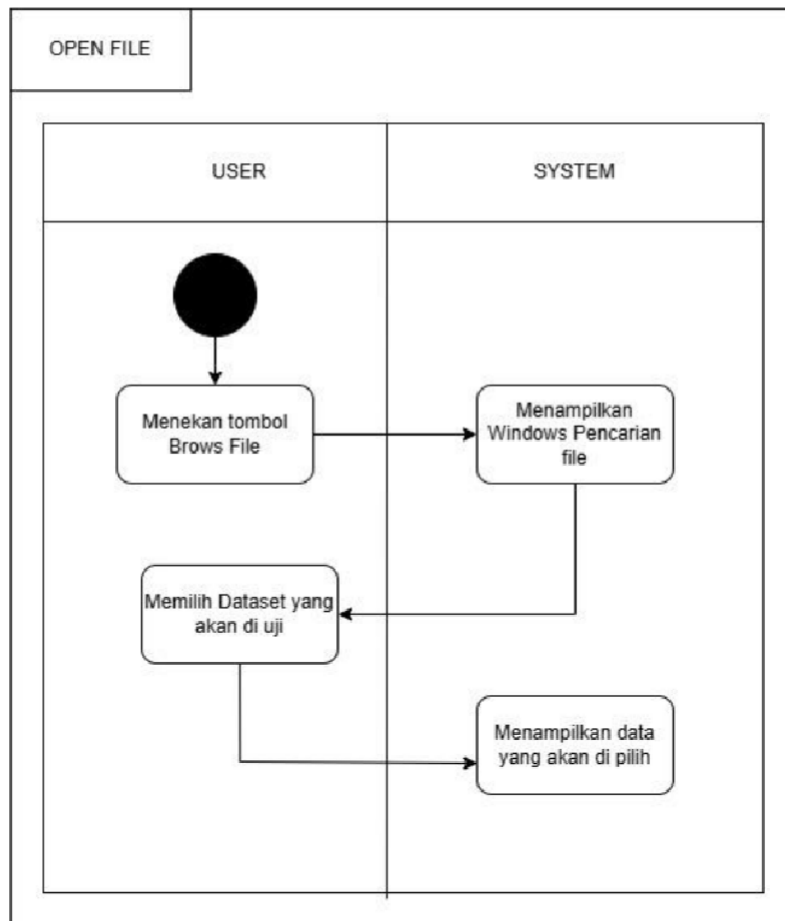
4.3.9. Kebutuhan

Bagian ini menjelaskan kebutuhan spesifikasi perangkat lunak yang akan di buat berdasarkan hasil design pada tahap sebelumnya. Dan saya menggunakan bahasa pemrograman IPYNB (*interactive python notebook*). Adapun perangkat yang di perlukan yaitu.

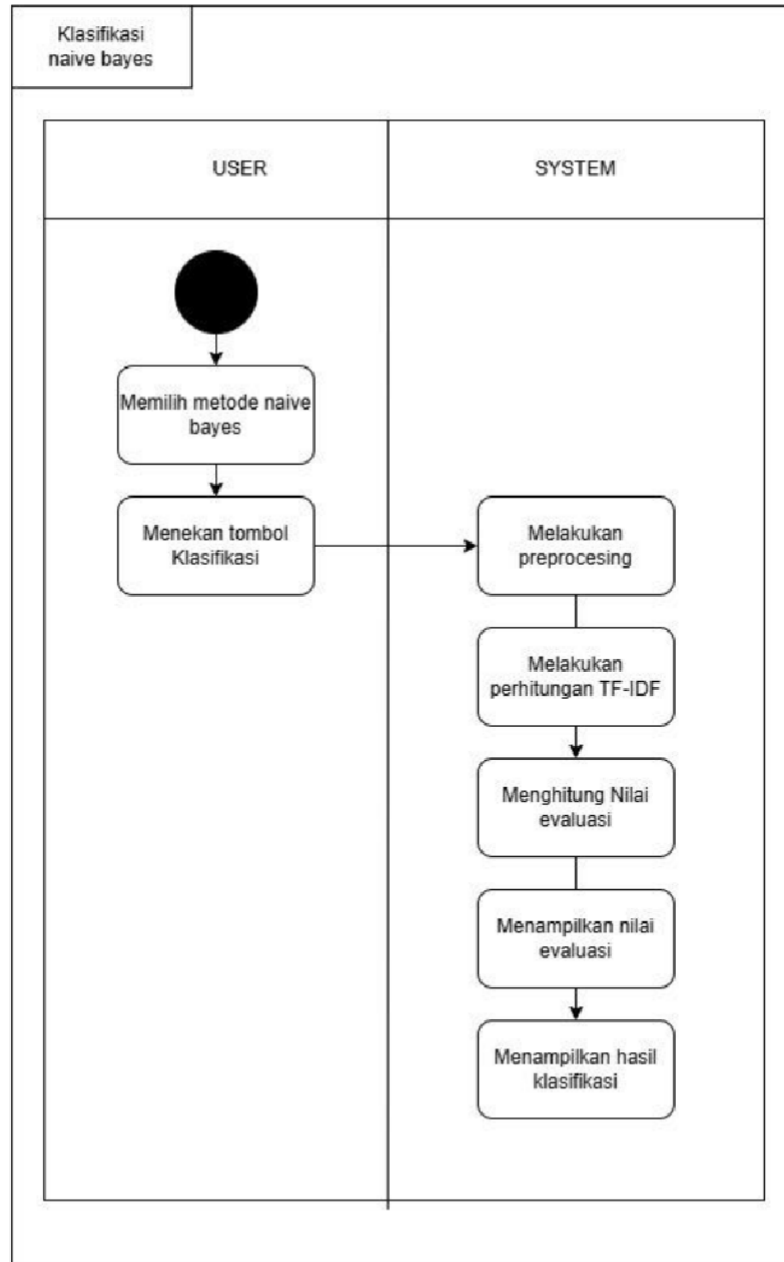
1. Processor : Intel
2. RAM : 8 GB
3. SSD : 512 GB
4. Sistem Operasi : Windows 10 64 bit
5. Teks Editor : Visual Studio Code

4.3.10. Activity Diagram

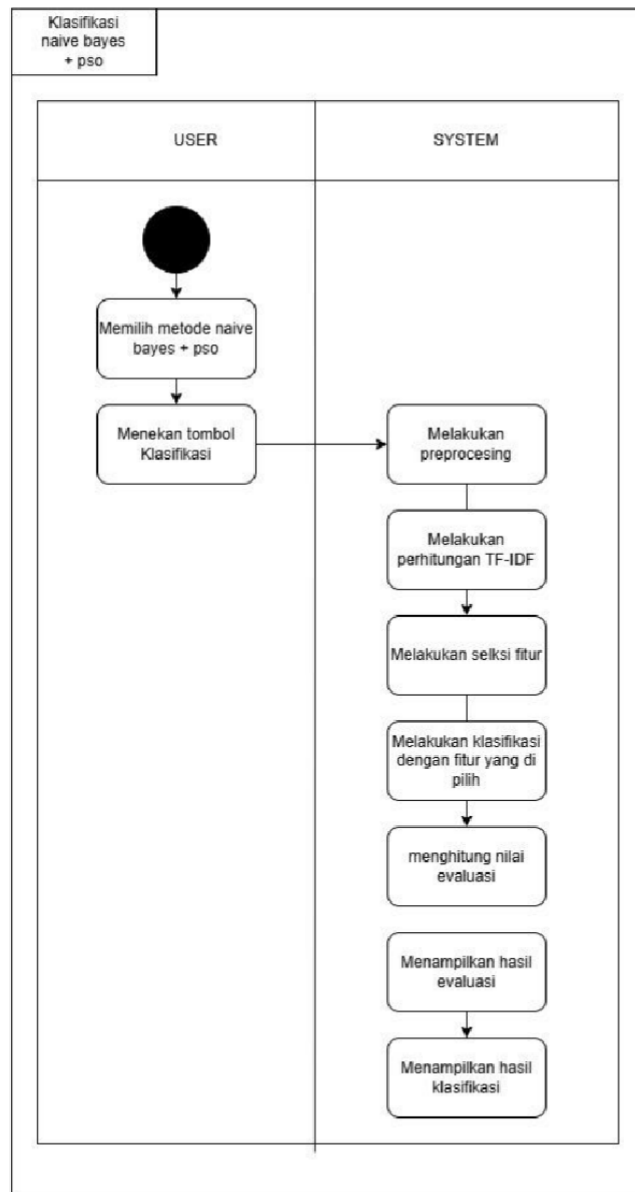
Diagram Aktivitas menampilkan alur aktivitas dalam suatu sistem. Berikut tiga Diagram Aktivitas untuk program yang akan dibangun menggunakan *Use Case* dari bagian sebelumnya



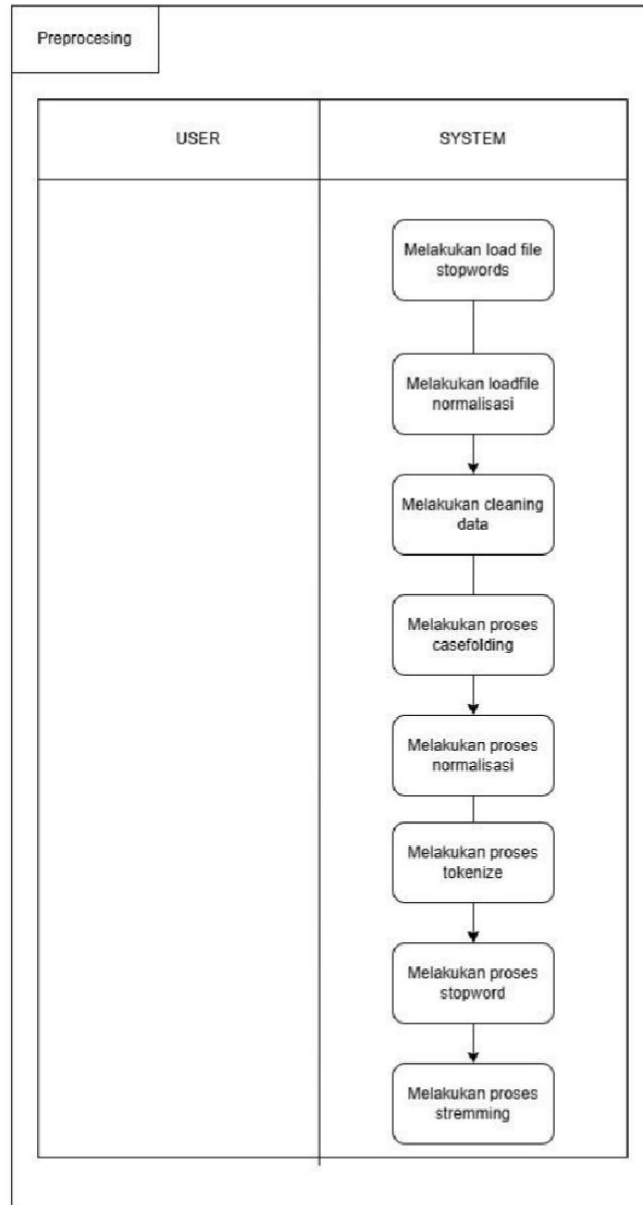
Gambar IV-2. Activity diagram open file



Gambar IV-3. Activity diagram Klasifikasi naive bayes



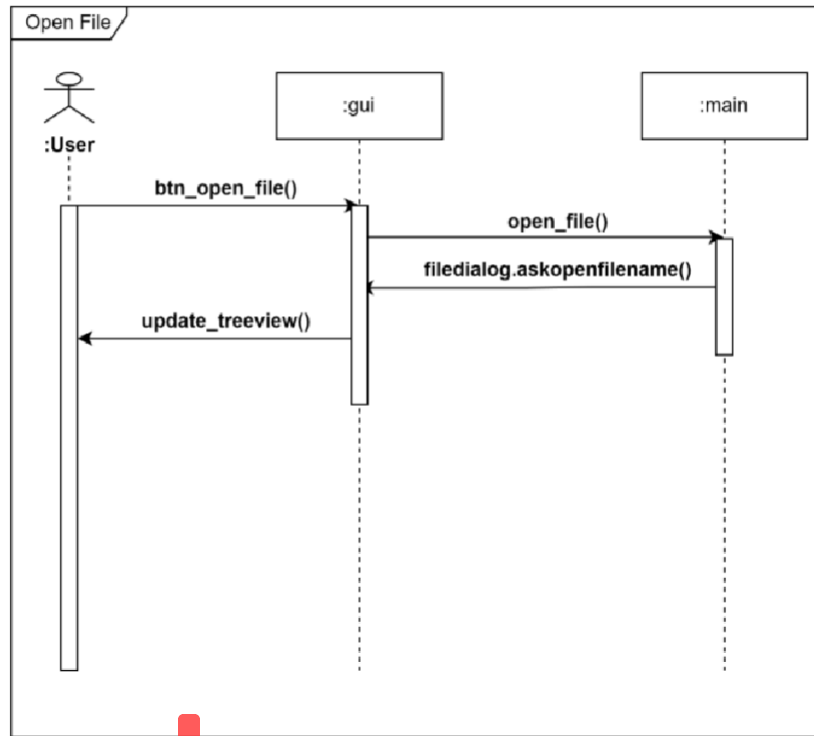
Gambar IV-4. Activity diagram klasifikasi naive bayes + pso



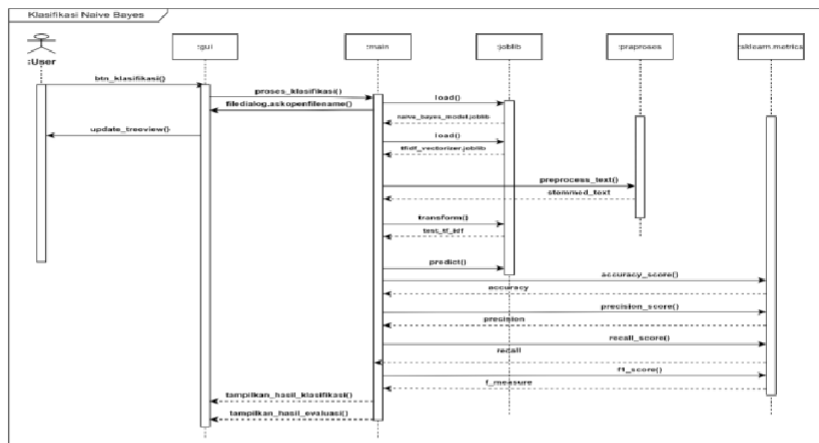
Gambar IV-5. Activity diagram preprocessing

4.3.11. Sequence Diagram

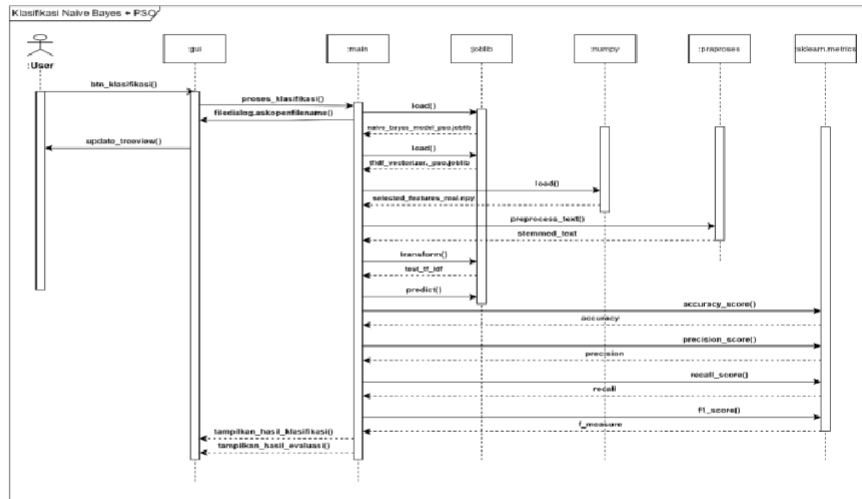
Sequence diagram adalah diagram yang menggambarkan interaksi antarobjek dalam sebuah periode waktu. Berikut adalah Sequence diagram perangkat lunak yang dibuat berdasar use case yang ditentukan dalam subabsebelumnya



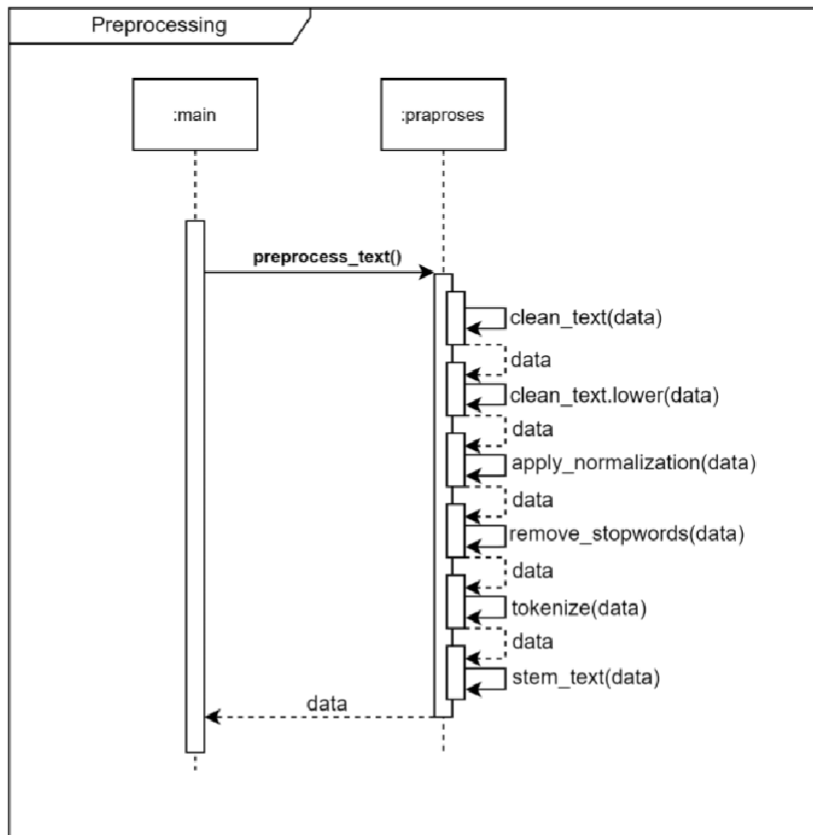
Gambar IV-6. *Sequence Diagram* Open file



Gambar IV-7 *Sequence Diagram* Klasifikasi Naive Bayes



Gambar IV-8. Sequence Diagram Klasifikasi Naive Bayes + pso



Gambar IV-9. Sequence Diagram Preprocessing

4.3.12. Fase Konstruksi

Fase ini, komponen utama dan fungsi pendukung program yang akan dikembangkan akan dibahas.

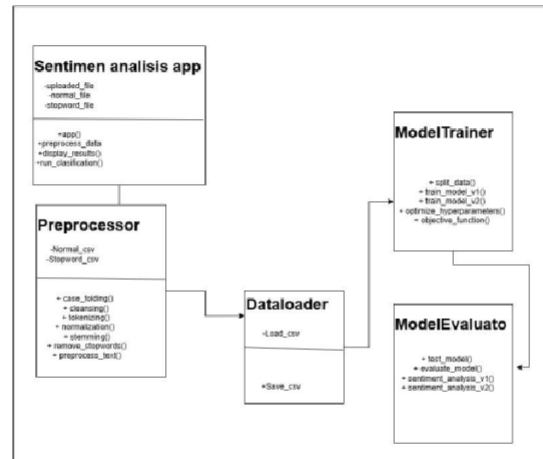
4.3.13. Kebutuhan sistem

Beberapa *library* Python digunakan oleh peneliti saat merancang perangkat lunak ini. *Library Tkinter* dan *sklearn* digunakan dalam penelitian ini. Modul *library sklearn* yang digunakan dalam beberapa kegiatan penelitian ini adalah sebagai berikut.

1. Modul untuk pembobotan TF-IDF yaitu `sklearn.feature_extraction.text.TfidfVectorizer`.
2. Modul untuk klasifikasi Naïve Bayes yaitu `sklearn.naive_bayes.MultinomialNB`.
3. Modul untuk menilai kinerja proses klasifikasi yaitu `sklearn.metrics.confusion_matrix`.

4.3.14. Diagram Kelas

Diagram kelas memberikan gambaran tentang kelas-kelas dalam sistem dan hubungan antar kelas tersebut. Perangkat lunak yang dikembangkan memiliki tiga kategori, yaitu GUI, Main, dan praproses. Representasi diagram kelas dari perangkat lunak



Gambar IV-6. Diagram kelas

4.3.15. Implementasi Kelas

Table IV-32. Table implementasi Kelas

No.	Nama Kelas	Keterangan
1	SentimenAnalysisApp	Mengelola alur utama aplikasi, menangani input pengguna, menampilkan data, dan menjalankan analisis sentimen.
2	Preprocessor	Menangani tugas pra-proses teks seperti case folding, cleansing, tokenizing, normalisasi, stemming, dan penghapusan kata berhenti (stopwords)
3	DataLoader:	Memuat dan menyimpan file CSV.
4	ModelTrainer:	Memisahkan data, melatih model (Naive Bayes dan Naive Bayes dengan PSO), dan mengoptimalkan hiperparameter menggunakan PSO.
5	ModelEvaluato:	Menguji model dan mengevaluasi kinerjanya dengan menghitung akurasi, matriks kebingungan (confusion

	matrix), presisi, recall, dan skor F1.
--	---

4.3.16. Implementasi antarmuka

Pada bagian ini akan di implementasikan berdasarkan Design antarmuka seperti Gambar IV-. Mengambarkan antarmuka halaman utama perangkat lunak.

Analisis Sentimen Aplikasi Satu Sehat Menggunakan Metode Naive Bayes dan Seleksi Fitur PSO

Upload CSV

Drag and drop file here
Limit 200MB per file • CSV

[Browse files](#)

 data1000.csv 68.9KB ×

Data

	full_text	sentimen
0	Bagus	POSITIF
1	Top	POSITIF
2	the best	POSITIF
3	Kagak jelas gblk mau ganti email sama no telpon GK bisa mau daftar LG di tolak gaje	NEGATIF

Gambar IV-7 Antarmuka Perangkat Lunak

PILIH METODE

- Naive Bayes
 Naive Bayes with PSO

Klasifikasi

Confusion Matrix:

0	1
99	1
40	60

Accuracy: 0.795

Precision: 0.8479183866021937

Recall: 0.7949999999999999

F1 Score: 0.7868967488760104

Gambar IV-7. Antarmuka Prangkat Lunak

4.4. Pemodelan Bisnis

Pengujian perangkat lunak dilakukan dengan metode pengujian blackbox. Rencana pengujian disusun berdasarkan kasus penggunaan yang telah diusulkan sejak awal.

5.1. Kebutuhan

Alat yang digunakan untuk pengujian dalam penelitian ini sama denganyang digunakan untuk membangun program. Perangkat yang digunakan adalah sebagai berikut.

1. Processor : AMD Ryzen 3 3250U
2. RAM : 8 GB
3. SSD : 512 SSD
4. Sistem Operasi : Windows 10 64 bit
5. Teks Editor : *Visual Studio Code*

5.2. Analisis dan perancangan

Bagian ini menjelaskan strategi pengujian perangkat lunak yang dikembangkan berdasarkan skenario use case dari langkah sebelumnya

5.3. Rencana pengujian

Beberapa tabel digunakan untuk memuat rencana uji perangkat lunak analisis sentimen Aplikasi SatuSehat. Skenario pengujian perangkat lunak dengan metode blackbox

Table IV-33. Rencana uji Use case proses data

No	Identifikasi	Deskripsi	Prosedur Pengujian
1.	U-1-101	Pra-proses yang data dipilih	Pengujian Unit

Table IV-. Rencana uji use case klasifikasi menggunakan naive bayes

No	Identifikasi	Deskripsi	Prosedur Pengujian
1.	U-2-101	Klasifikasi data Naïve Bayes	Pengujian Unit


Table IV-34. Rencana uji use case klasifikasi menggunakan naive bayes +

No	Identifikasi	Deskripsi	Prosedur Pengujian
1.	U-3-101	Klasifikasi data Naïve Bayes+PSO	Pengujian Unit

PSO

5.4. Implementasi

Fase ini mendefinisikan pengujian yang akan di lakukan berdasarkan strategi pengujian

1.  Pengujian use case proses data

Identifikasi	Deskripsi	Prosedur pengujian	Masukan	Output	Indikator hasil	Hasil yang di peroleh
U-1-101	Open file	Tekan tombol Brows file	File	Data tampil	File tampil dalam program dan dapat di proses oleh program	Program dapat mengimpor file data dan menunjukkan isinya

2. Pengujian Use case Naive bayes dan naive bayes + pso

Identifikasi	Deskripsi	Prosedur pengujian	Masukan	Output	Indikator hasil	Hasil yang di peroleh
U-2-101	Klasifikasi menggunakan naive bayes	Memilih model naive bayes dan menekan tombol Klasifikasi	Radio button algoritma naive bayes di pilih	Tampilan hasil klasifikasi	Menampilkan informasi nilai klasifikasi	Berhasil menampilkan nilai klasifikasi
Identifikasi	Deskripsi	Prosedur Pengujian	Masukan	Output	Indikator hasil	Hasil yang di peroleh
U-3-101	Klasifikasi menggunakan naive bayes + PSO	Memilih model naive bayes+ PSO dan menekan tombol Klasifikasi	Radio button algoritma naive bayes + pso di pilih	Tampilkan hasil klasifikasi	Menampilkan informasi nilai klasifikasi	Berhasil menampilkan nilai klasifikasi

5.5.Kesimpulan

Bab ini menjelaskan secara rinci proses pengembangan perangkat lunak sebagai alat bantu untuk studi analisis sentimen menggunakan metode Naïve Bayes, baik dengan maupun tanpa seleksi fitur PSO. Bab ini juga menguraikan metodologi pemrograman dan skenario pengujian yang digunakan untuk memastikan bahwa perangkat lunak memenuhi spesifikasi penelitian.

BAB V

HASIL DAN ANALISIS PENELITIAN

5.1. Pendahuluan

Bab ini menjelaskan hasil penelitian menggunakan perangkat lunak yang dibuat. Hasil penelitian meliputi *Confusion Matrix*, serta akurasi, presisi, *recall*, *F-measure*, jumlah fitur.

5.2. Data hasil penelitian

Program ini diuji menggunakan 1000 data dari dua kategori: 500 dari kategori positif dan 500 dari kategori negatif. Berdasarkan kerangka penelitian, prosedur pengujian dipisahkan menjadi dua tes: menguji teknik klasifikasi *Naïve Bayes* dengan data tanpa seleksi fitur dan menguji metode klasifikasi *Naïve Bayes* dengan seleksi fitur PSO.

5.3. Data hasil konfigurasi I

Berikut ini adalah data hasil konfigurasi split data untuk metode klasifikasi *Naïve Bayes* dan *Naïve Bayes*.

Table V-1 Data hasil confusion matrix *Naive bayes* dengan split data

Split data	TP	FP	TN	FN
90:10	49	1	18	32
80:20	100	0	29	71
70:30	148	2	43	107

Tabel V-1 menampilkan hasil *Confusion Matrix* dari hasil klasifikasi menggunakan *Naïve Bayes* dengan 3 skenario split data yang berbeda. Selanjutnya akan ditampilkan nilai hasil evaluasi dari *Confusion Matrix* pada tabel V-2.

Table V-2 Hasil Evaluasi Konfigurasi I

Split data	Akurasi	Precision	Recall	F-1
90:10	0.81	0.85	0.81	0.80
80:20	0.855	0.887	0.855	0.851
70:30	0.85	0.87	0.850	0.0848

5.4. Data Hasil Konfigurasi II

Berikut adalah hasil dari Klasifikasi menggunakan metode naive bayes dengan seleksi fitur PSO

Table V-3 Hasil Evaluasi Konfigurasi II

Uji ke	C1	C2	w	TP	FP	TN	FN
1	1	0.4	0.1	55	7	53	5
2	1.2	0.6	0.2	99	1	40	60
3	1.4	0.8	0.3	99	1	40	60
4	1.6	1	0.4	99	1	40	60
5	1.8	1.2	0.5	99	1	40	60

Table V-4 Hasil Evaluasi

Uji ke	C1	C2	w	Akurasi	Presisi	Recall	F1 Score
1	1	0.4	0.1	0.85	0.87	0.85	0.84
2	1.2	0.6	0.2	0.795	0.84	0.79	0.78
3	1.4	0.8	0.3	0.795	0.84	0.79	0.78
4	1.6	1	0.4	0.795	0.84	0.79	0.78
5	1.8	1.2	0.5	0.795	0.84	0.79	0.78

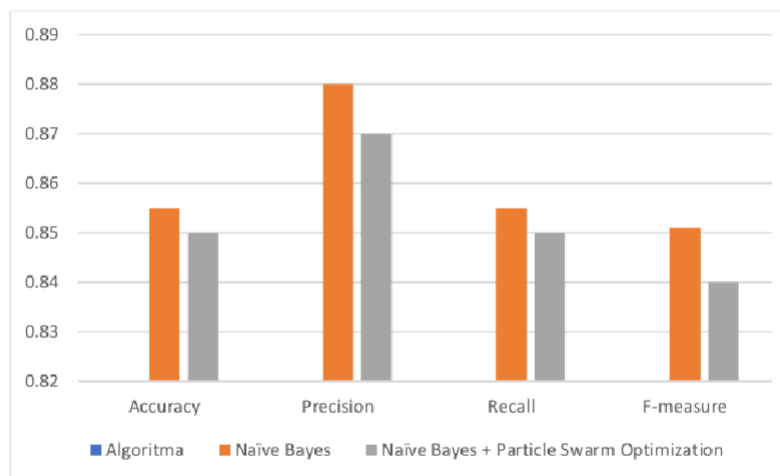
Tabel V-3 menampilkan nilai hasil evaluasi dari perhitungan untuk konfigurasi nilai c_1, c_2 dan w . Hasil evaluasi menunjukkan bahwa untuk konfigurasi dengan nilai $c_1 = 1.0$ dan $c_2 = 0.4$ dan $w = 0.1$ menunjukkan hasil akurasi terbaik sebesar 85% , sehingga konfigurasi tersebut akan digunakan untuk pemodelan aplikasi analisis sentimen aplikasi Satu Sehat menggunakan metode naive bayes

5.5. Analisis Hasil penelitian

Pada Pengujian ini di analisis hasil dengan menggunakan 2 model yang digunakan, yaitu Metode *Naive Bayes* tanpa seleksi fitur dan Naive Bayes dengan seleksi fitur PSO dengan menggunakan konfigurasi parameter PSO terbaik yang di dapatkan dari hasil evaluasi menggunakan *Confusion Matrix* pada subbab sebelumnya. Berikut adalah perbandingan hasil klasifikasi berdasarkan kedua model yang digunakan.

Algoritma	<i>Accuracy</i>	<i>Precision</i>	<i>Recall</i>	<i>F1-Score</i>
Naïve Bayes	85,5%	88%	85,5%	85,1%
Naïve Bayes + Particle Swarm Optimization	85%	87%	85%	84%

Tabel V-8 Menampilkan nilai evaluasi dari kedua model yang digunakan, menurut hasil tersebut. Metode Naïve Bayes dengan seleksi fitur PSO memberikannilai *akurasi, presisi, recall* dan *F-measure* yang lebih rendah.



Gambar V-2. Grafik perbandingan naive bayes dan naive bayes + pso

Hal ini menunjukkan bahwa klasifikasi menggunakan metode Naïve Bayes tanpa seleksi fitur menunjukkan nilai 85,5% sedangkan nilai setelah dilakukan seleksi fitur 85% yang berarti adanya penurunan nilai akurasi setelah dilakukan seleksi fitur PSO sebesar 0,5% , Nilai akurasi ini menunjukkan seberapa baik model dapat mengklasifikasikan data dengan benar secara keseluruhan, sedangkan untuk nilai dari precision menunjukkan penurunan dari 88% menjadi 87% , nilai precision ini merupakan tolak ukur model dalam mengidentifikasi dan memisahkan hasil klasifikasi pada kelas yang benar, pada nilai recall juga terjadi Penurunan sebesar 1% pada saat

dilakukan seleksi fitur PSO, nilai recall sendiri juga berfungsi untuk meminimalisir kesalahan *false negative*, yaitu menghindari mengklasifikasikan contoh negatif sebagai positif. Dapat juga dilihat pada grafik *F-measure* terjadi peningkatan nilai dari 85,5% menjadi 85%, *F-measure* sendiri merupakan gabungan antara *precision* dan *recall*. Nilai *F-measure* yang tinggi menunjukkan bahwa model memiliki presisi dan recall yang baik secara seimbang. Dalam praktiknya, ada keseimbangan antara presisi dan recall. Saat presisi meningkat, recall cenderung menurun, dan sebaliknya. *F-measure* membantu dalam mengevaluasi kinerja model secara komprehensif dengan mempertimbangkan kedua aspek ini. Terutama dalam kasus klasifikasi yang tidak seimbang, *F-measure* dapat memberikan gambaran yang lebih baik tentang kinerja model dibandingkan dengan akurasi yang mungkin terpengaruh oleh distribusi kelas.

5.6. Kesimpulan

Hasil dan analisis pengujian untuk mengkategorikan sentimen menggunakan Naïve Bayes dan Naïve Bayes dengan seleksi fitur PSO telah dijelaskan dalam bab ini. Berdasarkan hasil tersebut, ditentukan bahwa pendekatan Pemilihan Fitur PSO tidak berhasil meningkatkan kinerja dalam klasifikasi sentimen. Bahkan, akurasi menggunakan Naïve Bayes tanpa PSO lebih tinggi dibandingkan dengan menggunakan PSO, menunjukkan bahwa PSO tidak berhasil menemukan subset fitur yang lebih relevan dalam data.

BAB VI

KESIMPULAN DAN SARAN

6.1. Pendahuluan

Bab ini akan menarik kesimpulan dan memberikan saran berdasarkan hasil analisis penelitian bab sebelumnya. Kesimpulan dan saran diberikan sebagai sumber untuk studi lebih lanjut.

6.2. Kesimpulan

Kesimpulan penelitian adalah sebagai berikut.

1. Klasifikasi sentimen Aplikasi Satu Sehat menggunakan metode Naïve Bayes menghasilkan kinerja yang lebih baik dibandingkan dengan menggunakan metode Naïve Bayes dengan seleksi fitur PSO.
2. Seleksi Fitur PSO pada metode Naïve Bayes diharapkan dapat mengurangi fitur yang tidak relevan dalam kumpulan data, namun dalam penelitian ini, metode tersebut tidak berhasil meningkatkan akurasi, presisi, recall, dan nilai F-measure dari proses klasifikasi. Pengujian dengan model Naïve Bayes seleksi fitur PSO menghasilkan nilai akurasi = 85% yang menunjukkan penurunan akurasi sebanyak 0,5% dibandingkan dengan menggunakan model Naïve Bayes tanpa seleksi fitur.
3. Hasil penelitian menunjukkan bahwa penggunaan parameter PSO berpengaruh terhadap kinerja metode seleksi fitur PSO. Parameter terpilih menggunakan $c1 = 1,0$, $c2 = 0,4$, dan $w=0.1$ tidak memberikan kinerja yang lebih baik untuk dataset yang digunakan dalam penelitian ini. Namun, tidak menutup kemungkinan bahwa masih ada konfigurasi parameter yang dapat

memberikan kinerja yang lebih baik lagi.

6.3. Saran

Dalam melakukan penelitian selanjutnya terkait bidang ini, diharapkan agar :

1. **Eksplorasi Parameter PSO:** Disarankan untuk melakukan eksplorasi lebih lanjut terhadap konfigurasi parameter PSO. Variasi nilai c_1 , c_2 , dan w dapat dieksplorasi secara lebih mendalam untuk mencari kombinasi yang mungkin memberikan performa yang lebih baik pada dataset yang digunakan.
2. **Coba Metode Seleksi Fitur Lain:** Selain PSO, ada berbagai metode seleksi fitur lain yang dapat dicoba, seperti Genetic Algorithm (GA), Recursive Feature Elimination (RFE), atau metode statistika seperti Chi-Square dan Information Gain. Mencoba metode-metode ini bisa memberikan wawasan tambahan tentang bagaimana seleksi fitur dapat mempengaruhi performa model.
3. **Evaluasi pada Dataset Berbeda:** Menerapkan metode yang sama pada dataset yang berbeda dapat memberikan gambaran yang lebih lengkap tentang efektivitas PSO dalam seleksi fitur. Mungkin saja PSO bekerja lebih baik pada dataset dengan karakteristik yang berbeda.
4. **Peningkatan Proses Preprocessing:** Tinjau ulang proses preprocessing data untuk memastikan bahwa semua langkah yang relevan sudah diambil. Misalnya, normalisasi atau standardisasi data, penghapusan outliers, atau penggunaan teknik augmentasi data.

DAFTAR PUSTAKA

- Nugraha, Muhammad Tirta, Nina Nina Sulistiyowati, and Ultach Ultach Enri. "ANALISIS SENTIMEN ULASAN APLIKASI SATU SEHAT PADA GOOGLE PLAY STORE MENGGUNAKAN NAÏVE BAYES CLASSIFIER." *JATI (Jurnal Mahasiswa Teknik Informatika)* 7.5 (2023): 3593-3601.
- Lestandy, Merinda, Abdurrahim Abdurrahim, and Lailis Syafa'ah. "Analisis Sentimen Tweet Vaksin COVID-19 Menggunakan Recurrent Neural Network dan Naïve Bayes." *Jurnal RESTI (Rekayasa Sistem Dan Teknologi Informasi)* 5.4 (2021): 802-808.
- Kevin, Kevin, Margareta Enjeli, and Andri Wijaya. "Analisis Sentimen Penggunaan Aplikasi Kinemaster Menggunakan Metode Naive Bayes." *Jurnal Ilmiah Computer Science* 2.2 (2024): 89-98.
- Alsacedi, A., & Khan M., Z. (2019) A Study on Sentiment Analysis Techniques of Twitter Data. *International Journal of Advanced Computer Science and Applications*. 10(2): 361-374. www.ijacsa.thesai.org
- B. Liu. (2012). *Sentiment Analysis and Opinion Mining*, Morgan & Claypool Publisher
- Miller, T. W. (2005). *Data and text mining a business applications approach*
- Wu, X., & Kumar, V. (2009). *The Top Ten Algorithms in Data Mining*. London: Taylor & Francis.
- Tuegeh, M., Soeprijanto, A., & Hery P. M. (2009). Optimal Generator Scheduling based on Particle Swarm Optimization. *Jurnal Seminar Nasional Informatika*. 1(1): 25-32.
- Valle, Y. D., et al. (2008). Particle Swarm Optimization: Basic Concepts, Variants and Applications in Power Systems. *IEEE Transactions on Evolutionary Computation*. 12(2): 171-195. doi: 10.1109/TEVC.2007.896686.

- Wati, D. A. R. (2011). *Sistem Kendali Cerdas Fuzzy Logic Controller (FLC), Jaringan Syaraf Tiruan (JST), Algoritma Genetik (AG) dan Algoritma Particle Swarm Optimization (PSO)*. Yogyakarta: Graha Ilmu.
- Wu, X., & Kumar, V. (2009). *The Top Ten Algorithms in Data Mining*. London: Taylor & Francis.
- Miller, T. W. (2005). *Data and text mining a business applications approach*.
- Muhamad, H., dkk. (2017). Optimasi Naïve Bayes Classifier dengan Menggunakan *Particle Swarm Optimization* pada Data Iris. *Jurnal Teknologi Informasi dan Ilmu Komputer (JTIK)*. 4(3):180-184.
- Nurjannah, M., Hamdani, & Astuti, I. F. (2013). Penerapan Algoritma Term Frequency-Inverse Document Frequency (Tf-Idf) Untuk Text Mining. *Jurnal Informatika Mulawarman*. 8(3): 110–113.
- Prabowo, D. A., dkk. (2016). TF-IDF-Enhanced Genetic Algorithm Untuk Extractive Automatic Text Summarization. *Jurnal Teknologi Informasi dan Ilmu Komputer*. 3(3): 208. doi: 10.25126/jtiik.201633217.
- Prabowo, R., & Thelwall, M. (2009). Sentiment Analysis: A Combined Approach. *Journal of Informetrics*. 3(2): 143–157.
- Prabowo, W. A., & Wiguna, C. (2021). Sistem Informasi UMKM Bengkel dan Web Menggunakan Metode SCRUM. *Jurnal Media Informatika Budidarma*. 5(1): 149-156. doi: 10.30865/mib.v5i1.2604.
- Puspita, R., & Widodo, A. (2021). Perbandingan Metode KNN, Decision Tree dan Naïve Bayes terhadap Analisis Sentimen Pengguna Layanan BPJS. *Jurnal Informatika Universitas Pamulang*. 5(4): 646-654.

Rahman, M. F., Alamsah, D., Darmawidjadja, M. I., & Nurma, I. (2017). Klasifikasi Untuk Diagnosa Diabetes Menggunakan Metode Bayesian Regularization Neural Network (RBNN). *Jurnal Informatika*. 11(1):36. <https://doi.org/10.26555/jifo.v11i1.a5452>

Sari, Retno. (2019). Analisis Sentimen Review Restoran Menggunakan Algoritma Naïve Bayes dan Particle Swarm Optimization. *Jurnal Informatika*. 6(1): 23–28. <http://ejournal.bsi.ac.id/ejurnal/index.php/ji/article/view/4695>

Analisis sentimen aplikasi satu sehat menggunakan metode naive bayes dan particle swarm optimization

ORIGINALITY REPORT

13%

SIMILARITY INDEX

9%

INTERNET SOURCES

8%

PUBLICATIONS

9%

STUDENT PAPERS

PRIMARY SOURCES

1	Submitted to Sriwijaya University Student Paper	3%
2	Yi Li, Yongqi Tan, Yang Pu, Yunying Zhu, Haotian Xie. "Exploring the drivers of green supply chain management in the Chinese electronics industry: Evidence from a GDEMATEL-AISM approach", Cleaner Logistics and Supply Chain, 2023 Publication	2%
3	repository.uin-suska.ac.id Internet Source	2%
4	repository.ub.ac.id Internet Source	1%
5	fedorabg.bg.ac.rs Internet Source	1%
6	journal.uniku.ac.id Internet Source	1%

7

ejournal.almaata.ac.id

Internet Source

%

8	anakbungsu24.blogspot.com Internet Source	1%
9	Ракоњац, Иван М.. "Квантификација Ризика на Пројектима Освајања Индустрijског Производа", University of Belgrade (Serbia), 2024 Publication	1%
10	123dok.com Internet Source	1%
11	Submitted to Singapore Institute of Technology Student Paper	1%
12	dspace.uazuay.edu.ec Internet Source	1%

Exclude quotes On

Exclude bibliography On

Exclude matches < 1%

SURAT KETERANGAN PENGECEKAN SIMILARITY

Saya yang bertanda tangan di bawah ini


Nama : Reza Qumaini
Nim : 09021381924131
Prodi : Teknik Informatika Bilingual

Menyatakan bahwa benar hasil pengecekan similarity Skripsi/Tesis/Disertasi/Sp.
Penelitian yang berjudul "Analisis sentimen aplikasi satu sehat menggunakan metode naive bayes
dan particle swarm optimization" adalah 13%.

Dicek oleh operator *: 1. UPT Perpustakaan

Demikianlah surat keterangan ini saya buat dengan sebenarnya dan dapat saya pertanggung
jawabkan.

Menyetujui
Dosen pembimbing,


Rizki Kurniati, M.T.
NIP. 199107122019032016

Palmbang, 29 Juli 2024

Yang menyatakan,


Reza Qumaini
NIM. 09021381924131