BAB II
LANDASAN TEORI

2.1 Landasan Teori

2.1.1 Analisis Sentimen

Analisis sentimen adalah sentimen dari teks subjektif yang dianalisis, diproses, diringkas, dan inferensial. Ini merupakan cabang penelitian dari *text mining* (Cahyaningtyas dkk., 2021). Analisis sentimen adalah teknik *Natural Language Processing* (NLP) yang digunakan untuk menentukan apakah data tersebut merupakan data yang bersifat positif, negatif atau netral.

Analisis sentimen dan klasifikasi sentimen adalah dua metodologi yang digunakan dalam *opinion mining* ataupun *text mining*. Keduanya memiliki fitur independennya sendiri, namun terkadang dapat digunakan secara bergantian. Klasifikasi sentimen menunjukkan orientasi sentimen dengan menetapkan label kelas ke dokumen. Orientasi sentimen adalah jenis klasifikasi teks yang mengklasifikasikan data teks berdasarkan orientasi sentimen opini (Saka & Prasetyaningrum, 2025). Dengan orientasi sentimen dapat menunjukkan polaritas pendapat baik benar atau salah berdasarkan subjectivitas. Analisis subjektif adalah proses mengidentifikasi apakah teks atau ulasan data yang diberikan bersifat subjektif atau objektif.

2.1.2 *Text Mining*

Text mining merupakan rangkaian proses untuk mengolah teks mentah yang tidak terstruktur menjadi data terstruktur sehingga dapat dianalisis lebih lanjut. Melalui penerapan berbagai metode analisis, seperti *Naive Bayes* (NBC), *Particle*

Swarm Optimization (PSO), dan teknik pembelajaran lainnya, proses ini membantu menggali pola-pola penting serta informasi baru. Dengan demikian, organisasi dapat menemukan keterkaitan atau insight tersembunyi yang sebelumnya tidak terlihat dalam kumpulan data teks mereka (Fathonah & Herliana, 2021).

Menurut (Husada & Paramita 2021), proses pengolahan teks umumnya meliputi beberapa tahapan penting, seperti pembersihan data, penyeragaman huruf, penghapusan kata-kata umum yang tidak bermakna, serta pengubahan kata ke bentuk dasarnya. Tahapan-tahapan tersebut dijelaskan secara berikut :

a. *Cleaning*

Cleaning merupakan tahap awal untuk membersihkan data teks dari elemen-elemen yang tidak relevan atau mengganggu analisis. Pada tahap ini, berbagai komponen seperti tag HTML, emotikon, tanda pagar (hashtag), nama pengguna, hingga tautan URL dihilangkan agar teks menjadi lebih bersih dan siap diproses pada langkah berikutnya.

b. *Case Folding*

Case folding merupakan tahap penyeragaman teks dengan mengonversi seluruh huruf menjadi bentuk *lowercase*. Pada proses ini, berbagai karakter selain huruf yang tidak diperlukan dalam analisis juga dibuang, sehingga teks menjadi lebih konsisten dan mudah diproses pada langkah selanjutnya.

c. *Tokenizing*

Tokenizing merupakan tahap pemisahan kalimat menjadi unit-unit kata individual. Pada proses ini, rangkaian kata dalam sebuah kalimat dipecah

menjadi elemen-elemen kecil yang disebut token, sehingga setiap kata dapat dianalisis secara terpisah.

d. *Stopword Removal*

Stopword removal merupakan tahap penyaringan kata dengan cara menghapus kata-kata yang dianggap tidak memiliki kontribusi penting dalam analisis. Dengan membuang kata-kata tersebut, data menjadi lebih fokus dan dapat membantu meningkatkan akurasi model klasifikasi..

e. *Stemming*

Stemming merupakan teknik yang digunakan untuk mengembalikan sebuah kata ke bentuk dasarnya sesuai dengan aturan morfologi bahasa. Melalui proses ini, kata turunan diasumsikan memiliki makna yang sama dengan kata dasar yang menjadi acuannya..

2.1.3 Algoritma Text Mining

Algoritma *text mining* adalah algoritma *data mining*. Teknik *data mining* dan *text minig* telah sering digunakan untuk menganalisis kuisioner dan *review* data. Perbedaan mendasar dengan *data mining* pada umumnya. *Text mining* memerlukan beberapa tahap awal (*preprocessing*) yang pada intinya adalah mempersiapkan agar teks dapat diubah lebih terstruktur. Terdapat berbagai macam algoritma yang digunakan dalam *text mining* dengan tugasnya yang berbeda (Husada & Paramita, 2021).

Berikut ini adalah gambaran secara umum mengenai algoritma *text mining*:

a. Algoritma *Unsupervised*

Algoritma yang melakukan pembelajaran sendiri dan tidak menyediakan data latih untuk melatih sistem. Sistem yang berkembang sendiri dan menghasilkan hasil yang bergantung pada data yang diberikan. Algoritma ini biasanya digunakan dalam kategori pengelompokan dan permodelan topik. Kedua kategori tersebut digunakan untuk segmentasi data ke dalam kelompok.

b. Algoritma *Supervised*

Algoritma *supervised* kebalikannya dari algoritma *unsupervised*, pada algoritma ini memerlukan data latih, dan data algoritma ini masuk ke dalam kategori unruk klasifikasi.

2.1.4 Algoritma Naive Bayes Classifier

Naive Bayes merupakan salah satu algoritma yang digunakan untuk menghasilkan teks. Kegunaan dari algoritma tersebut adalah untuk memprediksi probabilitas di masa depan berdasarkan pengalaman di masa lalu.

Naive Bayes adalah model penyederhanaan dari metode *bayes* yang sesuai dalam mengklasifikasikan teks. *Naive Bayes* adalah teknik prediksi dengan berbasis probabilistik sederhana yang berdasarkan pada penerapan teorema *Bayes* dengan asumsi independensi (tidak ketergantungan) yang kuat.

Menurut (Setiawan dkk., 2024), *Naive Bayes* mudah digunakan dan tidak mahal secara komputasi dan sangat berguna untuk kasus di mana dimensi input tinggi juga untuk kelas tertentu sebagai positif atau negatif kata-kata independen satu sama lain secara kondisional. Ciri utama dari *Naive Bayes* ini adalah asumsi

yang sangat kuat (naif) akan independensi dari masing-masing kondisi atau kejadian.

Naive Bayes sering kali merupakan algoritma yang bagus untuk memulai pembelajaran dan sangat dapat diterapkan sebagai model awal dalam tahapan pembuktian konsep untuk proyek analitik. Ini juga berfungsi sebagai tolak ukur yang baik untuk dibandingkan dengan model lain. Penerapan model *Naive Bayes* dalam sistem produksi cukup mudah dan pengugunaan alat penambangan data bersifat opsional.

2.1.5 Persamaan Metode *Naive Bayes*

Fitur utama klasifikasi *Naive Bayes* adalah untuk mendapatkan hipotesis yang kuat dari setiap kondisi atau peristiwa. Dasar dari konsep *Naive Bayes* yang dipakai dengan rumus *Bayes*:

$$P(A|B) = \frac{\left(P(B|A)P(A)\right)}{P(B)} \qquad 2.1$$

Keterangan:

A : Hipotesis data dari suatu kelas *spesifik*

B : Data dengan kelas yang belum diketahui

P(A/B) : Probabilitas dari hipotesis A yang didasari dari kondisi B (posteriori probabilitas

P(A) : Probabilitas dari hipotesis A (prior probabilitas)

P(B/A) : Probabilitas B yang didasari dari kondisi pada hipotesis A

P(B) : Probabilitas B

Dalam menjelaskan mengenai metode dari *Nave Bayes* yang perlu diketahui bahwa, untuk proses klasifikasi perlu adanya sejumlah petunjuk dalam menentukan kelas mana yang cocok bagi sampel yang sedang dianalisis. Oleh karena itu, persamaan dari rumus metode *Naive Bayes* diatas telah disesuaikan sebagai berikut:

$$P(C|F_1 \dots F_n) = \frac{P(C) * P(F_1 \dots F_n|C)}{P(F_1 \dots F_n)} \quad 2.2$$

Dari rumus di atas dapat dijelaskan bahwa variabel C yang merupakan representasi dari kelas, sementara $F_1 \dots F_n$ adalah representasi dari karateristik petunjuk yang dibutuhkan dalam melakukan klasifikasi. Dari rumus tersebut menjelaskan peluang masuknya sampel karateristik tertentu dalam kelas C (*Posterior*) yang merupakan peluang munculnya kelas C (sebelum masuknya sampel tersebut, seringkali disebut *Prior*), dikali dengan peluang kemunculan karateristik-karateristik sampel pada kelas C (disebut juga *likelihood*), dibagi dengan peluang kemunculan karateristik-karateristik sampel secara global (disebut juga *evidance*). Oleh karena itu, rumus (2) dapat ditulis dengan sederhana seperti berikut ini:

$$Posterior = \frac{prior\ x\ likelihood}{evidence} \quad 2.3$$

Nilai dari *evidence* akan selalu tetap disetiap kelas pada satu sampel. Nilai dari *posterior* tersebut nantinya dibandingkan dengan nilai-nilai *posterior* kelas lainnya dalam menentukan kelas suatu sampel diklasifikasikan. Penjabaran lebih lanjur dari bayes tersebut dilakukan dengan menjabarkan $(C|F_1 \dots F_n)$ menggunakan aturan perkalian berikut:

$$P\left(C\middle|F_1 \dots F_n = P(C)P(F_1, \dots, F_n|C)\right)$$

$$= P(C)P(F_1|C)P(F_1, \dots, F_n|C, F_1)$$

$$= P(C)P(F_1|C)P(F_2|C, F_1)P(F_3, \dots, F_n|C, F_1, F_2)$$

$$= (C)P(F_1|C)P(F_2|C, F_1)P(F_3|C, F_1F_2)P(F_4, \dots, F_n|C, F_1, F_2, F_3)$$

$$= P(C)P(F_1|C)P(F_2|C, F_1)P(F_3|C, F_1F_2) \dots P\left(F_{n|}C, F_1, F_2, F_3, \dots, F_{n-1}\right) \quad 2.5$$

Persamaan diatas menyebabkan semakin banyak dan semakin kompleksnya faktor syarat yang dapat mempengaruhi nilai probabilitasnya, yang hampir tidak mungkin atau mustahil untuk dianalisa satu persatu. Akhirnya, perhitungan itu jadi sulit untuk dilakukan. Maka digunakan asumsi independensi yang tinggi (naif), bahwa masing-masing petunjuk $(F_1, F_2, \dots, F_n)$ saling bebas (independen) satu dengan yang lain. Dengan asumsi itu, maka terdapat suatu kesamaan sebagai berikut:

$$P\left(F_i\middle|F_j\right) = \frac{P\left(F_i \cap F_j\right)}{P\left(F_j\right)} = \frac{P(F_i)P\left(F_j\right)}{P\left(F_j\right)} = P(F_i) \quad 2.4$$

Untuk $i \neq j$, sehingga

$$\left(F_i\middle|C, F_j\right) = (F_i|C) \quad 2.6$$

Persamaan di atas merupakan model dari teorema *Naive Bayes* yang selanjutnya akan digunakan dalam proses klasifikasi. Untuk klasifikasi dengan data kontinyu digunakan rumus Densitas Gauss:

$$\left(X_i = x_i\middle|Y = y_j\right) = \frac{1}{\sqrt{2\pi\sigma ij}} e^{-(x_i - \mu ij)^2/2\sigma 2ij} \quad 2.7$$

Keterangan:

P : Peluang

X_i : Atribut ke i

x_i : Nilai Atribut ke i

Y : Kelas yang dicari

y_i : Sub Kelas Y yang dicari

μ : Mean, menyatakan rata-rata dari seluruh atribut

σ : Deviasi standar, menyatakan varian dari selurih atribut.

2.1.6 Alur Model *Naive Bayes* dalam Melakukan Klasifikasi Sentimen

Menurut (S Samsir, A Ambiyar, U Verawardina, F Edi, 2021) Teknik o*pinion mining* melalui Twitter (X) dengan metode *Naive Bayes* dimana *Crawling data* dilakukan dengan memberikan kata kunci, proses *labelling* untuk menentukan sentimen diselesaikan setelah data telah terkumpul. Tahap berikutnya dilakukan *preprocessing* untuk menyeleksi data dan menghapus kata-kata yang tidak bermakna. Tahap terakhir dalam *opinion mining* adalah ekstrak fitur untuk mempermudah dalam klasifikasi *Naive Bayes*. Tahap ini menghasilkan model dan digunakan untuk menunjukkan ketepatan hasil dari klasifikasi yang telah digunakan untuk menunjukkan ketepatan hasil dari klasifikasi yang telah dilakukan. Data dalam bentuk tweet di *Crawling* dari *Twitter* (X) dan disimpan dalam bentuk CSV file. Data dibagi menjadi dua dataset yaitu data latih dan data uji. Pelabelan akan diberikan untuk membedakan *tweet* positf dan negatif, dan netral. Metode *Naive Bayes* digunakan pada tahap klasifikasi sentimen dan interpretasi hasil analisis sentimen..

2.1.7 Data Crawling

Data *Crawling* atau bisa disebut juga dengan *web scraping* atau *spidering*, merupakan metode yang memungkinkan untuk mengekstrak data secara otomatis

dari internet (Talisman dkk., 2024). Kegiatan ini dilakukan oleh "Bot" atau perangkat lunak yang disebut *crawler*. Data yang diambil dari hasil *crwal* ini umumnya akan dianalisa, dan digunakan sebagai bahan pengembangan sistem, atau bahkan digunakan sebagai data penelitian tertentu. Aktivitas *Crawling* ini juga memungkinkan untuk mengumpulkan data dari berbagai sumber, seperti *database* dan API.

Berikut adalah beberapa metode yang dapat digunakan dalam proses *Crawling* data:

a. *Web Scraping*, merupakan teknik untuk mengubah data web yang tidak terstruktur menjadi data terstruktur yang dapat disimpan dan dianalisis dalam *spreadsheet* atau *database* pusat. Metode ini melibatkan pengambilan data yang didapatkan dari halaman web dengan cara mengekstrak informasi dari kode HTM, dalam melakukan *web sraping* dapat menggunakan bahasa pemograman seperti Python atau java.
b. API (*Application Programing Interface*), merupakan antar muka yang digunakan berkomunikasi dengan aplikasi atau platform tertentu. Dapat diartikan sebagai cara yang dilakukan komputer dengan aplikasi untuk mendapatkan informasi atau mengambil data dari berbagai *platform* seperti Twitter (X), Facebook, dan Instagram.
c. RSS-*Feeds* (*Realy Simple Syndication*), merupakan format yang digunakan untuk mengambil dan mengumpulkan data berita atau konten dari situs web secara periodik. RSS mengambil berita utama tersebut ke komputer untuk

pemindahan secara cepat. Ini merupakan cara sederhana untuk berbagi informasi antara situs web yang sebagia besar didasarkan pada XML.

d. *Web Crawling*, metode ini melibatkan pengambilan data dari berbagai halaman web dengan cara mengikuti tautan atau link yang ada pada halaman tersebut. *Web Crwaling* dapat dilakukan dengan menggunakan program bot atau *web crawler* seperti *Scrapy* atau *Beautiful Soup*.
e. *Social Media Crawling*, metode ini melibatkan pengambilan data dari platform media sosial seperti Twitter, Facebook, dan Instagram dengan menggunakan API atau *web scraping*.

2.1.8 Data Latih dan Data Uji

Data latih dan data uji merupakan dua set data atau *dataset* yang digunakan dalam proses pembelajaran mesin (*machine learning*). Data latih atau data training digunakan untuk melatih model mesin, sedangkan data uji atau data testing digunakan untuk menguji kinerja model yang telah dilatih (Savitri, N. L. P. C., Rahman, R. A., Venyutzky, R., & Rakhmawati, 2021). Tujuan dari menggunakan data uji adalah untuk memastikan bahwa model yang dilatih memiliki kemampuan untuk memprediksi dengan akurat data yang belum pernah dilihat sebelumnya.

2.1.9 Term Frequency-Inverse Document Frequency (TF-IDF)

Weighting Word merupakan mekanisme untuk memberikan skor pada frekuensi kemunculan suatu kata dalam dokumen teks. Salah satu metode popular yang digunakan untuk pembobotan kata adalah TF-IDF. TF-IDF adalah suatu metode yang digunakan untuk mengekstraksi keunikan dari suatu *text.* Metode ini digunakan dalam pembobotan posisi kata dalam suatu dokumen dan memiliki

efisiensi yang cukup tinggi serta tingkat akurasi yang baik. Metode ini menghitung nilai *Term Frequency* (TF) dan *Inverse Document Frequency* (IDF) pada setiap *term* (kata) pada setiap dokumen, sedangkan IDF, menunjukkan tingkat harga yang digunakan dalam dokumen.

TF-IDF ini memberikan bobot yang lebih tinggi untuk istilah yang penting dan bobot yang lebih rendah untuk istilah yang tidak penting. Nilai vektornya terletak diantara 0 hingga 1. 0 berarti istilah tersebut tidak penting dalam konteks dokumen yang kita cari dan 1 berarti istilah tersebut relevan. Untuk menghitung nilai TF-IDF, dokumen diubah menjadi file teks terbalik di mana untuk semua dokumen, kata-kata diekstraksi dan sesuai dengan setiap kata, bobot diberikan, yaitu nilai kemunculan istilah dan dari metode ini adalah sebagai berikut:

$$TFIDF_t = TF \times log \frac{N}{df} \quad 2.8$$

Dimana t adalah *term*, TF adalah jumlah dari *term* t yang muncul dalam dokumen, N adalah total dokumen, dan df adalah jumlah dokumen yang memuat *term* t.

Pada TF-IDF terdapat rumus untuk menghitung bobot (W) setiap dokumen untuk dikunci.

$$W_{ij} = tf_{ij} \times IDF_j \quad 2.9$$

Setelah bobot (W) masing-masing dokumen diketahui, selanjutnya dilakukan proses *sort* (pengurutan) dimana semakin besar nilainya, maka semakin besar pula tingkat kemiripan dokumen dengan kata kunci, begitu juga sebaliknya.

Term Frequency – Inverse Document Frequency (TF-IDF) merupakan metode pembobotan kata yang mengukur seberapa penting suatu kata *(term, t)*

terhadap suatu dokumen *(document, d)* dalam keseluruhan kumpulan dokumen. Nilai TF-IDF diperoleh dari hasil perkalian antara *Term Frequency (TF)* dan *Inverse Document Frequency (IDF).*

Rumus perhitungannya adalah sebagai berikut:

$$IDF(t) = \log\left(\frac{N}{df_t}\right)$$

$$w_{t,d} = TF(t,d) \times IDF(t) \qquad 2.10$$

Keterangan:

1. *t* : istilah *(term)* atau kata yang sedang dihitung bobotnya
2. *d* : dokumen tertentu
3. *f_{t,d}* : frekuensi kemunculan term *t* dalam dokumen *d*
4. *N* : jumlah total dokumen dalam dataset
5. *df_t* : jumlah dokumen yang mengandung term *t*
6. *w_{t,d}* : bobot akhir term *t* dalam dokumen *d*

TF menunjukkan frekuensi kemunculan sebuah kata di dalam satu dokumen, sementara IDF menggambarkan seberapa jarang kata tersebut muncul pada keseluruhan kumpulan dokumen. Jika suatu kata memiliki nilai TF-IDF yang tinggi, berarti kata tersebut dianggap lebih signifikan dan memiliki bobot penting dalam dokumen tempat kata itu muncul..

Dalam penelitian ini, pembobotan kata menggunakan TF-IDF diterapkan dengan bantuan fungsi *TfidfVectorizer* dari pustaka *scikit-learn*. Adapun beberapa parameter yang digunakan yaitu:

1. *ngram_range* menentukan panjang kombinasi kata yang akan dihitung, misalnya (1,1) hanya menghitung kata tunggal (*unigram*), sedangkan (1,2) menghitung *unigram* dan *bigram*.
2. *min_df* merupakan batas minimum frekuensi kemunculan suatu kata agar tetap dipertimbangkan; kata yang muncul lebih sedikit dari nilai ini akan diabaikan.
3. *max_df* menunjukkan batas maksimum proporsi dokumen yang mengandung suatu kata; kata yang muncul terlalu sering dianggap tidak informatif dan dihapus.
4. *max_features* menentukan jumlah maksimum fitur (kata unik) yang akan dipertimbangkan berdasarkan nilai TF-IDF tertinggi.

Pengaturan parameter tersebut bertujuan untuk menghasilkan representasi teks yang optimal, mengurangi *noise* dari kata yang terlalu umum atau terlalu jarang, serta meningkatkan kualitas hasil analisis sentimen.

2.1.10 Seleksi Fitur

Pemilihan fitur adalah teknik *preprocessing* penting yang bertujuan untuk meningkatkan algoritma pembelajaran (misalnya, klasifikasi) dengan meningkatkan kinerjanya atau mengurangi waktu pemrosesan atau keduanya. Seleksi fitur dilakukan agar metode dapat melakukan klasifikasi teks secara optimal, hal ini dikarenakan dokumen yang memiliki fitur dalam jumlah besar akan mengalami kesulitan dalam mengklasifikasikan teks. Dimensi fitur yang tinggi sangat berpengaruh dalam proses klasifikasi karena memungkinkan adanya duplikasi dan redudansi pada fitur (Efrizoni dkk., 2022).

2.1.11 Particle Swarm Optimization (PSO)

Particle Swarm Optimization merupakan teknik metaheuristik berbasis populasi yang terinspirasi dari perilaku kawanan burung dan ikan saat berpindah satu tempat ke tempat lain untuk mencari makanan. Menurut (Mahapatra, A. K., Panda, N., & Pattanayak, 2022), kinerja dari algoritma PSO sangat bergantung dengan strategi parameter yang tepat dalam menyempurnakan parameternya, *Inertia Weight* merupakan salah satu parameter PSO yang digunakan dalam menyeimbangkan eksporasi dan eksploitasi PSO.

PSO terdiri dari sekumpulan partikel, dimana setiap partikel mewakili sebuah kandidat solusi dan bergerak dalam ruang pencarian dengan kecepatan tertentu. Mekanisme utamanya adalah proses pencarian secara bersama-sama, dimana setiap partikel dipengaruhi oleh dua acuan yaitu posisi terbaik yang pernah dicapainya sendiri *(pbest*) dan posisi terbaik yang ditemukan oleh seluruh kelompok partikel (*gbest*).(Mahapatra, A. K., Panda, N., & Pattanayak, 2022).

Konsep perumusan dari *Particle Swarm Optimization* (PSO) adalah sebagai berikut:

$$Vi(t) = Vi(t-1) + c_1 r_1 [Xpbest_i - Xi(t)] + c_1 r_1 [Xgbest - Xi(t)] Xi(t)$$

$$= Xi(t-1) + Vi(t) \qquad 2.11$$

Keterangan:

$Vi(t)$: Kecepatan partikel i saat iterasi t

$Xi(t)$: Posisi partikel i saat iterasi t

$c_1\ dan\ c_2$: *Learning rates* untuk kemampuan individu dan pengaruh social

$r_1\ dan\ r_2$: Bilangan random yang berdistribusi uniformal dalam interval 0 dan 1

$Xpbest_i$: Posisi terbaik partikel i

$Xgbest$: Posisi terbaik global

Terdapat beberapa proses yang terjadi di dalam algoritma PSO, diantaranya sebagai berikut:

1. Inisialisasi

a. Inisialisasi kecepatan awal

Ketika iterasi ke = 0, dapat dipastikan bahwa nilai kecepatan awal dari semua partikel adalah 0.

b. Inisialisasi posisi awal partikel

Pada iterasi ke = 0, posisi awal partikel dibandingkan dengan persamaan:

$$X = X_{min} + rand[0,1] * (X_{max} - X_{min}) \quad 2.12$$

c. Inisialisasi pBest dan gBest

Pada iterasi ke = 0, pBest akan disamakan dengan nilai posisi awal partikel. Sedangkan gBest dipilih dari suatu pBest dengan *fitness* tertinggi.

2. *Update* Kecepatan

Untuk melakukan update kecepatan digunakan rumus:

$$V_{i.j}^{t=1} = w.v_{i.j}^{t} + c_1 * r_1\left(pBest_{i.j}^{t} - x_{i.j}^{t}\right) + c_2 * r_2\left(gBest_{g.j}^{t} - x_{i.j}^{t}\right) \quad 2.13$$

3. *Update* Posisi dan Hitung *Fitness*

Untuk melakukan *update* posisi, digunakan rumus:

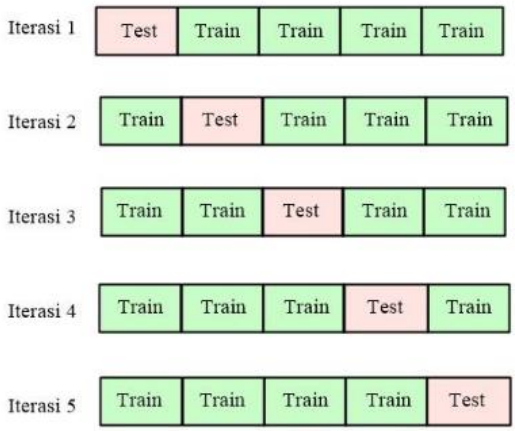
$$X_{i.j}^{t+1} = x_{i.j}^{t} + v_{i.j}^{t+1} \quad 2.14$$

4. *Update* pBest dan gBest

Setelah posisi partikel diperbarui, nilai *pBest* dibandingkan dengan nilai *fitness* hasil pergerakan terbaru. Jika *fitness* yang dihasilkan lebih tinggi, maka nilai tersebut akan menggantikan *pBest* sebelumnya. Dari kumpulan pBest tersebut, nilai *fitness* terbaik akan ditetapkan sebagai *gBest* yang baru.

2.1.12 K-Fold Cross Validation

K-fold Cross Validation merupakan metode validasi yang digunakan untuk menilai performa suatu algoritma dengan cara membagi dataset secara acak menjadi K bagian. Pada setiap putaran pengujian, satu bagian dijadikan data uji, sedangkan K–1 bagian lainnya digunakan sebagai data latih. Proses ini dilakukan berulang hingga seluruh bagian pernah menjadi data uji.(Nugroho dkk., 2020). Salah satu metode dalam *cross-validation* adalah *K-Fold Cross Validation*, yaitu teknik yang membagi dataset menjadi K bagian dengan ukuran yang seimbang. Pendekatan ini bertujuan meminimalkan bias dalam proses evaluasi model. Proses pelatihan dan pengujian dilakukan sebanyak K kali, di mana setiap bagian secara bergiliran berperan sebagai data uji. Alur kerja metode ini dapat digambarkan sebagai berikut. (Aziz dkk., 2020):



Gambar 2. 1 Ilustrasi K-Fold Cross Validation

2.1.13 Python

Python merupakan perangkat lunak *open-source* yang populer berkat bahasa pemrogramannya yang fleksibel dan mudah digunakan. *Python* memiliki beragam pustaka (*library*) yang mendukung berbagai kebutuhan pemrograman. Untuk memasang *library* tambahan, pengguna dapat memanfaatkan perintah pip (Nofiyanti dkk., 2021).

1. *Tweepy*

   Tweepy merupakan pustaka *(library)* Python yang digunakan untuk terhubung dengan *API X* dan melakukan pengambilan data dari *platform* tersebut.

2. *Pandas*

   Pandas adalah pustaka *Python* yang berfungsi untuk mengelola serta menganalisis data dalam bentuk terstruktur. Library ini dapat diakses dengan perintah *import pandas as pd*. Salah satu struktur data utama yang umum digunakan dalam *Pandas* adalah *DataFrame*, yang memungkinkan mempermudah proses analisis.

3. *Numerical* Python

   Numerical Python atau lebih dikenal sebagai *Numpy*, adalah pustaka *Python* yang digunakan untuk melakukan perhitungan numerik, khususnya operasi pada vektor dan matriks. *Library* ini mempermudah proses komputasi matematika yang kompleks secara efisien..

4. *Snscrape*

   Snscrape adalah program yang digunakan untuk menyaring layanan media sosial untuk mencari hal-hal yang diperlukan secara profil pengguna, tagar, atau pencarian.

5. CSV

   CSV (*Comma Separated Value*) adalah *library* Python sebagai tempat untuk menyimpan data dengan format ".csv".

2.1.14 Badan Penyelenggara Jaminan Sosial (BPJS)

Badan Penyelenggara Jaminan Sosial (BPJS) merupakan lembaga milik negara yang diberikan tugas oleh pemerintah untuk menyediakan layanan jaminan kesehatan bagi seluruh masyarakat Indonesia. Lembaga ini bertanggung jawab memastikan setiap warga mendapatkan akses terhadap pelayanan kesehatan yang memadai.. (Rosadi dkk., 2021).

Sesuai dengan Undang-Undang Nomor 24 Tahun 2011 pasal 60, pemerintah mulai mengoperasikan BPJS Kesehatan pada 1 Januari 2014 sebagai pelaksanaan mandat UU BPJS. Sejak saat itu, BPJS Kesehatan menjalankan program Jaminan Kesehatan Nasional, yang menjamin seluruh warga negara Indonesia pemegang

kartu BPJS memperoleh akses pelayanan kesehatan secara menyeluruh. (Rosadi dkk., 2021).

2.1.15 Twitter (X)

Aplikasi X atau lebih dikenal sebagai *Twitter* adalah salah satu *platform* media sosial terkemuka yang didirikan pada tahun 2006. *Twitter* memungkinkan penggunanya untuk membuat dan membagikan pesan singkat yang disebut *"tweet*" dalam format teks, gambar, video, atau tautan. Dengan batasan jumlah karakter yang relatif kecil (280 karakter per *tweet*), *Twitter* mempromosikan komunikasi yang singkat dan langsung (Soumya & Pramod, 2020).

Twitter telah menjadi salah satu sumber utama informasi dan diskusi tentang berbagai topik, termasuk politik, berita terkini, hiburan, dan lainnya. Pengguna *Twitter* dapat mengikuti akun-akun yang relevan dengan minat mereka dan berinteraksi dengan pesan yang diposting oleh akun-akun tersebut melalui mekanisme *retweet*, balasan, dan suka. Keunggulan *Twitter* sebagai *platform* media sosial dalam konteks penelitian ini adalah kemampuannya untuk memberikan akses *real-time* terhadap berbagai peristiwa dan topik yang sedang tren. Dengan analisis sentimen yang dilakukan terhadap data *Twitter*, peneliti dapat memperoleh pemahaman yang mendalam tentang *opini*, sikap, dan pandangan masyarakat terkait dengan pelayanan peseta BPJS.

Dalam penelitian ini, data dari aplikasi X (*Twitter*) akan menjadi sumber utama informasi untuk melakukan analisis sentimen terhadap pelayanan peserta BPJS. Dengan memahami dinamika dan tren yang terjadi di *Twitter*, penelitian ini

akan memberikan wawasan yang berharga tentang pola pikir dan sikap masyarakat dalam proses pelayanan peserta BPJS.

2.1.16 Synthetic Minority Over-sampling Technique (SMOTE)

Synthetic Minority Over-sampling Technique (SMOTE) adalah metode penyeimbang data kelas (*class balancing*) yang dikembangkan oleh Chawla dkk. (2002) untuk mengatasi permasalahan *imbalanced dataset*, yaitu kondisi ketika jumlah data pada satu kelas (mayoritas) jauh lebih banyak dibandingkan kelas lainnya (minoritas). Ketidakseimbangan ini sering menyebabkan model pembelajaran mesin bias terhadap kelas mayoritas, sehingga akurasi untuk kelas minoritas rendah. Dalam penelitian ini, SMOTE digunakan untuk menyeimbangkan distribusi data sentimen negatif, netral, dan positif sebelum dilakukan pelatihan model Naive Bayes yang dioptimasi menggunakan *Particle Swarm Optimization* (PSO). Dengan data yang seimbang, model diharapkan dapat memprediksi semua kelas secara lebih akurat dan konsisten.

Prinsip Kerja SMOTE

1. Untuk setiap sampel x_i pada kelas minoritas, cari k tetangga terdekat (nearest neighbors) menggunakan jarak Euclidean.
2. Pilih secara acak salah satu tetangga x_{nn} dari hasil pencarian tersebut.
3. Bentuk data sintetis baru dengan interpolasi linier antara x_i dan x_{nn} .

Rumus pembuatan sampel sintetis:

$$x_{new} = x_i + \delta \times (x_{nn} - x_i) \quad 2.15$$

dengan:

a. x_{new} = data sintetis yang dihasilkan

b. x_i = data asli dari kelas minoritas

c. x_{nn} = salah satu tetangga terdekat dari $x\ i$ x i

d. δ = bilangan acak antara 0 dan 1

Proses ini diulang hingga jumlah sampel pada kelas minoritas menjadi seimbang dengan kelas mayoritas.

Keunggulan SMOTE :

a. Menghasilkan data baru yang bervariasi sehingga mengurangi risiko *overfitting*.
b. Memperbaiki kemampuan model dalam mengenali pola pada kelas minoritas.
c. Dapat digabungkan dengan metode lain seperti *cross-validation* untuk evaluasi yang lebih reliabel.

2.1.17 Cross Validation

Cross-Validation adalah teknik evaluasi model yang digunakan untuk mengukur kinerja algoritma pembelajaran mesin secara lebih akurat dengan memanfaatkan seluruh data yang tersedia. Metode ini bertujuan mengurangi bias evaluasi yang mungkin terjadi jika data hanya dibagi sekali menjadi training set dan testing set.

Cross-Validation bekerja dengan cara membagi dataset menjadi beberapa subset (lipatan atau *fold*), kemudian melakukan pelatihan *(training)* dan pengujian *(testing)* model secara berulang pada kombinasi subset yang berbeda. Dengan cara

ini, setiap data memiliki kesempatan yang sama untuk digunakan sebagai data latih dan data uji.

Jenis *Cross Validation* :

1. *k-Fold Cross-Validation*

   Dataset dibagi menjadi k bagian yang berukuran sama. Model dilatih sebanyak k kali, setiap kali menggunakan k – 1 bagian sebagai data latih dan 1 bagian sebagai data uji. Nilai k yang sering digunakan adalah 5 atau 10.

   Rumus untuk menghitung rata-rata skor kinerja :

   $$Score_{avg} = \frac{1}{k} \sum_{i=1}^{k} score_{\text{i}} \qquad 2.16$$

   di mana:

   a. $Score_i$ = nilai metrik evaluasi (misalnya akurasi, F1-score) pada fold ke- i

   b. k = jumlah fold

2. *Stratified k-Fold Cross-Validation*

   Merupakan variasi dari *k-Fold* di mana pembagian data dilakukan dengan menjaga proporsi kelas yang sama pada setiap *fold*. Teknik ini cocok digunakan untuk dataset dengan distribusi kelas yang tidak seimbang.

3. *Leave-One-Out Cross-Validation (LOOCV)*

   Setiap iterasi hanya menyisakan satu data sebagai uji, sementara sisanya digunakan sebagai latih. Cocok untuk dataset kecil, namun memerlukan waktu komputasi yang lebih lama.

Dalam penelitian ini digunakan *Stratified k-Fold Cross-Validation* untuk memastikan setiap *fold* memiliki distribusi kelas sentimen yang seimbang. Hal ini penting karena dataset hasil scraping seringkali memiliki proporsi kelas yang tidak

merata. Penggunaan *Cross-Validation* diharapkan memberikan evaluasi kinerja model *Naive Bayes* yang dioptimasi dengan PSO secara lebih stabil dan tepercaya. Kelebihan *Cross-Validation*

1. Menggunakan seluruh data untuk pelatihan dan pengujian sehingga evaluasi lebih akurat.
2. Mengurangi kemungkinan bias akibat pembagian data yang tidak merata.
3. Cocok untuk dataset berukuran kecil hingga sedang

2.2 Penelitian Terkait

2.2.1 State of The Art

Terdapat beberapa penelitian terdahulu yang dilakukan dalam topik Optimasi *Naive Bayes Classifier* Menggunakan *Particle Swarm Optimization* Pada Analisis Sentimen Terhadap Pelayanan Peserta BPJS. Namun, setiap penelitian menggunakan metode yang unik dan memberikan hasil yang berbeda-beda. Berikut ini penelitian yang terkait dengan penelitian yang dilakukan disajikan pada Tabel 2.2.

Tabel 2. 1 State of The Art

No	Peneliti	Judul	Permasalahan	Algoritma	Hasil Penelitian
1.	(Arsi dkk., 2021)	Optimasi SVM Berbasis PSO pada Analisis Sentimen Wacana Pindah Ibu Kota Indonesia	Rencana pemindahan ibu kota tercantum dalam Rencana Pembangunan Jangka Menengah Nasional (RPJMN) 2020–2024. Tanggapan masyarakat terhadap rencana ini bervariasi, baik melalui media televisi maupun platform media sosial, khususnya Twitter. Pola opini dan kecenderungan sikap	*Support Vector Machine* (SVM)	Hasil eksperimen yang menggunakan 1.319 *tweet* (terdiri dari 457 *twee*t dengan sentimen positif dan 862 *tweet* dengan sentimen negatif) menunjukkan adanya peningkatan akurasi sebesar 2,09%. Akurasi ini meningkat dari 79,06% menjadi 81,15%, dan dikategorikan sebagai "*Good Classification*".

Tabel 2. 2 State of The Art (Lanjutan 1)

No	Peneliti	Judul	Permasalahan	Algoritma	Hasil Penelitian
			pengguna *Twitter* terhadap wacana pemerintah tersebut dapat dianalisis melalui metode analisis sentimen		
2.	(Falasari & Muslim, 2022)	*Optimize naïve bayes classifier using chi square and term frequency inverse document frequency for amazon review sentiment analysis*	Pesatnya perkembangan internet menyebabkan informasi tersebar dengan sangat cepat, termasuk dalam bidang perdagangan. Banyak konsumen yang menulis pendapat atau ulasan tentang produk yang mereka beli di media sosial maupun situs daring lainnya. Ulasan yang berbentuk teks panjang tersebut memerlukan pemrosesan otomatis agar opini yang terkandung di dalamnya dapat dikenali dengan tepat.	*Naïve Bayes*	Dalam penelitian ini, digunakan dataset berlabel sentimen (*field amazon_labelled*) yang diperoleh dari *UCI Machine Learning Repository.* Dataset tersebut terdiri dari 500 ulasan positif dan 500 ulasan negatif. Hasil eksperimen menunjukkan bahwa *Naïve Bayes Classifier* mampu mencapai akurasi sebesar 82% dalam menganalisis sentimen ulasan Amazon. Penerapan metode *Chi-Square* dan TF-IDF pada *Naïve Bayes Classifier* meningkatkan akurasi menjadi 83%.

Tabel 2. 3 State of The Art (Lanjutan 2)

No	Peneliti	Judul	Permasalahan	Algoritma	Hasil Penelitian
3.	(Undamayanti dkk., 2022)	Analisis Sentimen Menggunakan Metode Naive Bayes Berbasis Particle Swarm Optimization Terhadap Pelaksanaan Program Merdeka Belajar Kampus Merdeka	Selama pelaksanaan program MBKM di berbagai perguruan tinggi, terdapat beberapa permasalahan dan kendala yang dialami oleh mahasiswa. Salah satunya adalah ketidakjelasan implementasi kurikulum, karena belum tersedia acuan resmi seperti peraturan, buku panduan, petunjuk pelaksanaan, maupun prosedur operasional yang jelas selama kegiatan berlangsung. Selain itu, sebagian besar mitra yang bekerja sama dalam program ini berlokasi di Ibukota.	Naïve Bayes	Berdasarkan hasil penelitian, analisis sentimen menunjukkan bahwa 61,92% opini bersifat positif, yang menandakan bahwa program MBKM diterima cukup baik oleh pengguna Twitter, terutama kalangan mahasiswa. Meskipun demikian, terdapat pula sentimen negatif sebesar 38,08% yang muncul dari sebagian pengguna.
4.	(Taofik Safrudin dkk., 2023)	Optimasi Algoritma K-*Nearest Neighbor* Berbasis *Particle Swarm Optimization* Untuk	Persediaan barang di gudang merupakan komponen penting dalam memastikan kelancaran	K- *Nearest Neighbor*	Berdasarkan evaluasi model menggunakan *10-Fold Cross Validation*, diperoleh nilai akurasi sebesar 82%, *precision* sebesar

Tabel 2. 4 State of The Art (Lanjutan 3)

No	Peneliti	Judul	Permasalahan	Algoritma	Hasil Penelitian
		Meningkatkan Kebutuhan Barang	produksi, karena ketersediaan stok yang cukup mencegah terjadinya kekurangan bahan. Untuk mengatur dan memaksimalkan ketersediaan barang, diperlukan pengelolaan pengiriman yang tepat. Pada perusahaan yang bergerak di sektor industri bahan pokok, jumlah barang yang masuk setiap minggu atau bulan terus bertambah, sehingga penempatan dan pengelompokan barang sesuai dengan kebutuhan menjadi hal yang sangat penting agar pengelolaan gudang lebih efisien.		87,50%, dan *recall* sebesar 80%. Penggunaan *Cross Validation* memungkinkan pengukuran kinerja model yang mempertimbangkan variasi antar percobaan, sehingga dapat diketahui simpangan baku akurasi dan seberapa jauh nilai tiap percobaan menyimpang dari rata-rata.
5.	(Maulana dkk., 2024)	Analisis Sentimen Terhadap Aplikasi Pluang	Perkembangan teknologi yang sangat cepat telah	*Naïve Bayes* dan *Support*	Hasil penelitian memperlihatkan bahwa model SVM menunjukkan

Tabel 2. 5 State of The Art (Lanjutan 4)

No	Peneliti	Judul	Permasalahan	Algoritma	Hasil Penelitian
		Menggunakan Algoritma *Naive Bayes* dan *Support Vector Machine* (SVM)	mengubah perilaku masyarakat dalam berinvestasi. Banyak perusahaan berupaya menghadirkan platform digital untuk terus melayani konsumen, khususnya dalam bidang investasi. Seiring meningkatnya minat masyarakat terhadap investasi, jenis dan pilihan instrumen investasi yang tersedia pun menjadi semakin beragam.	*Vector Machine* (SVM)	kinerja yang lebih unggul dibandingkan model *Naive Bayes*. Secara rinci, model SVM mencapai akurasi 99,50%, *precision* 99,67%, *recall* 99,33%, dan *F1-score* 99,50%. Sementara itu, model *Naive Baye*s memperoleh akurasi 99,25%, p*recision* 99,44%, *recall* 99,06%, dan *F1-score* 99,25%.
6.	(Arinal & Purnomo, 2023)	Optimasi *Metode Decision Tree* Menggunakan *Particle Swarm Optimization* Untuk Analisis Sentimen *Review Game GTA V Roleplay*	*Twitter* adalah platform komunikasi yang memungkinkan penggunanya untuk saling bertukar informasi sekaligus berfungsi sebagai sarana hiburan. *Game GTA V Roleplay* banyak menjadi perbincangan dan	*Decision Tree* Menggunakan *Particle Swarm Optimization*	Hasil pengujian menggunakan metode *Decision Tree* menunjukkan akurasi sebesar 81,20%. Setelah penerapan seleksi fitur dengan *Particle Swarm Optimization* (PSO), akurasi meningkat menjadi 83,63%. Dari analisis sentimen, diperoleh 365 ulasan positif dan 71 ulasan negatif. Berdasarkan

Tabel 2. 6 State of The Art (Lanjutan 5)

No	Peneliti	Judul	Permasalahan	Algoritma	Hasil Penelitian
			direkomendasikan karena gaya bermainnya yang unik dan santai. Hal ini menimbulkan beragam tanggapan dari pengguna *Twitter* mengenai game tersebut. Penelitian ini bertujuan untuk menganalisis sentimen dari ulasan *game GTA V Roleplay* menggunakan metode *Decision Tree.*		dominasi jumlah sentimen positif, dapat disimpulkan bahwa opini pengguna *Twitter* terhadap *game GTA V Roleplay* cenderung positif.
7.	(Herliana & Muawiyah, 2024)	Komparasi Optimasi Analisis Sentimen *Cyberbullying* Pada Instagram Berbasis *Particle Swarm Optimization*	Sejak pandemi COVID-19 melanda, sekitar 78,19% masyarakat Indonesia memanfaatkan internet sebagai penunjang utama aktivitas sehari-hari. Kondisi ini membuat sebagian besar interaksi manusia berlangsung secara daring, termasuk untuk mengekspresikan	*Naïve Bayes*	Hasil penelitian menunjukkan bahwa kombinasi metode PSO dengan SVM menghasilkan kinerja yang lebih baik, dengan akurasi mencapai 78,60% serta dukungan *precision* kelas sebesar 100%. Sementara itu, metode *Naïve Bayes* hanya memperoleh akurasi 78,00% dengan dukungan *precision* kelas sebesar 99,74%.

Tabel 2. 7 State of The Art (Lanjutan 6)

No	Peneliti	Judul	Permasalahan	Algoritma	Hasil Penelitian
			diri. Media sosial, seperti Instagram, menjadi platform pilihan banyak orang di Indonesia untuk menyalurkan berbagai bentuk aspirasi. Namun, peningkatan jumlah unggahan di media sosial juga berdampak pada meningkatnya kasus perundungan daring, yang dikenal sebagai *cyberbullying*.		
8.	(Jatmiko dkk., 2022)	Optimasi *Naïve Bayes* dengan *Particle Swarm Optimization* Untuk Analisis Sentimen Formula E-Jakarta	Pemerintah Kota Jakarta berencana menggelar ajang balap Formula E sebagai upaya mempromosikan mobil listrik sebagai kendaraan masa depan. Namun, rencana ini tertunda akibat pandemi COVID-19 yang melanda kota tersebut.	*Naïve Bayes*	Penerapan optimasi *Particle Swarm Optimization* (PSO) pada metode *Naive Bayes* menunjukkan peningkatan kinerja model, dengan akurasi sebesar 89,16%, *precision* 91,10%, r*ecall* 86,81%, dan AUC sebesar 0,690.

Tabel 2. 8 State of The Art (Lanjutan 7)

No	Peneliti	Judul	Permasalahan	Algoritma	Hasil Penelitian
			Penundaan tersebut menimbulkan kontroversi di kalangan masyarakat di media sosial, karena meskipun kondisi Jakarta sedang terdampak pandemi, pemerintah tetap berkomitmen membayar biaya penyelenggaraan Formula E yang cukup besar.		
9.	(Yoga Religia & Amali, 2021)	Perbandingan Optimasi *Feature Selection* pada *Naïve Bayes* untuk Klasifikasi Kepuasan *Airline Passenger*	Sebagai negara kepulauan, Indonesia memerlukan sistem transportasi yang memudahkan mobilitas antar pulau, salah satunya melalui jalur udara. Hal ini menjadi peluang besar bagi perusahaan maskapai penerbangan untuk mengembangkan layanan mereka. Agar pelanggan tetap memilih maskapai	*Feature Selection* dan *Naïve Bayes*	Hasil pengujian menunjukkan bahwa model terbaik untuk klasifikasi data kepuasan penumpang maskapai (*Airline Passenger Satisfaction)* diperoleh dengan menggunakan algoritma *Naïve Bayes* yang dioptimasi dengan *Particle Swarm Optimization* (PSO). Model ini mencapai akurasi sebesar 86,13%, *precision* 87,90%, *recall* 87,29%, dan AUC sebesar 0,923.

Tabel 2. 9 State of The Art (Lanjutan 8)

No	Peneliti	Judul	Permasalahan	Algoritma	Hasil Penelitian
			tertentu, perusahaan harus menghadirkan pelayanan berkualitas. Penilaian kualitas layanan tidak dapat hanya dilihat dari perspektif perusahaan, tetapi harus didasarkan pada tingkat kepuasan pelanggan.		
10.	(Matondang dkk., 2024)	Analisis Sentimen Jasa Ekspedisi Pengiriman Barang Menggunakan Metode *Naive Bayes*	Berdasarkan kumpulan komentar yang diperoleh dari pihak Jalur Nugraha Ekakurir mengenai pelayanan kepada pelanggan, dilakukan analisis sentimen untuk mengklasifikasikan komentar tersebut menjadi kategori positif maupun negatif.	*Naïve Bayes*	Hasil penelitian memperlihatkan bahwa metode *Naïve Bayes* efektif dalam mengklasifikasikan komentar yang telah dikumpulkan. Nilai akurasi yang diperoleh untuk komentar positif mencapai 82,25%, sedangkan untuk komentar negatif sebesar 17,75%.

Tabel 2. 10 State of The Art (Lanjutan 9)

No	Peneliti	Judul	Permasalahan	Algoritma	Hasil Penelitian
11.	(Astuti & Astuti, 2022)	Analisis Sentimen Review Produk Skincare Dengan *Naïve Bayes Classifier* Berbasis *Particle Swarm Optimization* (PSO)	Produk perawatan kulit kini telah menjadi kebutuhan penting bagi berbagai kalangan, sehingga banyak brand bersaing untuk menarik konsumen. Namun, tidak semua produk menawarkan kualitas yang sesuai dengan ekspektasi pengguna. Konsumen cenderung mencari produk berkualitas dengan cara meninjau pengalaman orang lain. Ulasan dari pengguna sebelumnya, baik di marketplace maupun media sosial, memberikan gambaran yang memengaruhi minat dan keputusan konsumen sebelum membeli produk perawatan kulit.	*Naïve Bayes*	Penelitian ini menerapkan metode seleksi fitur *Particle Swarm Optimization* (PSO) untuk meningkatkan akurasi klasifikasi menggunakan *Naïve Bayes*. Dataset yang digunakan terdiri dari 800 ulasan, dan pengujian dilakukan dengan *10-Fold Cross Validation.* Hasil penelitian menunjukkan bahwa akurasi meningkat dari 77,96% menjadi 79,85% setelah penerapan PSO.

Tabel 2. 11 State of The Art (Lanjutan 10)

No	Peneliti	Judul	Permasalahan	Algoritma	Hasil Penelitian
12.	(Artanto, 2024)	*Support Vector Machine* Berbasis *Particle Swarm Optimization* Pada Analisis Sentimen Anggota KPPS	Pada Pemilu 2024 di Indonesia, menjadi anggota Kelompok Penyelenggara Pemungutan Suara (KPPS) menjadi topik yang menarik perhatian masyarakat, bahkan menjadi trending di media sosial X. Popularitas topik mengenai anggota KPPS ini mendorong berbagai diskusi dan respons dari pengguna platform tersebut.	*Support Vector Machine* (SVM)	Hasil pengujian model menunjukkan akurasi sebesar 70%. Analisis sentimen terhadap opini masyarakat di media sosial X mengenai anggota KPPS memperlihatkan bahwa 99% opini bersifat positif, sementara hanya 1% yang tergolong negatif.
13.	(Risawati dkk., 2020)	Optimasi Parameter PSO Berbasis SVM untuk Analisis Sentimen Review Jasa Maskapai Penerbangan Berbahasa Inggris	Dengan pesatnya kemajuan teknologi, banyak penumpang maskapai penerbangan membagikan ulasan mengenai pengalaman mereka menggunakan layanan tersebut melalui berbagai platform daring,.	*Support Vector Machine* (SVM)	*Particle Swarm Optimization* (PSO) terbukti mampu meningkatkan kinerja model SVM. Sebelum penerapan seleksi fitur menggunakan PSO, akurasi model SVM tercatat sebesar 84,25%. Setelah PSO diterapkan, akurasi meningkat menjadi 87,39%, menunjukkan peningkatan sebesar 3,14%.

Tabel 2. 12 State of The Art (Lanjutan 11)

No	Peneliti	Judul	Permasalahan	Algoritma	Hasil Penelitian
			seperti situs web maupun media sosial		
14.	(Que dkk., 2020)	Analisis Sentimen Transportasi Online Menggunakan *Support Vector Machine* Berbasis *Particle Swarm Optimization*	Fenomena transportasi online di Indonesia memunculkan berbagai masalah, termasuk kriminalitas dan penipuan, yang menimbulkan pro dan kontra di kalangan pengguna Twitter.	*Support Vector Machine* (SVM)	Analisis sentimen menunjukkan bahwa menggunakan metode SVM, opini positif mencapai 62% sedangkan opini negatif 38%. Sementara itu, dengan penerapan SVM yang dioptimasi menggunakan PSO, opini positif sebesar 53% dan negatif 47%. Hasil evaluasi menggunakan *10-Fold Cross Validation* menunjukkan bahwa model SVM memperoleh akurasi 95,46% dengan AUC 0,979 (kategori *excellent classification*), sedangkan SVM-PSO mencapai akurasi 96,04% dengan AUC 0,993 (kategori *excellent classification*).

Tabel 2. 13 State of The Art (Lanjutan 12)

No	Peneliti	Judul	Permasalahan	Algoritma	Hasil Penelitian
15.	(Penelitian yang dilakukan)	Optimasi *Naïve Bayes Classifier* menggunakan *Particle Swarm Optimization* (PSO) pada Analisis Sentimen terhadap Pelayanan Peserta BPJS Kesehatan	Belum diketahui seberapa baik kinerja algoritma *Naive Bayes Classifier* (NBC) dalam menganalisis sentimen terhadap pelayanan peserta BPJS Kesehatan, baik sebelum maupun sesudah dilakukan optimasi parameter menggunakan *Particle Swarm Optimization* (PSO). Selain itu, data sentimen yang digunakan memiliki ketidakseimbangan kelas dan belum dapat dipastikan kestabilan performanya, sehingga diperlukan penerapan SMOTE untuk menyeimbangkan data serta *Cross Validation* untuk memastikan keandalan hasil analisis.	Naïve Bayes Classifier, Particle Swarm Optimization, Cross Validation, Synthetic Minority Over-sampling Technique (SMOTE)	Hasil penelitian menunjukkan bahwa optimasi *Naive Bayes Classifier* (NBC) menggunakan PSO mampu meningkatkan akurasi dari 80,4% menjadi 80,7% dan *f1-score macro* dari 0,367 menjadi 0,500. Penerapan SMOTE berhasil memperbaiki ketidakseimbangan kelas dengan peningkatan *f1-score macro* menjadi 0,506, sementara evaluasi *Cross Validation* menghasilkan akurasi rata-rata 76,09% dan *f1-score macro* 0,520. Secara keseluruhan, model menjadi lebih seimbang dan konsisten dalam mengenali setiap kelas sentimen. Dengan demikian, metode NBC-PSO efektif untuk menganalisis persepsi publik dan mendukung peningkatan layanan BPJS Kesehatan.

Tabel 2. 14 State of The Art (Lanjutan 13)

No	Peneliti	Judul	Permasalahan	Algoritma	Hasil Penelitian
			Dengan demikian, penelitian ini perlu menguji apakah optimasi menggunakan PSO, penerapan SMOTE, dan evaluasi dengan *Cross Validation* benar-benar dapat meningkatkan kualitas dan keakuratan model NBC dalam mengklasifikasikan sentimen.		

Pada tabel tersebut merupakan rangkuman dari berbagai penelitian yang dilakukan dalam bidang analisis sentimen. Tabel tersebut menyajikan informasi mengenai peneliti, judul peneliti, permasalahan penelitian, algoritma yang dipakai pada penelitian, serta hasil dari penelitian tersebut. Penelitian-penelitian tersebut memiliki fokus yang serupa, yaitu menganalisis sentimen masyarakat terhadap suatu topik tertentu, baik itu terkait dengan pelayanan kesehatan, produk tertentu, maupun topik-topik lainnya yang menjadi perhatian masyarakat.

2.2.2 Matriks Penelitian

Tabel 2.15 Matriks Penelitian

No	Nama Penulis dan Tahun Penelitian	Judul Penelitian	Ruang Lingkup Penelitian											
			Framework/ Metode							*Tools*		Tujuan		
			Naïve Bayes	PSO	SVM	KNN	Decision Tree	Smote	Cross Validation	Python	Rapid Minner	Analisis	Perbandingan Metode	Optimasi
1.	(Matondang dkk., 2024)	Analisis Sentimen Jasa Ekspedisi Pengiriman Barang Menggunakan Metode *Naive Bayes*	✓								✓	✓		
2.	(Undamayanti dkk., 2022)	Analisis Sentimen Menggunakan Metode *Naive Bayes* Berbasis *Particle Swarm Optimization* Terhadap Pelaksanaan Program Merdeka Belajar Kampus Merdeka	✓	✓							✓			✓

Tabel 2. 15 Matriks Penelitian (Lanjutan 1)

No	Nama Penulis dan Tahun Penelitian	Judul Penelitian	Ruang Lingkup Penelitian											
			Framework/ Metode							*Tools*		Tujuan		
			Naïve Bayes	PSO	SVM	KNN	Decision Tree	Smote	Cross Validation	Python	Rapid Minner	Analisis	Perbandingan Metode	Optimasi
3.	(Taofik Safrudin dkk., 2023)	Optimasi Algoritma K-*Nearest Neighbor* Berbasis *Particle Swarm Optimization* Untuk Meningkatkan Kebutuhan Barang		✓		✓								✓
4.	(Maulana dkk., 2024)	Analisis Sentimen Terhadap Aplikasi Pluang Menggunakan Algoritma *Naive Bayes* dan *Support Vector Machine* (SVM)	✓		✓								✓	
5.	(Arinal & Purnomo, 2023)	Optimasi Metode Decision Tree Menggunakan Particle Swarm Optimization		✓			✓							✓

Tabel 2. 17 Matriks Penelitian (Lanjutan 2)

No	Nama Penulis dan Tahun Penelitian	Judul Penelitian	Ruang Lingkup Penelitian											
			Framework/ Metode							*Tools*		Tujuan		
			Naïve Bayes	PSO	SVM	KNN	Decision Tree	Smote	Cross Validation	Python	Rapid Minner	Analisis	Perbandingan Metode	Optimasi
		Untuk Analisis Sentimen Review Game GTA V Roleplay												
6.	(Herliana & Muawiyah, 2024)	Komparasi Optimasi Analisis Sentimen Cyberbullying Pada Instagram Berbasis Particle Swarm Optimization		✓										✓
7.	(Arsi dkk., 2021)	Optimasi SVM Berbasis PSO pada Analisis Sentimen Wacana Pindah Ibu Kota Indonesia		✓	✓									✓

Tabel 2. 18 Matriks Penelitian (Lanjutan 3)

No	Nama Penulis dan Tahun Penelitian	Judul Penelitian	Ruang Lingkup Penelitian											
			Framework/ Metode							*Tools*		Tujuan		
			Naïve Bayes	PSO	SVM	KNN	Decision Tree	Smote	Cross Validation	Python	Rapid Minner	Analisis	Perbandingan Metode	Optimasi
8.	(Jatmiko dkk., 2022)	Optimasi *Naïve Bayes* dengan *Particle Swarm Optimization* Untuk Analisis Sentimen Formula E-Jakarta	✓	✓							✓			✓
9.	(Artanto, 2024)	*Support Vector Machine* Berbasis *Particle Swarm Optimization* Pada Analisis Sentimen Anggota KPPS		✓	✓						✓			✓

Tabel 2. 19 Matriks Penelitian (Lanjutan 4)

No	Nama Penulis dan Tahun Penelitian	Judul Penelitian	Ruang Lingkup Penelitian											
			Framework/ Metode							*Tools*		Tujuan		
			Naïve Bayes	PSO	SVM	KNN	Decision Tree	Smote	Cross Validation	Python	Rapid Minner	Analisis	Perbandingan Metode	Optimasi
10.	(Risawati dkk., 2020)	Optimasi Parameter PSO Berbasis SVM untuk Analisis Sentimen Review Jasa Maskapai Penerbangan Berbahasa Inggris		✓	✓									✓
11.	(Que dkk., 2020)	Analisis Sentimen Transportasi Online Menggunakan *Support Vector Machine* Berbasis *Particle Swarm Optimization*		✓	✓									✓

Tabel 2.20 Matriks Penelitian (Lanjutan 5)

No	Nama Penulis dan Tahun Penelitian	Judul Penelitian	Ruang Lingkup Penelitian											
			Framework/ Metode							*Tools*		Tujuan		
			Naïve Bayes	PSO	SVM	KNN	Decision Tree	Smote	Cross Validation	Python	Rapid Minner	Analisis	Perbandingan Metode	Optimasi
12.	(Penelitian yang dilakukan)	Optimasi Naïve Bayes Classifier menggunakan Particle Swarm Optimization (PSO) pada Analisis Sentimen terhadap Pelayanan Peserta BPJS Kesehatan	✓	✓				✓	✓	✓				✓

Tabel tersebut memberikan perbandingan yang komprehensif antara penelitian terdahulu dan penelitian yang akan dilakukan. Berbagai penelitian sebelumnya telah menerapkan beragam metode optimasi dalam analisis sentimen, seperti optimasi *Naive Bayes* menggunakan *Support Vector Machine* (SVM), *K-Nearest Neighbor* (KNN), serta pendekatan yang menggabungkan *Naive Bayes Classifier* (NBC) dengan *Particle Swarm Optimization* (PSO). Meskipun PSO telah digunakan sebagai teknik optimasi untuk meningkatkan kinerja *Naive Bayes*, sebagian besar penelitian terdahulu hanya berfokus pada proses optimasi parameter dan belum mengintegrasikan tahapan pendukung lain yang penting dalam proses klasifikasi teks. Penelitian-penelitian sebelumnya juga tidak

menambahkan metode penanganan ketidakseimbangan kelas maupun teknik evaluasi kestabilan model, sehingga hasil yang diperoleh belum sepenuhnya mencerminkan performa model yang stabil dan akurat pada kondisi data nyata. Selain itu, penelitian sebelumnya dilakukan pada jenis data dan platform yang berbeda, serta belum secara khusus digunakan untuk menganalisis sentimen terkait pelayanan BPJS Kesehatan dari aplikasi X (*Twitter*).

Oleh karena itu, penelitian ini mengembangkan pendekatan yang lebih komprehensif dibandingkan penelitian terdahulu. Tidak hanya mengoptimasi *Naive Bayes Classifier* (NBC)menggunakan PSO, penelitian ini juga menambahkan *Synthetic Minority Over-sampling Technique* (SMOTE) untuk menangani ketidakseimbangan distribusi kelas serta *Stratified 10-Fold Cross Validation* untuk memastikan kestabilan dan keandalan model. Dengan mengintegrasikan NBC, PSO, SMOTE, dan *Cross Validation* pada konteks pelayanan BPJS Kesehatan, penelitian ini mengisi kekosongan yang belum disentuh oleh penelitian sebelumnya dan menghasilkan model analisis sentimen yang lebih akurat, seimbang, dan konsisten dalam performanya.