

Peningkatan Performa Klasifikasi Sentimen Tweet Kaesang Menggunakan Naïve Bayes dengan PSO pada Dataset Kecil

Muhammad Ravil, Surya Agustian*, Muhammad Fikry, Fitri Insani

Fakultas Sains dan Teknologi, Teknik Informatika, Universitas Islam Negeri Sultan Syarif Kasim Riau, Pekanbaru, Indonesia

Email: ¹12050111695@students.uin-suska.ac.id, ^{2*}surya.agustian@uin-suska.ac.id, ³muhammad.fikry@uin-suska.ac.id,

⁴fitri.insani@uin-suska.ac.id

Email Penulis Korespondensi: surya.agustian@uin-suska.ac.id

Abstrak—Setelah munculnya berita tentang penunjukan Kaesang sebagai Ketua Umum Partai Solidaritas Indonesia (PSI), berbagai spekulasi muncul di media sosial, khususnya di Twitter (X). Penelitian ini bertujuan untuk melakukan klasifikasi sentimen terhadap berita pengangkatan Kaesang sebagai Ketua Umum PSI menggunakan algoritma Naïve Bayes yang dioptimalkan dengan Particle Swarm Optimization (PSO). Data yang digunakan dalam penelitian ini tweet tentang Kaesang dan tweet terkait COVID-19. Proses text preprocessing mencakup cleaning, case folding, tokenizing, stemming, dan stopword removal. TF-IDF digunakan untuk merepresentasikan kata dalam bentuk vektor. Pada eksperimen awal, Naïve Bayes melakukan klasifikasi dengan menggunakan data Kaesang ditambah dengan data COVID-19, masing-masing sebanyak 300 data untuk setiap label. Particle Swarm Optimization digunakan untuk meningkatkan kinerja algoritma Naïve Bayes. Hasil eksperimen menunjukkan bahwa model yang diuji dengan data uji memperoleh nilai f1-score tertinggi sebesar 50%.

Kata Kunci: Kaesang Pangarep; Klasifikasi Sentimen; PSO; Naïve Bayes; Media Sosial

Abstract—After the news of Kaesang's appointment as the Chairman of the Indonesian Solidarity Party (PSI), various speculations emerged on social media, particularly on Twitter (X). This study aims to classify sentiments regarding Kaesang's appointment as PSI Chairman using the Naïve Bayes algorithm optimized with Particle Swarm Optimization (PSO). The data used in this study consists tweets about Kaesang and tweets related to COVID-19. The text preprocessing process includes cleaning, case folding, tokenizing, stemming, and stopword removal. TF-IDF is used to represent words in vector form. In the initial experiment, Naïve Bayes performed classification using Kaesang data combined with COVID-19 data, with 300 data points for each label. Particle Swarm Optimization was used to improve the performance of the Naïve Bayes algorithm. The experiment results showed that the model tested with test data achieved the highest f1-score of 50%.

Keywords: Kaesang Pangarep; Sentiment Classification; PSO; Naïve Bayes; Social Media

1. PENDAHULUAN

Saat ini, media sosial telah menjadi salah satu sarana paling populer di kalangan masyarakat Indonesia. Media sosial adalah platform berbasis online, baik dalam bentuk situs web maupun aplikasi, yang memungkinkan orang berinteraksi tanpa harus bertemu secara langsung [1]. Twitter adalah media sosial yang berisi kumpulan pesan tentang opini, ekspresi emosi. Manfaat Twitter tidak hanya sebagai media informasi dan komunikasi tetapi juga sebagai media untuk menyuarakan opini publik [2]. Pada tahun 2024, dengan pemilihan umum yang dijadwalkan pada bulan Februari, perhatian terhadap isu politik menjadi sangat signifikan, terutama terkait dengan penunjukan Kaesang sebagai ketua umum PSI. Kaesang Pangarep, putra bungsu Presiden RI Joko Widodo (Jokowi), baru saja diangkat menjadi Ketua Umum Partai Solidaritas Indonesia (PSI). Kaesang diresmikan sebagai Ketua Umum PSI pada acara Kopdarnas PSI di Djakarta Theater, Jakarta Pusat, Senin (25/9/2023). Dalam acara ini, PSI mengumumkan Kaesang Pangarep Ketua Umum PSI [3]. Bagi pihak yang terlibat, pemahaman mengenai sentimen masyarakat terhadap isu tersebut menjadi kunci untuk memetakan citra mereka. Oleh karena itu, perhitungan cepat terhadap arah sentimen positif atau negatif yang terkait dengan pihak mereka menjadi suatu kebutuhan mendesak.

Pentingnya pemahaman sentimen masyarakat melalui media sosial menjadikan analisis sentimen atau *opinion mining* sebagai alat yang sangat diperlukan. Analisis sentimen merupakan proses otomatis untuk memahami, mengekstrak, dan mengolah data tekstual guna mendapatkan informasi mengenai sentimen dalam suatu opini atau kalimat [4]. Dalam konteks Pemilihan Umum, teknologi ini menjadi semakin penting untuk mengukur respons dan pandangan masyarakat terhadap isu politik, khususnya terkait dengan kepemimpinan Kaesang di PSI. Analisis sentimen, yang cenderung berkembang pesat, dapat memberikan wawasan mendalam melalui pengumpulan data dari media sosial, terutama platform seperti Twitter [5].

Particle Swarm Optimization (PSO) merupakan algoritma optimisasi metaheuristik yang terinspirasi dari perilaku sosial kelompok burung atau ikan saat mencari makanan. [6]. PSO adalah metode penelitian populasi yang memanfaatkan populasi individu yang diperbarui di setiap iterasi. Untuk mencapai solusi optimal, setiap partikel bergerak menuju posisi terbaik yang pernah dicapainya (pbest) serta posisi terbaik secara global (gbest). [7]. Naive Bayes Classifier merupakan Metode pengklasifikasian dengan menggunakan metode probabilitas dan statistik yang dikemukakan oleh ilmuwan Inggris Thomas Bayes, yaitu memprediksi peluang di masa depan berdasarkan pengalaman di masa sebelumnya sehingga dikenal sebagai Teorema Bayes [8]. Salah satu contoh ciri metode ini yaitu dimana asumsi kondisi sangat data kuat. Dimana algoritma ini memanfaatkan teori probabilitas yang bisa memprediksi masa depan berdasarkan data latih sebelumnya [9]. Naive Bayes adalah metode klasifikasi yang memanfaatkan teorema Bayes dengan asumsi bahwa fitur-fitur yang digunakan dalam proses klasifikasi bersifat independen satu sama lain. Meskipun asumsi ini seringkali tidak realistis dalam praktik, Naive Bayes tetap menjadi salah satu metode klasifikasi yang populer dan efektif [10].

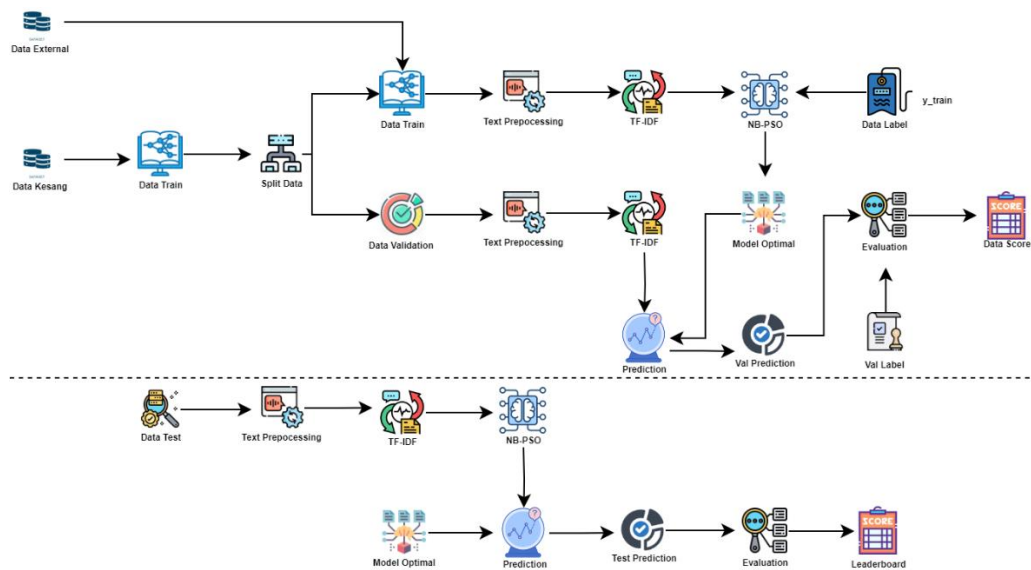
Beberapa penelitian sebelumnya telah mengkaji analisis sentimen.[11] melakukan Implementasi Algoritma Multinomial Naïve Bayes Untuk Klasifikasi Sentimen Pemerintah Terhadap Penanganan Covid-19, hasil pengujian diperoleh bahwa weighted average untuk precision, recall dan akurasi sebesar 74%. Penelitian menunjukkan bahwa akurasi metode yang diusulkan Ini memiliki tingkatan cukup baik. [12] menganalisis Sentimen Pengguna Twitter Terhadap BTS Menggunakan Algoritma Support Vector, dengan akurasi 90% pada kedua skenario penelitian. Selanjutnya, [13] menerapkan Metode SVM dan Naive Bayes untuk Analisis Sentimen ChatGPT di Twitter, dengan hasil akurasi terbesar mencapai 59%. Dapat dilihat bahwa ketiga penelitian tersebut mendapatkan skor tertinggi untuk algoritma Naive Bayes. Selain itu, beberapa penelitian menggunakan Particle Swarm Optimization (PSO). [1] melakukan Analisis Sentimen Media Sosial Twitter Terhadap RUU Omnibus Law dengan Metode Naïve Bayes dan Particle Swarm Optimization, yang meningkatkan akurasi dari 85% menjadi 91%. Begitu pula dengan penelitian [7] yang menganalisis Sentimen Review Produk Skincare Dengan Naïve Bayes Classifier Berbasis Particle Swarm Optimization (PSO), yang menghasilkan peningkatan akurasi dari 77.96% menjadi 79.85%. Peningkatan akurasi ini mencapai 1.89%.

Sebuah shared task bersama tentang klasifikasi sentimen diuraikan dalam [14], memperkenalkan tantangan klasifikasi dengan jumlah data pelatihan yang terbatas. Dataset pelatihan yang tersedia untuk mengembangkan model pembelajaran mesin hanya mencakup 300 tweet, yang dianggap tidak memadai untuk menghasilkan hasil yang optimal. Dengan demikian, penelitian ini bertujuan untuk menganalisis sentimen masyarakat terkait isu kontroversial pengangkatan Kaesang sebagai Ketua PSI menggunakan pendekatan algoritma Naïve Bayes Classifier yang telah dioptimalkan dengan Particle Swarm Optimization (PSO) di platform media sosial Twitter. Data yang diperoleh berasal dari kumpulan tweet, yang kemudian dianalisis dan diklasifikasikan ke dalam kategori positif, negatif, atau netral terkait peristiwa tersebut. Proses analisis melibatkan penggunaan keyword yang relevan dengan isu pengangkatan Kaesang, dan data kemudian diproses menggunakan metode Naïve Bayes Classifier dan Particle Swarm Optimization.

2. METODOLOGI PENELITIAN

2.1 Tahapan Penelitian

Tahapan penelitian menjelaskan tahap-tahap yang akan dilakukan untuk menyelesaikan penelitian, adapun tahapannya bisa dilihat pada gambar 1 dibawah.



Gambar 1. Metodologi Penelitian

Penelitian ini mencakup beberapa langkah, seperti Dataset, Text preprocessing, TF-IDF, penerapan klasifikasi menggunakan Naïve Bayes dan Particle Swarm Optimization, dilanjutkan dengan tahap evaluasi menggunakan Confussion Matrix. Penjelasan terkait tahapan penelitian telah dijelaskan pada subbab berikutnya.

2.3 Dataset

Penelitian ini menggunakan dataset yang dikumpulkan dengan dua topik yang berbeda. Topik pertama adalah dataset sentimen pengangkatan kaesang menjadi ketua partai PSI (Data Kaesang). Topik kedua adalah dataset sentimen terkait program vaksin Covid-19 (Data Covid). Penelitian ini terutama berfokus pada data Kaesang, yang dikumpulkan dengan tujuan khusus untuk mengeksplorasi bagaimana penggunaan data yang terbatas dapat memengaruhi pelatihan machine learning.

Dataset Kaesang diambil untuk penelitian ini sebagai bagian dari sistem shared task, dengan tujuan untuk mempelajari penggunaan data terbatas dalam pelatihan machine learning. Data dikumpulkan melalui crawling Twitter

(X) dari tanggal 25 September 2023 hingga 03 Oktober 2023 dengan menggunakan kata kunci "Kaesang PSI". Sentimen pada setiap tweet diberi label melalui crowdsourcing, dengan minimal empat anotator per tweet. Label positif, netral, atau negatif ditentukan berdasarkan suara mayoritas. Tweet tanpa label dominan dihapus dan dianggap tidak valid.

Data training terdiri dari dua dataset, masing-masing 300 tweet yang diberi label positif, negatif, dan netral. Dataset pelatihan pertama disebut Kaesang V1 dan yang kedua Kaesang V2. Untuk pengujian model, tersedia 924 tweet dengan label gold standard yang tersimpan di server leaderboard. Model terbaik dari penelitian ini akan diuji pada data pengujian tersebut, dan hasil prediksinya akan dikirimkan ke sistem leaderboard untuk evaluasi skor [14].

Data Covid atau data eksternal yang digunakan telah dikumpulkan oleh penelitian sebelumnya [15],[16]. Data ini terdiri dari 8000 tweet yang telah diberi label positif, negatif, dan netral. Data Covid akan digunakan sebagai tambahan dataset pelatihan untuk data Kaesang. Pembagian semua dataset bisa dilihat pada tabel 1 dibawah.

Tabel 1. Jumlah dataset

Dataset	Jumlah Tweet
Train Kaesang V1	300
Train Kaesang V2	300
Data Test Kaesang	924
Data Covid	8000

2.4 Text Preprocessing

Pemrosesan teks adalah proses pembersihan dan transformasi data teks mentah sebelum digunakan dalam tugas pemrosesan bahasa alami seperti klasifikasi atau klustering. Ini melibatkan tugas-tugas seperti cleaning, menangani karakter khusus, mengubah teks menjadi huruf kecil, serta stemming dan stopword [17].

- Cleaning: Cleaning atau pembersihan adalah proses awal dalam pengolahan data teks yang bertujuan untuk membersihkan data dari unsur-unsur yang tidak relevan atau kotor.
- Tokenisasi: Sebuah dokumen teks terdiri dari serangkaian kalimat, proses tokenisasi memecah dokumen menjadi bagian-bagian kata yang disebut token [18].
- Case Folding: Dalam prapemrosesan teks, proses case folding bertujuan untuk mengubah semua huruf dalam dokumen teks menjadi huruf kecil [18].
- Stopword: Tujuan penghapusan kata adalah untuk menghilangkan kata-kata yang tidak penting pada dokumen [18].
- Stemming: Stemming adalah proses memetakan dan memecah bentuk suatu kata menjadi bentuk dasarnya. Lebih mudahnya, proses mengubah kata yang memiliki imbuhan menjadi kata dasar [18].

2.5 TF-IDF

Algoritma TF-IDF merupakan penilaian komprehensif terhadap pentingnya sebuah kata untuk sebuah teks atau kelas teks. TF (frekuensi kata) adalah frekuensi kemunculan kata dalam artikel atau kelas dokumen tersebut secara intuitif menunjukkan pentingnya kata tersebut untuk artikel atau kelas dokumen tersebut; IDF (dokumen terbalik frekuensi) mencirikan kemampuan kata untuk membedakan terhadap klasifikasi teks [19].

$$TF_{ij} = \frac{n_{ij}}{n_{total}} \quad (1)$$

$$IDF_i = \log\left(\frac{N}{df_i}\right) \quad (2)$$

Keterangan

n_{ij} = jumlah kemunculan kata i dalam dokumen j

n_{total} = total kata dalam dokumen j

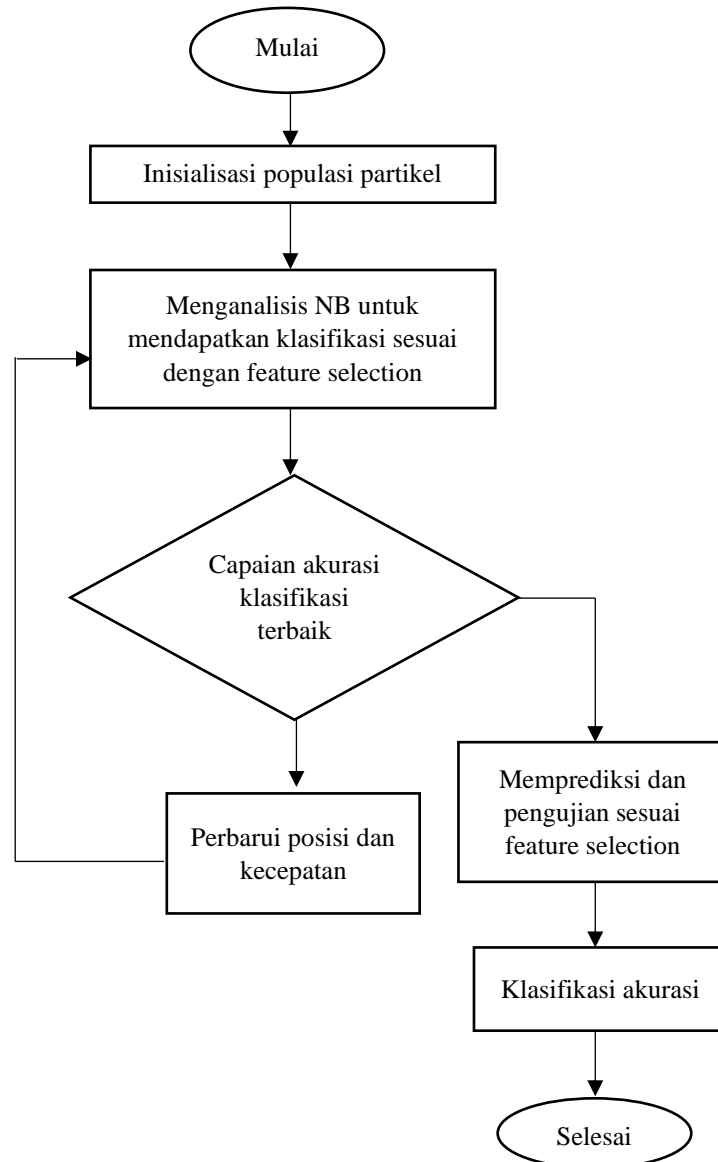
N = jumlah total dokumen dalam korpus

df_i = jumlah dokumen yang mengandung kata i

2.6 Klasifikasi NB-PSO

Untuk proses klasifikasi, penelitian ini menggunakan algoritma Naïve Bayes (NB) yang dikombinasikan dengan Particle Swarm Optimization (PSO) untuk mengoptimalkan hasil. Naive Bayes Classifier merupakan Metode pengklasifikasian dengan menggunakan metode probabilitas dan statistik yang dikemukakan oleh ilmuwan Inggris Thomas Bayes, yaitu memprediksi peluang di masa depan berdasarkan pengalaman di masa sebelumnya sehingga dikenal sebagai Teorema Bayes [8].

Dalam penelitian ini, algoritma Naïve Bayes akan memperoleh akurasi klasifikasi sesuai dengan fitur yang dipilih berdasarkan Particle Swarm Optimization (PSO). Particle Swarm Optimization (PSO) merupakan metode optimasi yang digunakan untuk meningkatkan akurasi hasil dengan mengontrol pemilihan subset. PSO, sebagai metode penelitian populasi yang terinspirasi dari perilaku kelompok hewan mencari makanan, telah menarik minat banyak peneliti di berbagai bidang karena sifatnya yang sederhana dan efektif dalam mengoptimalkan hasil. Dengan cara ini, PSO memberikan kontribusi penting dalam mengatasi permasalahan optimasi dalam berbagai konteks penelitian [20].



Gambar 2. Flowchart NB-PSO

Proses dimulai dengan inisialisasi populasi partikel, di mana partikel-partikel dibuat dengan posisi dan kecepatan acak. Setiap partikel mewakili satu set fitur yang dipilih, dan Naïve Bayes (NB) digunakan untuk mengklasifikasikan data berdasarkan fitur-fitur yang dipilih oleh masing-masing partikel.

Setiap partikel kemudian dievaluasi menggunakan algoritma Naïve Bayes (NB) untuk mengklasifikasikan data berdasarkan fitur-fitur yang dipilih oleh masing-masing partikel. Akurasi klasifikasi yang dicapai oleh setiap partikel dievaluasi. Jika akurasi yang dicapai oleh partikel tersebut adalah yang terbaik sejauh ini, nilai tersebut disimpan sebagai akurasi terbaik pribadi partikel (pBest). Selain itu, jika akurasi tersebut juga merupakan yang terbaik dibandingkan seluruh partikel dalam populasi, nilai tersebut disimpan sebagai akurasi terbaik global (gBest).

Setelah evaluasi, posisi dan kecepatan setiap partikel diperbarui menggunakan persamaan PSO, yang didasarkan pada pengalaman pribadi terbaik (pBest) dan pengalaman global terbaik (gBest). Rumus update kecepatan partikel dapat dilihat pada persamaan rumus (3) dibawah ini.

$$v_i(t + 1) = wv_i(t) + c_1r_1(pBest_i - x_i(t)) + c_2c_2(gBest_1 - x_i(t)) \quad (3)$$

Dan untuk rumus pembaruan posisi dapat dilihat pada persamaan rumus (4) dibawah ini.

$$x_i(t + 1) = x_i(t) + v_i(t + 1) \quad (4)$$

Setelah beberapa iterasi atau kondisi tertentu terpenuhi, dilakukan prediksi dan pengujian akhir menggunakan set fitur yang dipilih oleh partikel terbaik. Akhirnya, akurasi klasifikasi akhir dievaluasi dan dilaporkan. Proses ini selesai ketika seluruh langkah telah dilaksanakan. Dengan demikian, PSO digunakan untuk mencari kombinasi fitur yang optimal dalam meningkatkan kinerja klasifikasi Naïve Bayes, melalui serangkaian evaluasi dan pembaruan posisi serta kecepatan partikel.

2.7 Evaluasi

Pada evaluasi, model Naive Bayes akan dievaluasi setelah mendapatkan kombinasi optimal dari fitur-fitur dan parameter. Confusion Matrix digunakan untuk menunjukkan hasil klasifikasi dengan jumlah data yang diklasifikasikan secara tepat atau tidak tepat. Umumnya, Confusion Matrix digunakan untuk mengukur akurasi, presisi, dan recall. Namun, dalam penelitian ini, evaluasi model akan berdasarkan nilai F1-score, yang menjadi skor resmi dalam shared task ini. Persamaan F1-score dapat dilihat pada persamaan (5) dibawah.

$$F_1 = 2 \times \frac{\text{precision} \times \text{recall}}{\text{precision} + \text{recall}} \quad (5)$$

3. HASIL DAN PEMBAHASAN

3.1 Dataset

Data tweet Kaesang yang didapat melalui proses crawling di Twitter telah melalui tahap seleksi. Total data yang berhasil didapat sebanyak 1.524 tweet. Data ini kemudian dibagi menjadi dua bagian, yaitu 300 tweet sebagai data Train dan 924 tweet sebagai data Test. Pembagian ini dilakukan untuk keperluan pelatihan dan pengujian model, di mana data Test tidak pernah digunakan selama proses pelatihan. Selanjutnya, untuk menemukan model yang optimal, data Train yang terdiri dari 300 tweet akan dibagi lagi dengan proporsi 80% untuk pelatihan dan 20% untuk validasi. Contoh dataset yang digunakan dapat dilihat pada tabel 2 dibawah.

Tabel 2. Contoh isi dataset

No	Tweet	Label
1	@emerson_yuntho @psi_id Gagalnya dimana. Kaesang emang baru terdaftar di psi, tapi jauh sebelum itu kaesang udah sama sama belajar di PSI, dan kaesang diberi mandat hasil rapat para ketua DPW se Indonesia. Beda halnya dengan AHY walaupun dia lama di PD tapi cara dia menjadi ketum sistem tunjuk.	Positif
2	@bobby_risakotta @Cerdas007Cermat @jokowi @ganjarpranowo Cuman tunggu waktu saja Kaesang tarik kembali PSI dukung GP. Ngak juga gpp sih? Ngak ngaru.	Netral
3	@satriohendri Pegemar Anis ga suka kalau PSI & Kaesang maju.	Negatif

3.2 Text Preprocessing

Text preprocessing adalah suatu proses untuk menyeleksi data text agar menjadi lebih terstruktur lagi dengan melalui serangkaian tahapan. Langkah ini bertujuan untuk membersihkan teks pada dataset tweet agar siap untuk tahap selanjutnya. Pada langkah ini text preprocessing yang dilakukan adalah cleaning, case folding, tokenisasi, stopword removal dan stemming. text preprocessing ini dilakukan guna untuk meningkatkan hasil akurasi dari model. Hasil dari text preprocessing tersebut bisa dilihat pada tabel 3.

Tabel 3. Hasil proses text preprocessing

No	Proses	Sebelum Proses	Setelah proses
1	Cleaning	@CNNIndonesia Saya suka gaya anak mudah ini (kaesang) PSI selalu di hati	Saya suka gaya anak mudah ini kaesang PSI selalu di hati
2	Case Folding	Saya suka gaya anak mudah ini kaesang PSI selalu di hati	saya suka gaya anak mudah ini kaesang psi selalu di hati
3	Tokenizing	saya suka gaya anak mudah ini kaesang psi selalu di hati	["saya", "suka", "gaya", "anak", "mudah", "ini", "kaesang", "psi", "selalu", "di", "hati"]
4	Stopword	"saya", "suka", "gaya", "anak", "mudah", "ini", "kaesang", "psi", "selalu", "di", "hati"	["suka", "gaya", "anak", "mudah", "kaesang", "psi", "hati"]
5	Stemming	"suka", "gaya", "anak", "mudah", "kaesang", "psi", "hati"	["suka", "gaya", "anak", "mudah", "kaesang", "psi", "hati"]

3.3 TF-IDF

TF-IDF (Term Frequency-Inverse Document Frequency) adalah metode yang digunakan untuk mengukur seberapa penting suatu kata dalam sebuah dokumen dalam kumpulan dokumen. Dalam penelitian ini, peneliti menggunakan TfidfVectorizer dari library scikit-learn untuk menghasilkan vektor dari teks yang telah melalui tahap preprocessing sebelumnya. Tahap preprocessing ini mencakup langkah-langkah seperti cleaning text, case folding, tokenisasi, stopword removal dan stemming. Bebrapa hasil dari vektorisasi kata bisa dilihat pada gambar 3 dibawah.

	feature	tfidf
0	lyke	0.630259
1	be	0.595474
2	masuk	0.375427
3	psi	0.242334
4	kaesang	0.220257

Gambar 3. Hasil vektorisasi data

3.4 Klasifikasi NB-PSO

Pada tahap ini, peneliti menggunakan algoritma Naïve Bayes yang dioptimalkan oleh algoritma Particle Swarm Optimization (PSO). Kombinasi NB dan PSO digunakan untuk meningkatkan akurasi klasifikasi dalam analisis sentimen.

Untuk mengetahui kinerja sistem, peneliti terlebih dahulu mencari model terbaik pada algoritma Naïve Bayes tanpa menggunakan optimasi Particle Swarm Optimization (PSO). Dalam eksperimen awal ini, peneliti menggunakan data latih Kaesang V1 dan ditambahkan dengan data sentimen covid-19 sebanyak 300 data per label. Untuk mengevaluasi performa model, peneliti menerapkan metode 10-fold cross-validation. Hasil dari eksperimen ini disajikan dalam Tabel 4 di bawah. Berdasarkan hasil yang diperoleh, validasi kelima menunjukkan F1-score tertinggi sebesar 63.15%, sementara validasi kesembilan menghasilkan F1-score terendah sebesar 42.98%. Hal ini menunjukkan adanya variasi kinerja model pada berbagai fold validasi, yang menekankan pentingnya optimasi lebih lanjut untuk mencapai konsistensi dan performa yang lebih tinggi.

Tabel 4. Hasil f1-score dengan 10-fold cross validation

Eksperimen	F1-Score
1	42.98%
2	52.63%
3	56.14%
4	58.77%
5	63.15%
6	59.64%
7	58.77%
8	63.15%
9	57.89%
10	57.01%

Dalam eksperimen kedua, peneliti memperkenalkan optimasi menggunakan algoritma Particle Swarm Optimization (PSO) pada model Naïve Bayes terbaik yang telah diidentifikasi dari eksperimen pertama. Eksperimen ini difokuskan pada eksplorasi parameter PSO, dengan jumlah partikel yang bervariasi dari 1 hingga 10. Berat inersia (w) dalam eksperimen ini ditetapkan sebesar 0.1, dan jumlah iterasi yang digunakan adalah 30. Hasil dari eksperimen kedua ini dapat dilihat pada Tabel 5 di bawah. Dari hasil tersebut, terlihat bahwa partikel ke-9 menghasilkan nilai F1-score tertinggi, yaitu 68.28%. Penemuan ini menunjukkan bahwa pemilihan jumlah partikel yang tepat dapat berdampak signifikan terhadap kinerja model.

Tabel 5. Hasil eksperimen f1-score pada partikel

Partikel	F1-Score
1	56.63%
2	59.62%
3	61.16%
4	61.20%
5	62.79%
6	62.79%
7	62.92%
8	65.07%
9	68.28%
10	62.79%

Pada eksperimen ketiga, peneliti fokus pada eksplorasi nilai berat inersia (w) dalam algoritma PSO, dengan nilai yang bervariasi dari 0.1 hingga 1.0. Dalam eksperimen ini, jumlah partikel ditetapkan sebanyak 9, karena partikel ke-9 telah menunjukkan performa terbaik pada eksperimen sebelumnya. Hasil dari eksperimen ini juga disajikan dalam Tabel 6 di bawah. Dari hasil tersebut, terlihat bahwa nilai berat inersia memiliki pengaruh yang signifikan terhadap F1-score yang dihasilkan. Nilai F1-score terbaik diperoleh pada berat inersia sebesar 0.9, dengan F1-score mencapai 72.96%. Temuan ini menegaskan bahwa selain jumlah partikel, pengaturan parameter berat inersia juga sangat penting dalam optimasi PSO untuk mencapai performa model yang optimal.

Tabel 6. Hasil eksperimen f1-score pada berat inersia

Inersia	F1-Score
0.1	68.28%
0.2	61.92%
0.3	65.82%
0.4	64.68%
0.5	62.18%
0.6	62.80%
0.7	65.18%
0.8	64.24%
0.9	72.96%
1.0	69.79%

Untuk eksperimen terakhir, peneliti melakukan perubahan pada jumlah iterasi. Dalam eksperimen ini, peneliti menggunakan partikel ke-9 dan nilai berat inersia (w) sebesar 0.9 sesuai hasil dari eksperimen kedua dan ketiga. Jumlah iterasi yang digunakan berkisar antara 30 hingga 120. Hasilnya dapat dilihat pada Tabel 7 di bawah. Tabel tersebut menunjukkan bahwa, meskipun terjadi penambahan jumlah iterasi, tidak ada perubahan signifikan yang teramati pada hasil akurasi yang diukur menggunakan f1-score. Hal ini mengindikasikan bahwa setelah titik tertentu, peningkatan jumlah iterasi tidak memberikan keuntungan tambahan dalam hal akurasi model.

Tabel 7. Hasil eksperimen f1-score pada jumlah iterasi

Iterasi	F1-Score
30	72.96%
40	72.96%
50	72.96%
60	72.96%
70	72.96%
80	72.96%
90	72.96%
100	72.96%
110	72.96%
120	72.96%

3.5 Confusion Matrix

Hasil akurasi pengujian akan direpresentasikan dengan tabel matriks menggunakan Confusion Matrix. Dengan menggunakan matriks ini, dapat menghindari hasil yang tidak valid dari proses klasifikasi. Berdasarkan hasil evaluasi yang ditampilkan pada gambar Confusion Matrix, kita bisa melihat distribusi prediksi model setelah dioptimasi menggunakan PSO (Particle Swarm Optimization).

Tabel 8. Tabel confusion matrix

Eskperimen	F1-Score (%)	Akurasi (%)	Presisi (%)	Recall (%)
Naïve Bayes	52.70	55.00	57.50	55.00
Naïve Bayes+PSO	72.96	73.33	74.79	73.33

Dapat dilihat pada tabel 8 bahwa PSO dapat meningkatkan hasil klasifikasi dari Naïve Bayes. Hasil akurasi menunjukkan bahwa nilai F1-score dari Naïve Bayes memiliki akurasi sekitar 52,70%, sedangkan setelah dioptimasi oleh PSO, nilai F1-score meningkat menjadi 72,96%. Jadi, dapat disimpulkan bahwa pada kasus klasifikasi sentimen, optimasi Naïve Bayes dengan PSO dapat memberikan peningkatan kinerja yang signifikan.

3.6 Pengujian Menggunakan Data Uji

Model optimal yang telah ditemukan sebelumnya akan diuji dengan data uji yang belum pernah dilihat oleh model sebelumnya. Hasilnya bisa dilihat pada tabel 9 dibawah. Dari tabel di bawah terlihat bahwa kombinasi dataset Kaesang V1 + V2 + Covid (900) tanpa stopword dan stemming memberikan hasil terbaik dengan F1-Score sebesar 50.59%, akurasi 59.74%, presisi 52.54%, dan recall 58.83%.

Tabel 9. Evaluasi terhadap data uji

Dataset	Stopword dan stemming	F1-Score (%)	Akurasi (%)	Presisi (%)	Recall (%)
Kaesang V1 + Covid (900)	Ya	40.51	48.00	41.64	47.20
Kaesang V1 + V2 + Covid (900)	Ya	44.79	52.87	46.27	51.98
Kaesang V1+ V2 + Covid (900)	Tidak	50.59	59.74	52.54	58.83

3.7 Perbandingan Pengujian

Untuk perbandingan dengan metode lainnya, dapat dilihat pada Tabel 10 di bawah. Perbandingan ini menunjukkan bahwa metode SVM + Word2Vec dari tim Rank 1 sedikit lebih unggul dibandingkan metode NB + PSO yang digunakan dalam penelitian ini. Tim Rank 1 mencapai F1-Score yang lebih tinggi dan akurasi yang lebih tinggi dengan presisi dan recall juga sedikit lebih tinggi. Namun, metode NB + PSO dalam penelitian ini tetap menunjukkan hasil yang kompetitif dengan perbedaan yang tidak terlalu besar, menunjukkan bahwa NB + PSO merupakan pendekatan yang cukup efektif dalam klasifikasi data meskipun tidak mencapai performa tertinggi seperti metode SVM + Word2Vec.

Tabel 10. Perbandingan hasil pengujian

Tim	Metode	F1-Score(%)	Akurasi(%)	Presisi(%)	Recall(%)
Rank 1	SVM + Word2Vec	54.23	63.81	53.72	59.70
Organizer	SVM + TF - IDF	51.28	61.21	52.89	59.95
Admin	Baseline	40.38	45.45	49.53	48.80
Penelitian ini	NB + PSO	50.59	59.74	52.54	58.83

4. KESIMPULAN

Penelitian ini mengimplementasikan metode Naïve Bayes dan Particle Swarm Optimization (PSO) untuk klasifikasi sentimen terkait pengangkatan Kaesang sebagai ketua umum Partai Solidaritas Indonesia (PSI), menggunakan dataset tweet yang dikumpulkan dari Twitter antara 25 September 2023 hingga 03 Oktober 2023, yang diberi label negatif, positif, atau netral. Untuk mengatasi keterbatasan data, digunakan data tambahan dari penelitian sebelumnya mengenai sentimen vaksin COVID-19, dengan 8.000 tweet berlabel serupa. Dari 300 tweet pelatihan Kaesang, data dibagi menjadi 80% untuk pelatihan dan 20% untuk validasi, dan ditambah 300 tweet vaksin COVID-19 per label. Model Naïve Bayes awal menghasilkan f1-score 52,70%, namun setelah mengaplikasikan PSO untuk optimasi parameter, f1-score meningkat signifikan menjadi 72,96%. Uji akhir dengan data yang belum pernah dilihat sebelumnya menghasilkan f1-score tertinggi 50%. Hasil eksperimen menunjukkan bahwa PSO secara signifikan meningkatkan performa model, membuktikan efektivitasnya dalam optimasi parameter, terutama dalam kondisi keterbatasan data pelatihan.

REFERENCES

- [1] S.A. Dainamang, N. Hayatin, D.R. Chandranegara, Analisis Sentimen Media Sosial Twitter terhadap RUU Omnibus Law dengan Metode Naive Bayes dan Particle Swarm Optimization, *Komputika J. Sist. Komput.* 11 (2022) 211–218. <https://doi.org/10.34010/komputika.v11i2.6037>.
- [2] N. Hayatin, G.I. Marthasari, L. Nuraini, Optimization of Sentiment Analysis for Indonesian Presidential Election using Naïve Bayes and Particle Swarm Optimization, *J. Online Inform.* 5 (2020) 81–88. <https://doi.org/10.15575/join.v5i1.558>.
- [3] Emir Yanwardhana, Kaesang Jadi Ketum PSI Bikin Heboh! Parpol Buka Suara, *CNBC Indones.* (2023). <https://www.cnbcindonesia.com/news/20230926142517-4-475661/kaesang-jadi-ketum-psi-bikin-heboh-parpol-buka-suara> (accessed April 23, 2024).
- [4] M.K. Anam, B.N. Pikir, M.B. Firdaus, Penerapan Naïve Bayes Classifier, K-Nearest Neighbor (KNN) dan Decision Tree untuk Menganalisis Sentimen pada Interaksi Netizen danPemerintah, *MATRIK J. Manajemen, Tek. Inform. Dan Rekayasa Komput.* 21 (2021) 139–150. <https://doi.org/10.30812/matrik.v21i1.1092>.
- [5] A.F. Nugraha, Naïve Bayes dan Support Vector Machine Berbasis PSO untuk Seleksi Fitur pada Sentiment Analysis, *Innov. Res. Informatics* 4 (2022) 56–61. <https://doi.org/10.37058/innovatics.v4i2.5291>.
- [6] T.M. Shami, A.A. El-Saleh, M. Alswaitti, Q. Al-Tashi, M.A. Summakieh, S. Mirjalili, Particle Swarm Optimization: A Comprehensive Survey, *IEEE Access* 10 (2022) 10031–10061. <https://doi.org/10.1109/ACCESS.2022.3142859>.
- [7] T. Astuti, Y. Astuti, Analisis Sentimen Review Produk Skincare Dengan Naïve Bayes Classifier Berbasis Particle Swarm Optimization (PSO), *J. Media Inform. Budidarma* 6 (2022) 1806. <https://doi.org/10.30865/mib.v6i4.4119>.
- [8] Rayuwati, Husna Gemasih, Irma Nizar, Implementasi Algoritma Naive Bayes Untuk Memprediksi Tingkat Penyebaran Covid, *Jural Ris. Rumpun Ilmu Tek.* 1 (2022) 38–46. <https://doi.org/10.55606/jurritek.v1i1.127>.
- [9] N.B. Muliawan, I.A. Sulistijono, Indonesian Journal of Computer Science, *Indones. J. Comput. Sci.* 12 (2023) 284–301. <https://doi.org/https://doi.org/10.33022/ijcs.v12i1.3135>.
- [10] S. Dey, S. Wasif, D.S. Tonmoy, S. Sultana, J. Sarkar, M. Dey, A Comparative Study of Support Vector Machine and Naive Bayes Classifier for Sentiment Analysis on Amazon Product Reviews, 2020 *Int. Conf. Contemp. Comput. Appl. IC3A 2020* (2020) 217–220. <https://doi.org/10.1109/IC3A48958.2020.233300>.
- [11] Yuyun, Nurul Hidayah, Supriadi Sahibu, Algoritma Multinomial Naïve Bayes Untuk Klasifikasi Sentimen Pemerintah Terhadap Penanganan Covid-19 Menggunakan Data Twitter, *J. RESTI (Rekayasa Sist. Dan Teknol. Informasi)* 5 (2021) 820–826. <https://doi.org/10.29207/resti.v5i4.3146>.
- [12] T. Safitri, Y. Umaidah, I. Maulana, Analisis Sentimen Pengguna Twitter Terhadap Grup Musik BTS Menggunakan Algoritma Support Vector Machine, *J. Appl. Informatics Comput.* 7 (2023) 28–35. <https://doi.org/10.30871/jaic.v7i1.5039>.
- [13] D. Atmajaya, A. Febrianti, H. Darwis, Metode SVM dan Naive Bayes untuk Analisis Sentimen ChatGPT di Twitter, *Indones. J. Comput. Sci.* 12 (2023) 2173–2181. <https://doi.org/10.33022/ijcs.v12i4.3341>.
- [14] Surya Agustian, Rahmat Abdillah, Muhammad Irfansyah, Arah baru penelitian klasifikasi teks: Memaksimalkan Kinerja Klasifikasi Sentimen dari Data Terbatas, *MALCOM (Indonesia J. Mach. Learn. Comput.* 4 (2024).
- [15] M. Ihsan, Benny Sukma Negara, Surya Agustian, LSTM (Long Short Term Memory) for Sentiment COVID-19 Vaccine Classification on Twitter, *Digit. Zo. J. Teknol. Inf. Dan Komun.* 13 (2022) 79–89.

- <https://doi.org/10.31849/digitalzone.v13i1.9950>.
- [16] M. Sahbuddin, S. Agustian, Support Vector Machine Method with Word2vec for Covid-19 Vaccine Sentiment Classification on Twitter, *J. Informatics Telecommun. Eng.* 6 (2022) 288–297. <https://doi.org/10.31289/jite.v6i1.7534>.
- [17] U. Naseem, I. Razzak, P.W. Eklund, A survey of pre-processing techniques to improve short-text quality: a case study on hate speech detection on twitter, *Multimed. Tools Appl.* 80 (2021) 35239–35266. <https://doi.org/10.1007/s11042-020-10082-6>.
- [18] M.A. Rosid, A.S. Fitriani, I.R.I. Astutik, N.I. Mulloh, H.A. Gozali, Improving Text Preprocessing for Student Complaint Document Classification Using Sastrawi, *IOP Conf. Ser. Mater. Sci. Eng.* 874 (2020). <https://doi.org/10.1088/1757-899X/874/1/012017>.
- [19] M. Liang, T. Niu, Research on Text Classification Techniques Based on Improved TF-IDF Algorithm and LSTM Inputs, *Procedia Comput. Sci.* 208 (2022) 460–470. <https://doi.org/10.1016/j.procs.2022.10.064>.
- [20] M. Akkaya, "Applying Particle Swarm Optimization: New Solutions and Cases for Optimized Portfolios," *International Series in Operations Research & Management Science*, vol. 306, pp. 392, 2021. doi: 10.1007/978-3-030-70281-6..