

## PERBANDINGAN KINERJA ALGORITMA *NAÏVE BAYES* DAN K-NN DALAM ANALISIS SENTIMEN ULASAN APLIKASI SIP DISDUKCAPIL JEMBER

**Sandi Dwikurniawan<sup>1</sup>, Rosita Yanuarti<sup>2</sup>, Henny Wahyu Sulisty<sup>3</sup>**

Program Studi Teknik Informatika, Fakultas Teknik, Universitas Muhammadiyah Jember

<sup>1</sup>sandidwi1711@gmail.com, <sup>2</sup>rosita.yanuarti@unmuhjember.ac.id, <sup>3</sup>henny.sulisty<sup>o</sup>@unmuhjember.ac.id

### **ABSTRACT**

*The Sistem Informasi Pelayanan also known as SIP of Disdukcapil Jember Regency is an online platform that facilitates population administration services. In this application, users can provide feedback on the services received through written reviews and star ratings ranging from 0-5. To analyze these user reviews effectively, Sentiment Analysis techniques can be employed to provide efficient insights. This study compares two classification algorithms, Naïve Bayes and K-Nearest Neighbor (K-NN), to determine which performs better in Sentiment Analysis of SIP reviews. The dataset consists of user feedback collected between January to December 2024. Model performance is evaluated using the Classification Report, Confusion Matrix, Matthews Correlation Coefficient (MCC), and ROC-AUC. The comparison aims to identify the most effective algorithm in distinguishing positive and negative sentiments in online service reviews. The results are expected to provide insights that support data-driven decision-making and contribute to improving the overall quality of public service delivery.*

**Keywords:** Comparison, *Naïve Bayes*, *K-NN*, SIP Disdukcapil Jember.

### **ABSTRAK**

Sistem Informasi Pelayanan atau yang disingkat SIP Disdukcapil Kabupaten Jember merupakan sistem layanan pengurusan administrasi kependudukan secara daring. Pada aplikasi tersebut pengguna dapat memberikan ulasan atas layanan yang didapatkan berupa teks ulasan dan *rating* dengan skala nilai 0-5 yang dipresentasikan dalam bentuk bintang. Dalam mengkaji ulasan pengguna aplikasi SIP tersebut, teknik Analisis Sentimen dapat diterapkan untuk hasil yang efisien. Tujuan penelitian ini membandingkan dua algoritma klasifikasi, yaitu *Naïve Bayes* dan *K-Nearest Neighbor (K-NN)*, dalam menganalisis sentimen ulasan layanan SIP Disdukcapil Jember. Perbandingan ini dilakukan guna mendapatkan hasil pemodelan klasifikasi yang terbaik. *Dataset* penelitian berupa ulasan pengguna periode Januari-Desember pada tahun 2024. Teknik evaluasi kinerja model dilakukan menggunakan *Classification Report*, *Confusion Matrix*, *Matthews Correlation Coefficient (MCC)*, serta *ROC-AUC*. Perbandingan ini diharapkan mampu menunjukkan algoritma yang lebih optimal dalam mengklasifikasikan sentimen positif dan negatif pada ulasan layanan daring SIP Disdukcapil Kabupaten Jember, sehingga dapat mendukung pengambilan keputusan dan peningkatan kualitas pelayanan publik.

**Kata Kunci:** Perbandingan, *Naïve Bayes*, *K-NN*, SIP Disdukcapil Jember.

### **PENDAHULUAN**

Sistem Informasi Pelayanan atau yang sering disingkat *SIP* Disdukcapil Kabupaten Jember merupakan sebuah *website* yang disediakan oleh Disdukcapil Kabupaten Jember teruntuk warga Jember agar dapat mengakses layanan administrasi kependudukan secara daring. Pada *website* tersebut terdapat fitur *Indeks Kepuasan Masyarakat* yang menampung data ulasan dari pengguna. Pada umumnya kita dapat dengan mudah membedakan ulasan positif atau tidak dari sebuah teks ulasan, sebagai contoh pada kalimat ulasan “pelayanan buruk, sangat lama”, dapat didefinisikan sebagai ulasan negatif yang berupa keluhan. Terkesan cukup mudah dalam memahami sebuah teks ulasan, akan tetapi jika jumlah data ulasan yang perlu dipahami berjumlah banyak hingga beribu-ribu data atau lebih, tentunya akan sulit dan menghabiskan banyak waktu jika ingin mengulik informasi dari

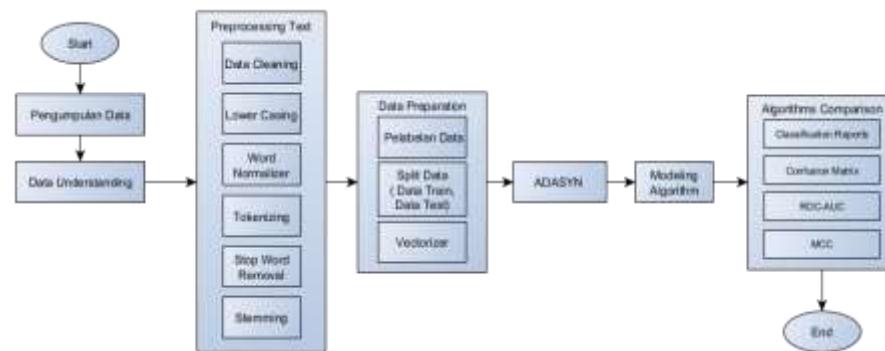
beribu-ribu atau bahkan lebih dari ulasan tersebut. Analisis Sentimen dapat digunakan untuk menjawab tantangan tersebut. Analisis Sentimen yang dapat disebut juga dengan *opinion mining* merupakan sebuah bidang studi yang menganalisa opini, sentimen, evaluasi, penilaian, sikap, dan perasaan masyarakat terhadap entitas seperti produk, pelayanan, organisasi, individu, isu, peristiwa, topik, dan atributnya (Liu, 2022). Dengan melakukan Analisis Sentimen, Disdukcapil Kabupaten Jember akan lebih mudah memperoleh informasi terkait tingkat kepuasan pengguna lebih mendalam dengan cepat.

Dalam melakukan Analisis Sentimen dibutuhkan pemilihan algoritma klasifikasi yang tepat agar mendapatkan hasil analisis yang akurat, adapun algoritma klasifikasi tersebut dapat berupa *Naïve Bayes* dan *K-Nearest Neighbor*. Algoritma *Naïve Bayes* menggunakan teori keputusan Bayesian (Surohman, dkk., 2020). Teori keputusan *Bayesian* merupakan konsep fundamental dalam analisis pola dan pengenalan pola yang mengacu pada pendekatan pemodelan probabilitas untuk membuat keputusan. Berbeda dengan *Naïve Bayes*, algoritma *K-Nearest Neighbor* mengklasifikasikan data baru dengan mencari data-data lain (tetangga terdekat) yang paling mirip. Kelas yang paling sering muncul di antara tetangga terdekat tersebut menjadi kelas pada data yang diprediksi (Prasetyo, dkk., 2023). Dengan perbedaan algoritma dalam pemrosesan tersebut, maka hasil yang didapat dari setiap model tersebut dapat berbeda juga, sebagai contoh pada penelitian Surohman, dkk. (2020) yang menguji kinerja algoritma klasifikasi *Naïve Bayes*, dan *K-Nearest Neighbor* dalam Analisis Sentimen pada ulasan penggunaan *FinTech* di Aplikasi Dana, didapat akurasi *Naïve Bayes* sebesar 84,85% dan *K-Nearest Neighbor* sebesar 82,96%. Sebagai perbandingan lain, pada penelitian Hakim Z. R., dkk. (2024) dengan penelitian Analisis Sentimen data *tweet* dengan topik Kereta Cepat Jakarta – Bandung pada aplikasi *X* didapatkan hasil akurasi *Naïve Bayes* 99.49% dan *K-Nearest Neighbor* 99.76 %. Pada penelitian Prasetyo S. D., dkk. (2021), terkait penelitian perbandingan metode *Naïve Bayes*, *K-Nearest Neighbor*, dalam Analisis Sentimen relokasi Ibu Kota Negara pada aplikasi *X*, didapatkan hasil akurasi *Naïve Bayes* sebesar 82.27% dan, *K-Nearest Neighbor* sebesar 88,12%.

Berdasarkan hasil dari penelitian terdahulu tersebut, demi mendapatkan hasil Analisis Sentimen yang baik, maka tujuan dari penelitian ini adalah melakukan perbandingan Analisis Sentimen terhadap model algoritma klasifikasi *Naïve Bayes* dengan *K-Nearest Neighbor* dengan menggunakan *dataset* ulasan layanan daring aplikasi *SIP* Disdukcapil Kabupaten Jember. *Dataset* didapatkan dengan melakukan *Web Scraping* pada halaman *Indeks Kepuasan Masyarakat* di *website SIP* Disdukcapil Kabupaten. Data penelitian yang digunakan adalah *dataset* ulasan pengguna pada tahun 2024. Hasil dari penelitian ini adalah dapat membandingkan kinerja dari model algoritma klasifikasi *Naïve Bayes* dengan *K-Nearest Neighbor* dengan membandingkan kinerjanya berdasarkan hasil *Classification Report*, *Confusion Matrix*, *MCC*, dan *ROC-AUC*. Diharapkan dengan dilakukannya perbandingan ini dapat mengetahui metode mana yang lebih baik *Naïve Bayes* atau *K-Nearest Neighbor* jika diterapkan dalam melakukan Analisis Sentimen layanan daring *SIP* Disdukcapil Kabupaten Jember.

## METODE

Agar menjaga penelitian tetap terarah pada tujuan yang telah ditetapkan, metodologi penelitian perlu dilakukan perancangan yang sistematis dan terstruktur. Adapun untuk metodelogi penelitian ini digambarkan dalam Gambar 1 untuk memudahkan peneliti dalam memahami dan menerapkannya.



**Gambar 1.** Flowchart Metodologi Penelitian

Memahami dan mendefinisikan masalah penelitian dengan jelas, memandu peneliti dalam memilih metode pengumpulan data yang tepat, membantu peneliti dalam menganalisis data secara objektif, meningkatkan kredibilitas dan validitas hasil penelitian, dan memudahkan peneliti lain untuk memahami dan berimprovisasi dalam penelitian ini merupakan manfaat lain yang bisa didapatkan dengan melakukan penelitian yang sistematis dan terstruktur.

## HASIL DAN PEMBAHASAN

### Analisis *Data Understanding*

Hasil analisa memahami karakteristik dari *dataset* yang didapatkan, dapat dilihat pada tabel 1.

**Tabel 1.** Identifikasi permasalahan dalam *dataset*

NO	Identifikasi Masalah	Proses Penyelesaian
1	Terdapat kolom data yang tidak diperlukan dalam penelitian.	Penghapusan kolom data yang tidak diperlukan.
2	Terdapat duplikasi data ulasan.	Penghapusan data duplikasi.
3	Terdapat data dengan nilai <i>null</i> pada kolom ulasan.	Penghapusan data bernilai <i>null</i> pada kolom ulasan, karena tidak relevan dalam analisis teks.
4	Terdapat karakter <i>noise</i> seperti : emoji, angka, dan tanda baca pada data teks ulasan.	Penghapusan karakter <i>noise</i> pada teks ulasan.
5	Terdapat kata dengan kesalahan proses pengetikan, dan penggunaan bahasa informal.	Normalisasi kata yang telah diidentifikasi.
6	Rating bernilai 3 yang dianggap netral	Menghapus data netral.
7	Mengelompokkan nilai rating dikarenakan berupa skala 0-5.	Nilai <i>rating</i> $< 3$ diubah menjadi negative, dan nilai <i>rating</i> $> 3$ menjadi positif.

Dari proses identifikasi permasalahan pada *dataset* tersebut, penulis dapat memetakan strategi-strategi yang perlu diterapkan dalam penyelesaian pada permasalahan *dataset* tersebut. Adapun penyelesaian dari masalah-masalah tersebut, akan diselesaikan pada proses *Preprocessing Text*.

### Hasil Proses *Preprocessing Text*

Selain menyelesaikan dari permasalahan *dataset* yang telah diidentifikasi, proses *preprocessing text* dilakukan untuk menyiapkan *dataset* agar dapat diterapkan dalam pemodelan klasifikasi *Naïve Bayes* maupun *K-NN*. Pertama-tama *dataset* hasil dari proses *scraping* akan dimuat terlebih dahulu menggunakan *library pandas*. Adapun hasil dari memuat *dataset* tersebut dapat dilihat pada gambar 2.

Unnamed: 0	#	Tanggal	Nama	Ulasan	Rating
0	0	31/12/2024 14:00	Mabruzah	mantap dan enak pokok	5.0
1	1	31/12/2024 13:59	DULASMI	惩戒	5.0
2	2	31/12/2024 13:57	Sawal	-	5.0
3	3	31/12/2024 13:56	PONIDI	-	5.0
4	4	31/12/2024 13:54	Samsul Arifin	-	5.0
5	5	31/12/2024 13:53	SAENA	-	5.0
6	6	31/12/2024 13:50	Mabruzah	pokok gempang lah	5.0
7	7	31/12/2024 13:49	Samsul Arifin	-	5.0
8	8	31/12/2024 13:48	JAHARUDDIN	TERBAIK	5.0
9	9	31/12/2024 13:43	PONIDI	-	5.0

**Gambar 1.** Contoh *dataset* masih kotor

### Hasil Proses *Data Cleaning*

Setelah identifikasi masalah pada proses sebelumnya, penerapan proses penyelesaian yang tertera pada tabel 1 akan diterapkan pada tahapan ini. Seperti yang dapat dilihat pada gambar 3, terdapat kolom data yang tidak relevan untuk proses pemodelan klasifikasi dan perlu dihapus, diantaranya adalah kolom *Unnamed:0*, #, dan Tanggal. Dikarenakan pemodelan klasifikasi pada penelitian ini berbasis teks, maka baris data pada kolom ‘Ulasan’ yang memiliki nilai *null* atau tidak bernilai juga perlu dihapus. Selain hal tersebut penghapusan data ganda juga dilakukan, seperti yang dapat dilihat pada data *index* ketiga dan keempat pada gambar 3, terdapat data yang terindikasi duplikasi yang mana juga perlu dihapus. Dalam penelitian ini akan memfokuskan pada kelas Positif dan Negatif saja, oleh karena itu data ulasan yang dianggap Netral perlu dihapus. Dikarenakan proses pelabelan berdasarkan nilai *rating* pengguna, maka nilai *rating* 3 dianggap netral dan perlu dihapus.

Karakter simbol, angka dan *emoji* juga akan dihapus dengan memanfaatkan *library regular expression* atau *regex*. Pada data teks, karakter *emoji* direpresentasikan berupa pengkodean unik berupa *Unicode*, dengan kode unik *Unicode* tersebut, proses penghapusan karakter *emoji* dapat dilakukan dengan menghapus data teks yang terdapat karakter *Unicode* tersebut. Adapun contoh karakter *Unicode* tersebut dapat berupa seperti \U0001F600-\U0001F64F, kode *Unicode* tersebut merepresentasikan karakter *emoji* 😊, 😃, 😄, 😊, 😔, 😊, 🎉, dll. Setelah dilakukan pembersihan data tersebut *dataset* menjadi berkurang dari yang awalnya 35.332 berkurang menjadi 9.188. Untuk contoh data yang telah dibersihkan dapat dilihat pada gambar 4.

	Nama	Ulasan	Rating
0	Mabruzah	mantap dan enak pokok	5.0
1	Mabruzah	pokok gempang lah	5.0
2	JAHARUDDIN	TERBAIK	5.0
3	Muhammad Syaif ***	Alhamdulillah dengan adanya aplikasi ini mempe...	5.0
4	DIAN MAYASARI	Joss Pelayanan semudah dan se simpel ini kuran...	5.0
5	PONIDI	Lambat	1.0
6	SAIFUL AZIZ	Sangat membantu dan memuaskan	5.0
7	MOHAMMAD RIZAL ***	Semoga pelayanari online dapat berjalan dengan ...	5.0
8	NILA ADIAS PUT ***	Agak susah upload file berkas yang diajukan ka...	3.5

**Gambar 2.** Contoh hasil *data cleaning*

### Hasil Proses *Lowercasing*

Pada tahapan ini data yang telah dibersihkan sebelumnya, pada data kolom ulasan dilakukan konversi untuk merubah karakter menjadi huruf kecil. Hal ini dilakukan untuk memastikan konsistensi dan mengurangi variasi yang disebabkan oleh perbedaan penggunaan huruf besar dan kecil pada *dataset*. Contoh hasil *lowercasing* dapat dilihat pada gambar 4.

	Nama	Ulasan	Rating
0	Mabruzah	mantap dan enak pokok	5.0
1	Mabruzah	pokok gempang lah	5.0
2	JAHARUDDIN	terbaik	5.0
3	Muhammad Syaif ***	alhamdulillah dengan adanya aplikasi ini mempe...	5.0
4	DIAN MAYASARI	joss pelayanan semudah dan se simpel ini kuran...	5.0
5	PONIDI	lambat	1.0
6	SAIFUL AZIZ	sangat membantu dan memuaskan	5.0
7	MOHAMMAD RIZAL ***	semoga pelayanari online dapat berjalan dengan ...	5.0
8	NILA ADIAS PUT ***	agak susah upload file berkas yang diajukan ka...	3.5
9	ABDUL MUIS	sangat puas pelayanannya bagus	5.0

**Gambar 4.** Contoh hasil *lowercasing*

## Hasil Proses *Word Normalizer*

Pada tahapan *word normalizer* atau normalisasi kata, setiap kata pada data kolom ulasan yang memiliki kesalahan proses ketik dan penggunaan kata informal akan diperbaiki. Adaupun contoh beberapa kata yang perlu diperbaiki dapat dilihat pada tabel 2.

**Tabel 2.** Contoh normalisasi kata

No	Kata Asal	Hasil Normalisasi
1	Mantab	Mantap
2	Mntap	Mantap
3	Mntp	Mantap
4	Tp	Tapi
5	Kk	Kartu Keluarga
6	Yg	Yang
7	Diengan	Dengan
Dst..	Dst...	Dst...

## Hasil Proses *Tokenizing*

Tahapan *tokenizing* atau tokenisasi dilakukan dengan memecah tiap kata yang ada pada data kolom ulasan ke dalam bentuk token. Hal ini akan membuat data akan lebih mudah untuk diolah dalam tahapan lanjutan di *preprocessing text*. Adapun untuk contoh hasil proses tokenisasi dapat dilihat pada gambar 5. Setiap kata pada token-token tersebut dipisahkan oleh karakter koma ( , ).

```

0           [mantap, dan, enak, pokok]
1           [pokok, gampang, lah]
2           [terbaik]
3   [alhamdulillah, dengan, adanya, aplikasi, ini, ...]
4   [joss, pelayanan, semudah, dan, se, simpel, in...]
5           [lambat]
6           [sangat, membantu, dan, memuaskan]
7   [semoga, pelayanan, online, dapat, berjalan, d...]
8   [agak, susah, upload, file, berkas, yang, diaj...]
9           [sangat, puas, pelayanan, nya, bagus]

```

**Gambar 3.** Contoh hasil proses tokenisasi

## Hasil Proses *Stop Word Removal*

Tahapan menghilangkan *stop word* dilakukan dengan tujuan agar dapat menghilangkan kata yang tidak relevan juga. Kata-kata yang tidak relevan yang termasuk ke dalam *stop word* dapat berupa seperti kata “yang”, “di”, “dan”, dll. Untuk menghapus kata *stop word* diperlukan kumpulan kata *stop word* tersebut agar dapat diidentifikasi, dalam hal ini *library Sastrawi* dengan *class StopWordRemover* dapat menjadi solusi tersebut. Contoh dari hasil proses penghapusan *stop word* dapat dilihat pada gambar 6.

```

0           [mantap, enak, pokok]
1           [pokok, gampang, lah]
2           [terbaik]
3   [alhamdulillah, adanya, aplikasi, mempermudah, ...]
4   [joss, pelayanan, semudah, se, simpel, kurang, ...]
5           [lambat]
6           [sangat, membantu, memuaskan]
7   [semoga, pelayanan, online, berjalan, lancar, ...]
8   [susah, upload, file, berkas, diajukan, karna, ...]
9           [sangat, puas, pelayanan, nya, bagus]

```

**Gambar 4.** Contoh hasil proses *stop word removal*

Pada gambar 5 dan gambar 6 dapat dilakukan perbandingan, pada gambar 6 untuk data nomor 0 terdapat penghapusan kata “dan” yang mana kata tersebut termasuk dalam pustaka *stop word* *Sastrawi*.

## Hasil Proses *Stemming*

Pada tahapan *stemming*, setiap kata pada data sebelumnya akan diubah ke dalam bentuk dasarnya. Dengan merubah kata ke dalam bentuk dasarnya, data yang memiliki makna yang sama akan menjadi lebih seragam. Sebagai contoh kata dasar “layan” dapat menjadi kata “pelayanan” maupun “layanan”, kedua kata tersebut memiliki makna yang sama meskipun memiliki perbedaan dalam penggunaan kalimatnya, namun pada konsep *Machine Learning* kata “pelayanan” dan kata “layanan” akan dianggap sebagai entitas yang berbeda jika tidak dilakukan proses *stemming*. Sama seperti pada tahapan *stop word*, proses ini membutuhkan pustaka yang mengandung data referensi bentuk kata dasar dari kata yang telah mengalami imbuhan maupun perubahan bentuk. Hal ini juga dapat dipenuhi oleh *library sastrawi* dengan menggunakan *class StemmerFactory*. Contoh hasil dari perubahan ke dalam bentuk kata dasar tersebut dapat dilihat pada gambar 7.

```

: 0 [mantap, enak, pokok]
1 [pokok, gampang, lah]
2 [baik]
3 [alhamdulillah, ada, aplikasi, mudah, buat, kar...
4 [jos, layan, mudah, se, simpel, kurang, apa, c...
5 [lambat]
6 [sangat, bantu, muas]
7 [moga, layan, online, jalan, lancar, baik, cep...
8 [susah, upload, file, berkas, aju, karna, ukur...
9 [sangat, puas, layan, nya, bagus]
10 [sip]
11 [sip]
12 [sangat, bantu]
13 [moga, lebih, cepat]

```

**Gambar 5.** Contoh hasil proses *stemming*

Setelah merubah kata ke dalam bentuk dasarnya, langkah selanjutnya adalah merubah kembali bentuk data yang telah menjadi bentuk token hasil proses *tokenizing* sebelumnya ke dalam bentuk normalnya. Untuk contoh hasilnya dapat dilihat pada gambar 8.

	Nama	Rating	Ulasan
0	Mabruzah	5.0	mantap enak pokok
1	Mabruzah	5.0	pokok gampang lah
2	JAHARUDDIN	5.0	baik
3	Muhammad Syaif ***	5.0	alhamdulillah ada aplikasi mudah buat kartu kel...
4	DIAN MAYASARI	5.0	jos layan mudah se simpel kurang apa cobak
5	PONIDI	1.0	lambat
6	SAIFUL AZIZ	5.0	sangat bantu muas
7	MOHAMMAD RIZAL ***	5.0	moga layan online jalan lancar baik cepat muda..
8	NILA ADIAS PUT ***	3.5	susah upload file berkas aju karna ukur file
9	ABDUL MUIS	5.0	sangat puas layan nya bagus
10	ARIFIN	5.0	sip

**Gambar 6.** Contoh hasil proses pengembalian data ke bentuk normal

## Hasil Proses *Data Preparation*

Setelah melaksanakan tahapan-tahapan *preprocessing text* sebelumnya, maka data bersih yang siap digunakan dalam pemodelan algoritma klasifikasi telah didapatkan. Tahapan-tahapan pada proses *Data Preparation* adalah melakukan proses pelabelan data, melakukan proses *splitting data*, dan

menerapkan proses *vectorizer*.

## Hasil Proses Pelabelan Data

Sebelum melakukan pemodelan algoritma klasifikasi, data pada kolom *rating* direpresentasikan menjadi dua kelas yaitu positif dan negatif. Untuk data dengan nilai positif didapatkan melalui nilai *rating* dimana rentang nilainya lebih dari tiga. Nilai negatif didapatkan melalui nilai *rating* yang nilainya kurang dari tiga. Untuk data yang bernilai ‘Positif’ dipresentasikan menjadi nilai ‘1’, sedangkan untuk nilai ‘Negatif’ dipresentasikan menjadi nilai ‘0’. Adapun untuk data yang bernilai ‘1’ atau positif berjumlah 8.633 dan untuk data yang bernilai ‘0’ atau negatif berjumlah 555.

	Nama	Rating	Ulasan
0	Mabruzah	1	mantap enak pokok
1	Mabruzah	1	pokok gampang lah
2	JAHARUDDIN	1	baik
3	Muhammad Syaif ***	1	alhamdulillah ada aplikasi mudah buat kartu kel...
4	DIAN MAYASARI	1	jos layan mudah se simpel kurang apa cobak
5	PONIDI	0	lambat
6	SAIFUL AZIZ	1	sangat bantu muas
7	MOHAMMAD RIZAL ***	1	moga layan online jalan lancar baik cepat muda...
8	NILA ADIAS PUT ***	1	susah upload file berkas aju karna ukur file
9	ABDUL MUIS	1	sangat puas layan nya bagus
10	ARIFIN	1	sip

**Gambar 7.** Hasil proses pelabelan

## Hasil Proses *Splitting Data*

Setelah melakukan pelabelan data pada data di kolom *rating*, langkah selanjutnya adalah membagi data menjadi dua bagian yaitu *data testing* dan *data training*. Dalam pembagiannya dibagi menjadi empat skenario yaitu:

1. Skenario A dengan pembagian data 75% *training* dan 25% *testing*,
2. Skenario B dengan pembagian data 80% *training* dan 20% *testing*,
3. Skenario C dengan pembagian data 85% *training* dan 15% *testing*,
4. Skenario D dengan pembagian data 90% *training* dan 10% *testing*.

## Hasil Proses *Vectorizer*

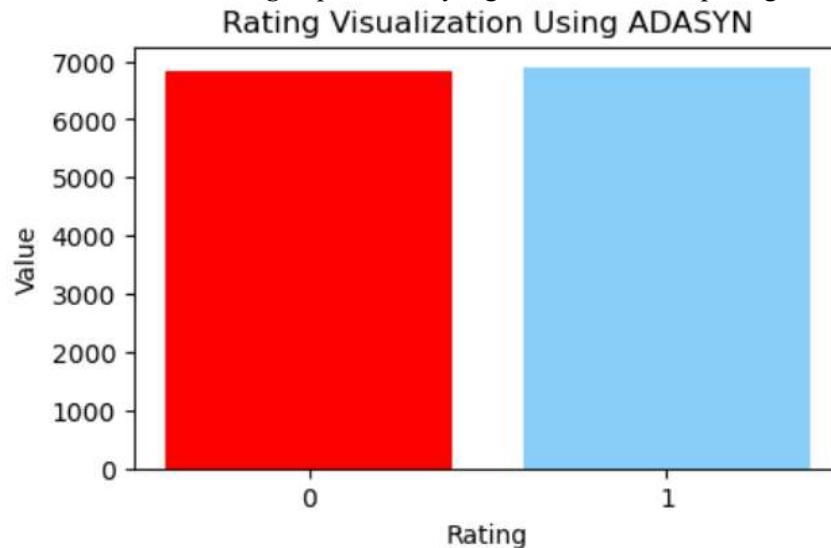
Langkah yang diambil setelah melakukan pembagian *data testing* dan *data training* adalah dengan melakukan pembobotan kata atau *vectorizer*. Adapun contoh dari hasil proses *vectorizer* dapat dilihat pada gambar 10.

0	ac	aces	acu	ad	ada	adamohon	adm	admduk	admhn	admuk
1	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
2	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
3	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
4	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
***	***	***	***	***	***	***	***	***	***	***
7345	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
7346	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
7347	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
7348	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
7349	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
	ying	yo	yogyakarta	you	yra	yuk	zainal	zaman	zkip	zksd
0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
1	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
2	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
3	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
4	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
***	***	***	***	***	***	***	***	***	***	***
7345	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
7346	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
7347	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
7348	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
7349	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0

**Gambar 8.** Contoh hasil proses *vetrozier*

### Hasil Proses *ADASYN*

Proses selanjutnya setelah *vectorizer* adalah proses *data balancing*, proses ini dilakukan dengan menggunakan metode *ADASYN*. Dengan melakukan proses *ADASYN*, model *dataset* yang seimbang dapat diciptakan sehingga dapat menghindari klasifikasi yang bias dimana cenderung pada kelas mayoritas. Untuk hasil *data balancing* dapat dilihat yang divisualisasikan pada gambar 11.



**Gambar 9.** Hasil proses *ADASYN*

### Hasil Pemodelan Algoritma Klasifikasi

Pemodelan algoritma klasifikasi dilakukan pada empat skenario pembagian data latih dan data uji, sesuai pada tahapan *splitting data* sebelumnya. Keempat Skenario tersebut adalah Skenario A dengan pembagian data 75% *training* dan 25% *testing*, Skenario B data 80% *training* dan 20% *testing*, Skenario C 85% *training* dan 15% *testing*, dan Skenario D 90% *training* dan 10% *testing*.

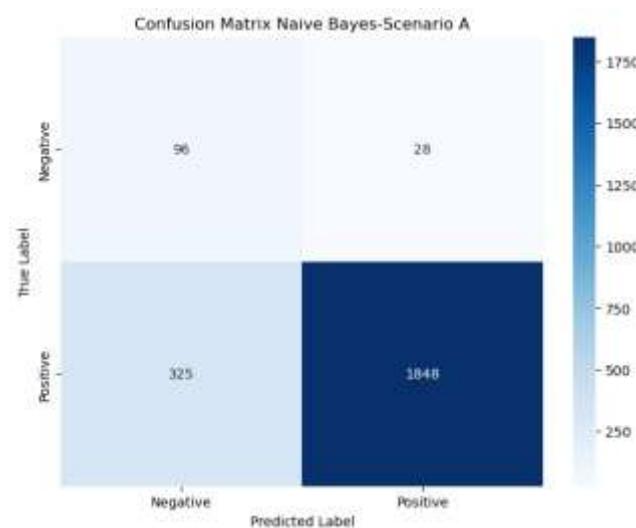
### Hasil Pemodelan *Naïve Bayes*

Pemodelan menggunakan algoritma *Naïve Bayes* dilakukan pada empat skenario pembagian data untuk menguji kinerja klasifikasi dalam menganalisis ulasan pengguna terhadap SIP Disdukcapil Kabupaten Jember. Evaluasi kinerja dilakukan dengan empat matrik analisis perbandingan, yaitu *Classification Report*, *Confusion Matrix*, *ROC-AUC*, dan *Matthews Correlation Coefficient (MCC)*. Berdasarkan hasil matrik *classification reports*, untuk hasil pemodelan *Naïve Bayes* terhadap keempat skenario pembagian tersebut dapat dilihat pada tabel 3.

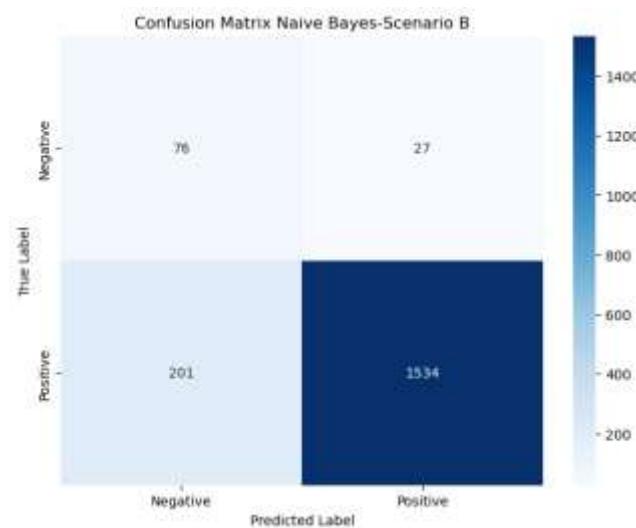
**Tabel 3.** *Classification Reports Naïve Bayes*

Terhadap Pemodelan	Accuracy	Precision	Recall	F1-Score
Naïve Bayes - Skenario A	84%	94%	85%	88%
Naïve Bayes - Skenario B	87%	94%	88%	90%
Naïve Bayes - Skenario C	87%	94%	88%	90%
Naïve Bayes - Skenario D	84%	94%	85%	88%

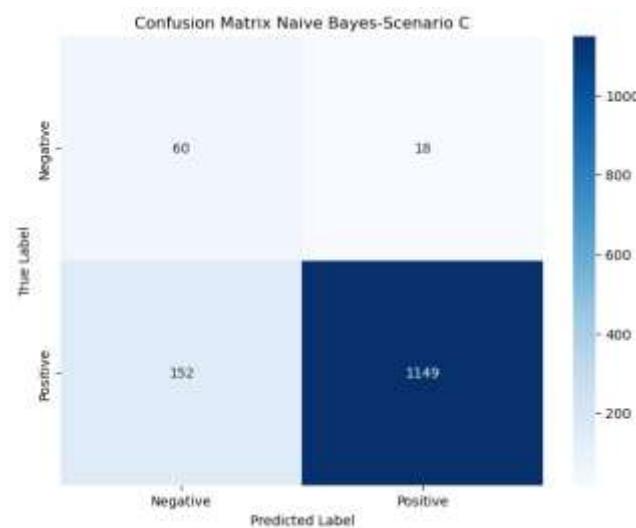
Berdasarkan matrik *classification reports* tersebut, Skenario B dan C unggul terhadap skenario lain dengan perolehan *accuracy* tertinggi yang sama-sama sebesar 87%, dan tetap unggul pada parameter lainnya dengan *precision* sebesar 94%, *recall* 88%, *F1-Score* 90%. Berdasarkan hasil matrik *confusion matrix*, untuk hasil pemodelan *Naïve Bayes* terhadap keempat skenario pembagian tersebut, masing-masing dapat dilihat pada gambar 12, gambar 13, gambar 14, gambar 15.



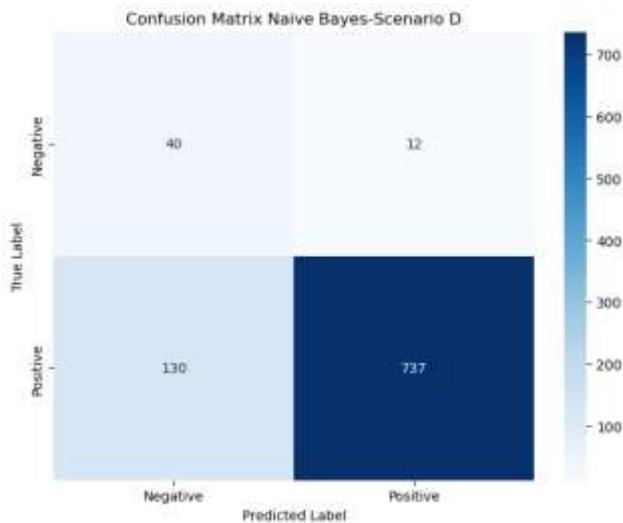
**Gambar 10.** Confusion Matrix Naïve Bayes Skenario A



**Gambar 11.** Confusion Matrix Naïve Bayes Skenario B



**Gambar 12.** Confusion Matrix Naïve Bayes Skenario C



**Gambar 13.** Confusion Matrix Naïve Bayes Skenario D

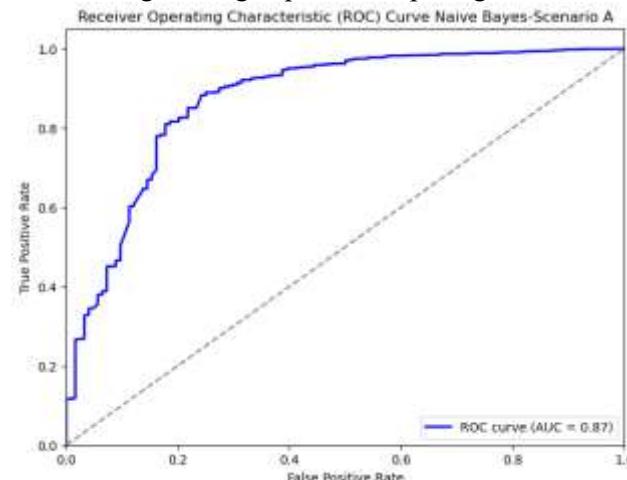
Berdasarkan hasil keempat *confusion matrix* Naïve Bayes tersebut, dapat dirangkum bahwa keempat *confusion matrix* tersebut memiliki pola kesalahan klasifikasi yang tidak jauh berbeda. Pernyataan ini dapat dibuktikan dengan menghitung pola kesalahan dengan *False Negative Rate* (*FNR*), dan *False Positive Rate* (*FPR*). Adapun hasil perbandingan *confusion matrix* tersebut dapat dilihat pada tabel 4.

**Tabel 4.** Perbandingan *confusion matrix* Naïve Bayes

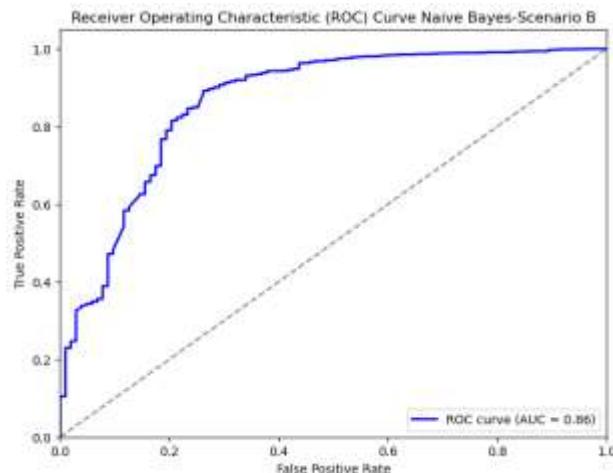
Skenario	TP	FN	TN	FP	FPR	FNR
Skenario A	1149	325	96	28	22.05%	22.58%
Skenario B	1534	201	76	27	11.59%	26.21%
Skenario C	1149	152	60	18	11.68%	23.08%
Skenario D	737	130	40	12	14.99%	23.08%

Berdasarkan tabel 4, Skenario C unggul terhadap skenario-skenario lainnya, dengan *False Positive Rate* (*FPR*) 11.68% dan *False Negative Rate* (*FNR*) 23.08%. Skenario B sedikit lebih unggul dalam memprediksi data positif daripada Skenario C namun hanya berselisih 0.09%, sedangkan dalam memprediksi data negatif skenario ini 3.13% lebih buruk dibandingkan Skenario C, sehingga dapat disimpulkan Skenario B tidak lebih baik daripada Skenario C.

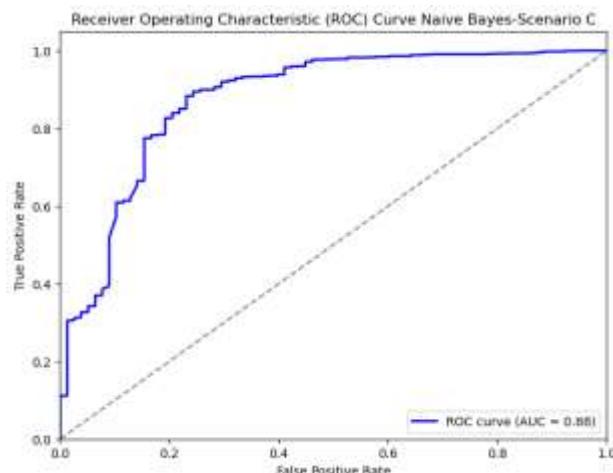
Berdasarkan hasil matrik *ROC-AUC*, untuk hasil pemodelan Naïve Bayes terhadap keempat skenario pembagian tersebut, masing-masing dapat dilihat pada gambar 16, 17, 18, dan 19.



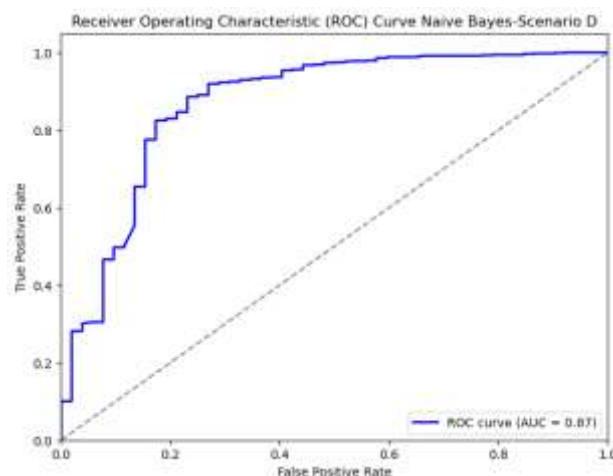
**Gambar 14.** Hasil ROC-AUC Naïve Bayes Skenario A



Gambar 15. Hasil ROC-AUC Naïve Bayes Skenario B



Gambar 16. Hasil ROC-AUC Naïve Bayes Skenario C



Gambar 17. Hasil ROC-AUC Naïve Bayes Skenario D

Berdasarkan keempat hasil matrik ROC-AUC tersebut, Skenario C unggul dengan nilai  $AUC$  0.88 terhadap skenario-skenario lain. Berdasarkan hasil matrik  $MCC$ , untuk hasil pemodelan Naïve Bayes terhadap keempat skenario pembagian tersebut, masing-masing dapat dilihat pada tabel 5.

**Tabel 4.** Hasil *MCC Naïve Bayes*

Terhadap Pemodelan	Nilai <i>MCC</i>
Naïve Bayes - Skenario A	0.36
Naïve Bayes - Skenario B	0.40
Naïve Bayes - Skenario C	0.41
Naïve Bayes - Skenario D	0.36

Berdasarkan hasil *MCC* pada tabel 5 tersebut, Skenario C unggul terhadap skenario-skenario lain dengan nilai *MCC* sebesar 0.41, disusul dengan Skenario B sebesar 0.40, dan Skenario C dan D yang sama-sama bernilai 0.36.

Kesimpulan yang dapat diambil dari analisis pada pemodelan *Naïve Bayes* tersebut, pada pembagian data Skenario C dengan pembagian data *training* 85% dan data *testing* 15% unggul terhadap skenario pembagian data yang lain. Skenario C unggul dengan nilai terbaik pada seluruh aspek matrik evaluasi yang diujikan.

Pemodelan *Naïve Bayes* pada penelitian ini dapat dikatakan cukup baik. Pada pemodelan Skenario C dengan rasio pembagian data 85% *data training* dan 15% *data testing*, model menunjukkan dapat cukup baik dalam melakukan klasifikasi pada kelas positif maupun negatif, hal ini didukung oleh nilai *Accuracy* yang didapatkan sebesar 87% dan nilai *AUC* 0.88. Meskipun dapat membedakan kelas positif dan negatif dengan baik, model *Naïve Bayes* masih terdapat beberapa kesalahan dalam proses klasifikasinya, hal ini didukung oleh hasil *MCC* yang moderat dengan nilai *MCC* 0.41 dan pola kesalahan pada *False Positive* 13 dan *False Negative* 191. Model *Naïve Bayes* pada penelitian ini terkadang melakukan kesalahan saat memprediksi kelas aktual positif yang mana diprediksi sebagai kelas negatif.

#### Hasil Pemodelan *K-NN*

Sama seperti pada pemodelan *Naïve Bayes* sebelumnya, pemodelan *K-NN* juga terapkan pada empat skenario pembagian data. Dikarenakan pada proses pemodelan *K-NN* sangat bergantung pada nilai *k* atau yang disebut *Nearest Neighbor*, maka pengujian pemodelan pada setiap skenario dilakukan sebanyak enam kali dengan nilai *k* berturut-turut 3, 5, 7, 9, 11, 13. Pada model dengan nilai *k* yang mendapat nilai *accuracy* tertinggi akan dijadikan pedoman untuk tahapan uji evaluasi selanjutnya meliputi matrik evaluasi *Confusion Matrix*, *ROC-AUC*, dan *Matthews Correlation Coefficient (MCC)*. Berdasarkan hasil pemodelan yang dilakukan dengan pembagian data Skenario A, dengan 75% *data training* dan 25% *data testing*, didapati hasil *Classification Reports* sebagai berikut yang dapat dilihat pada tabel 6.

**Tabel 5.** Hasil pemodelan *K-NN* Skenario A

<i>Nearest Neighbor (k)</i>	<i>Accuracy</i>	<i>Precision</i>	<i>Recall</i>	<i>F1-Score</i>
3	80%	94%	80%	85%
5	72%	94%	73%	80%
7	70%	94%	70%	79%
9	65%	94%	66%	75%
11	63%	94%	63%	73%
13	61%	94%	62%	72%

Berdasarkan hasil pada pemodelan *K-NN* pada skenario pembagian data Skenario A, didapati hasil terbaik pada nilai *k* sebesar 3 dengan nilai *accuracy* sebesar 80%, *precision* 94%, *recall* 80%, dan *F1-Score* 85%. Pada model dengan nilai *k* lebih dari 3 tampak terjadinya penurunan *accuracy*. Hal ini menunjukkan bahwa pada pengujian ini, seiring bertambahnya nilai tetangga terdekat yang sebagai acuan menunjukkan penurunan hasil kinerja model.

Berdasarkan hasil pemodelan yang dilakukan dengan pembagian data Skenario B, dengan 80% *data training* dan 20% *data testing*, didapati hasil *Classification Reports* sebagai berikut yang dapat dilihat pada tabel 7.

**Tabel 6.** Hasil pemodelan *K-NN* Skenario B

Nearest Neighbor (k)	Accuracy	Precision	Recall	F1-Score
3	80%	94%	80%	85%
5	74%	93%	74%	81%
7	72%	94%	72%	80%
9	70%	94%	70%	78%
11	67%	94%	68%	77%
13	66%	94%	67%	76%

Berdasarkan hasil pada pemodelan *K-NN* pada skenario pembagian data Skenario B, didapatkan hasil terbaik pada nilai *k* sebesar 3 dengan nilai *accuracy* sebesar 80%, *precision* 94%, *recall* 80%, dan *F1-Score* 85%. Hasil yang didapatkan pada Skenario B dengan *k* = 3 menunjukkan hasil kinerja model yang sama dengan Skenario A. Selain hal tersebut, pada Skenario ini juga menunjukkan hasil karakteristik yang sama seperti pada Skenario A, dimana model menunjukkan seiring bertambahnya jumlah tetangga terdekat, model cenderung mendapatkan penurunan kinerja.

Berdasarkan hasil pemodelan yang dilakukan dengan pembagian data Skenario C, dengan 85% *data training* dan 15% *data testing*, didapatkan hasil *Classification Reports* sebagai berikut, yang dapat dilihat pada tabel 8.

**Tabel 7.** Hasil pemodelan *K-NN* Skenario C

Nearest Neighbor (k)	Accuracy	Precision	Recall	F1-Score
3	80%	94%	80%	85%
5	76%	94%	77%	83%
7	74%	94%	74%	81%
9	73%	94%	74%	81%
11	71%	94%	71%	79%
13	70%	94%	70%	78%

Berdasarkan hasil pada pemodelan *K-NN* pada skenario pembagian data Skenario C, didapatkan hasil terbaik pada nilai *k* = 3 dengan nilai *accuracy* sebesar 80%, dimana hasil ini sama dengan hasil yang diperoleh Skenario A dan B. Pada model dengan nilai *k* lebih dari 3 tampak terjadinya penurunan *accuracy* sama seperti yang terjadi pada Skenario A dan B.

Berdasarkan hasil pemodelan yang dilakukan dengan pembagian data Skenario D, dengan 90% *data training* dan 10% *data testing*, didapatkan hasil *Classification Reports* sebagai berikut, yang dapat dilihat pada tabel 9.

**Tabel 8.** Hasil pemodelan *K-NN* Skenario D

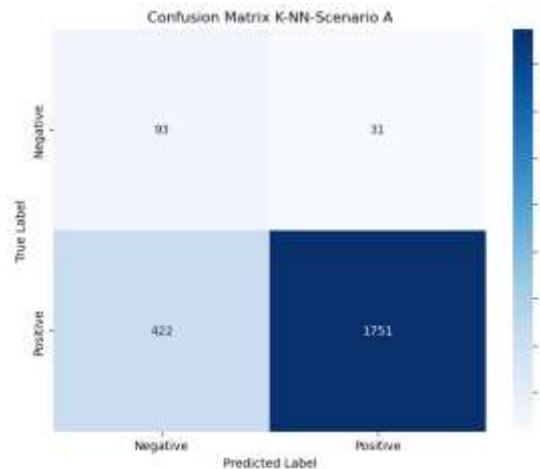
Nearest Neighbor (k)	Accuracy	Precision	Recall	F1-Score
3	79%	93%	80%	85%
5	69%	93%	70%	78%
7	66%	93%	67%	76%
9	65%	93%	66%	75%
11	63%	93%	64%	73%
13	62%	94%	63%	72%

Berdasarkan hasil pada pemodelan *K-NN* pada skenario pembagian data Skenario D, didapatkan hasil terbaik pada nilai *k* sebesar 3 dengan nilai *accuracy* sebesar 79%. Pada model didapatkan penurunan kinerja sebesar 1% jika dibandingkan dengan Skenario A, B, dan C yang mendapat nilai *accuracy* 80%.

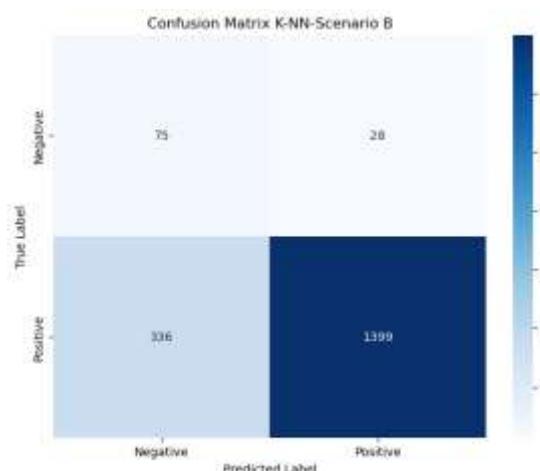
Berdasarkan hasil *Classification* pada tabel 4.6, tabel 4.7, tabel 4.8 dan tabel 4.9, didapatkan nilai *accuracy* tertinggi sebesar 80% pada matriks dengan Skenario A, B, dan C dengan nilai *k* = 3. Pada setiap skenario pembagian data, untuk *accuracy* tertinggi sama-sama didapatkan pada nilai *k* = 3. Hal ini menunjukkan bahwa nilai *k* = 3 merupakan nilai tetangga terdekat terbaik pada pemodelan algoritma *K-NN* pada penelitian ini. Berdasarkan hasil tersebut, maka model matriks dengan nilai *k* = 3 pada

setiap skenario akan digunakan pada pengujian analisis matriks lebih lanjut.

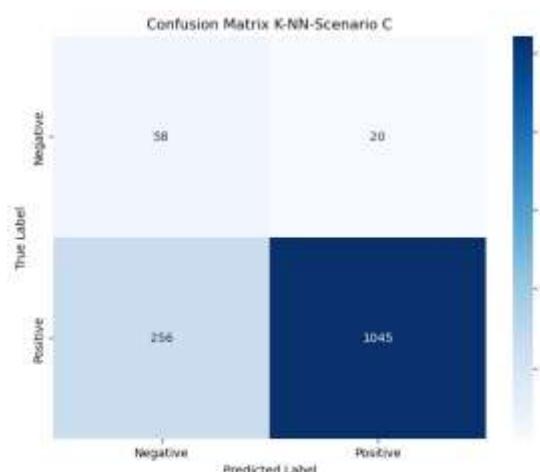
Berdasarkan hasil *confusion matrix*, untuk hasil pemodelan  $K$ -NN terhadap keempat skenario pembagian data dengan nilai  $k = 3$  tersebut, masing-masing hasil matriks dapat dilihat pada gambar 20, gambar 21, gambar 22, gambar 23.



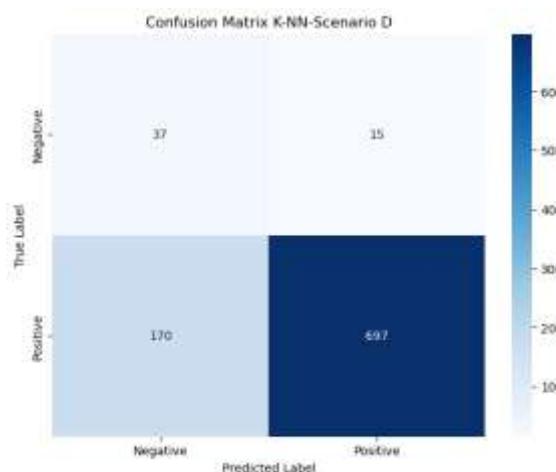
**Gambar 18.** *Confusion Matrix K-NN Skenario A k = 3*



**Gambar 19.** *Confusion Matrix K-NN Skenario B k = 3*



**Gambar 20.** *Confusion Matrix K-NN Skenario C k = 3*



**Gambar 21.** *Confusion Matrix K-NN Skenario D k = 3*

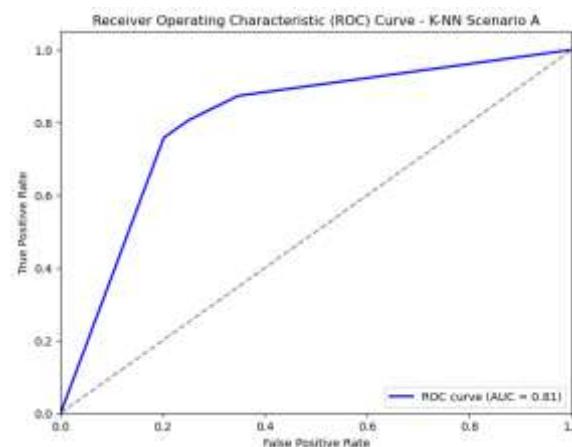
Berdasarkan hasil keempat *confusion matrix K-NN* tersebut, dapat dirangkum bahwa dari keempat hasil *confusion matrix* tersebut, pola kesalahan klasifikasi terbesar terjadi pada Skenario A dan pada Skenario B, C, D terlihat cenderung memiliki pola kesalahan klasifikasi yang tidak jauh berbeda. Pernyataan ini dapat dibuktikan dengan menghitung pola kesalahan klasifikasi dengan *False Negative Rate (FNR)*, dan *False Positive Rate (FPR)* berdasarkan *True Positive*, *False Negative*, *True Negative* dan *False Positive* yang didapatkan dari masing-masing skenario. Adapun hasil perbandingan *confusion matrix* tersebut dapat dilihat pada tabel 10.

**Tabel 10.** Perbandingan *confusion matrix K-NN k=3*

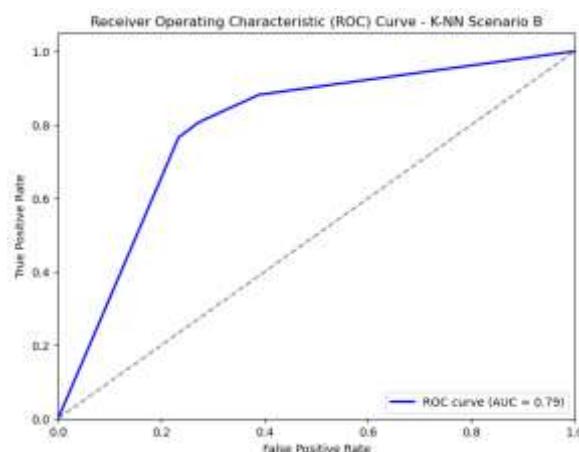
Skenario	TP	FN	TN	FP	FPR	FNR
Skenario A	1291	371	83	26	22.32%	23.85%
Skenario B	1065	273	64	15	20.40%	18.99%
Skenario C	810	197	46	10	19.56%	17.86%
Skenario D	537	133	32	7	19.85%	17.95%

Berdasarkan tabel 10, pada model pengujian Skenario C menunjukkan pola kesalahan klasifikasi terbaik, dimana capaian persentase kesalahan klasifikasi yang dihasilkan mendapatkan hasil terendah dengan nilai *FPR* sebesar 19.56% dan *FNR* sebesar 17.86%. Pada model pengujian Skenario A, terlihat model menunjukkan terjadinya pola kesalahan klasifikasi tertinggi dimana nilai *FPR* sebesar 22.32% dan *FNR* sebesar 23.85%.

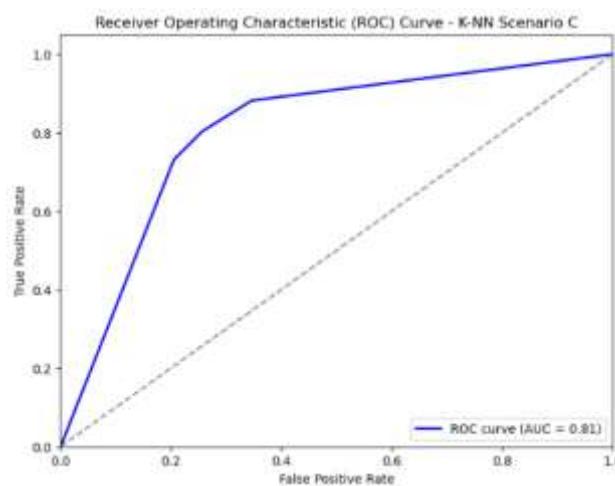
Berdasarkan hasil matriks *ROC-AUC*, untuk hasil *pemodelan K-NN* terhadap keempat skenario pembagian data dengan nilai *k=3* tersebut, masing-masing dapat dilihat pada gambar 24, 25, 26, dan 27.



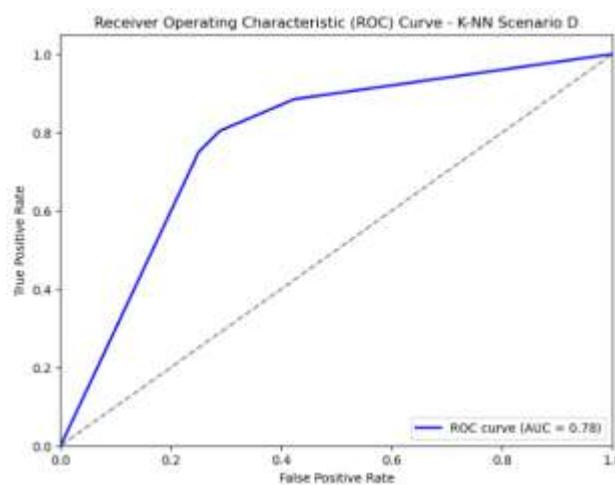
**Gambar 22.** Hasil *ROC-AUC* K-NN Skenario A



**Gambar 23.** Hasil ROC-AUC K-NN Skenario B



**Gambar 24.** Hasil ROC-AUC K-NN Skenario C



**Gambar 25.** Hasil ROC-AUC K-NN Skenario D

Berdasarkan keempat hasil matriks ROC-AUC K-NN tersebut, Skenario A, dan Skenario C unggul dengan nilai *AUC* yang sama yaitu di angka 0.81. Sedangkan pada Skenario B menunjukkan nilai *AUC* di angka 0.79, dan Skenario D menunjukkan nilai *AUC* terendah di angka 0.78. Berdasarkan hasil pengujian matriks *MCC*, untuk hasil pemodelan K-NN terhadap keempat skenario pembagian

data tersebut, masing-masing hasil matriks dapat dilihat pada tabel 11.

**Tabel 9.** Hasil *MCC K-NN*

Terhadap Pemodelan	Nearest Neighbor ( <i>k</i> )	Nilai <i>MCC</i>
<i>K-NN</i> – Skenario A	3	0.30
<i>K-NN</i> – Skenario B	3	0.29
<i>K-NN</i> – Skenario C	3	0.30
<i>K-NN</i> – Skenario D	3	0.28

Berdasarkan tabel 11 tersebut, hasil *MCC* tertinggi didapatkan pada Skenario A dan Skenario C dimana memiliki nilai yang sama yaitu di angka 0.30. Skenario B mendapatkan hasil *MCC* sebesar 0.29 dan Skenario D mendapatkan hasil *MCC* terendah yaitu di angka 0.28.

Kesimpulan yang dapat diambil dari analisis pada pemodelan *K-NN* tersebut, adalah pada pembagian data Skenario C dengan pembagian *data training* 85% dan *data testing* 15% unggul terhadap skenario pembagian data yang lain. Skenario C unggul dengan nilai terbaik pada pengujian *Accuracy*, *Recall*, *Precision*, *F1-Score*, *Confusion Matrix*, *ROC-AUC* dan *MCC*. Hal ini mengungguli Skenario A yang memiliki rasio pembagian data 80% *data training* dan 20% *data testing*. Hasil pada Skenario A memiliki nilai *Accuracy*, *Recall*, *Precision*, *F1-Score*, *AUC* dan *MCC* yang sama dengan Skenario C, akan tetapi pada model pengujian ini terlihat lebih lemah pada sisi *confusion matrix* dimana mendapatkan hasil yang paling lemah diantara Skenario lainnya dengan nilai *FPR* 22.32% dan *FNR* 23.85%.

Pemodelan *K-NN* pada penelitian ini dapat dikatakan cukup baik. Pada pemodelan Skenario C dengan rasio pembagian data 85% *data training* dan 15% *data testing*, model menunjukkan dapat cukup baik dalam melakukan klasifikasi pada kelas positif dan negatif, hal ini didukung oleh nilai *Accuracy* yang didapatkan sebesar 80% dan nilai *AUC* 0.81. Meskipun dapat membedakan kelas positif dan negatif dengan baik, model *K-NN* masih terdapat beberapa kesalahan dalam proses klasifikasinya, hal ini didukung oleh hasil *MCC* yang moderat dengan nilai *MCC* 0.30 dan pola kesalahan pada *False Positive* 10 dan *False Negative* 197. Model *K-NN* pada penelitian ini terkadang melakukan kesalahan saat memprediksi kelas aktual positif yang mana diprediksi sebagai kelas negatif

## KESIMPULAN

Berdasarkan hasil penelitian yang telah diuraikan sebelumnya, maka untuk menentukan hasil perbandingan model manakah yang lebih baik perlu dilakukan analisa dalam tabel perbandingan. Teknik perbandingan dilakukan dengan memilih salah satu model pengujian terbaik pada setiap teknik pemodelan yang dilakukan. Pada model *Naïve Bayes*, model pengujian terbaik yang dipilih adalah model pengujian yang dilakukan pada model pembagian data Skenario C. Sedangkan pada model *K-NN*, model pengujian terbaik yang dipilih adalah model pengujian yang dilakukan pada model pembagian data Skenario C dengan rasio pembagian data 85% *data training* dan 15% *data testing*, dengan nilai tetangga terdekat *k*=3.

**Tabel 12.** Hasil perbandingan model pengujian terbaik

Jenis Pemodelan	Accuracy	FN	FP	FNR	FPR	ROC-AUC	MCC
<i>Naïve Bayes</i> – Skenario C	87%	152	18	11.68%	23.08%	0.88	0.41
<i>K-NN</i> – Skenario C, <i>k</i> =3	80%	197	10	19.56%	17.86%	0.81	0.30

Berdasarkan Tabel 12, dapat dilihat bahwa kinerja algoritma *Naïve Bayes* pada Skenario C menunjukkan hasil yang lebih baik dibandingkan algoritma *K-NN* pada Skenario C, *k*=3. Pemodelan *Naïve Bayes* menunjukkan hasil yang unggul pada mayoritas matrik evaluasi, kecuali pada sisi pola

kesalahan klasifikasi pada kelas negatif yang sedikit lebih besar daripada pemodelan *K-NN*, dengan selisih *FPR* 5,22%.

Dengan demikian, dapat disimpulkan bahwa *Naïve Bayes* lebih unggul dibandingkan *K-NN* pada penelitian ini, baik dari sisi akurasi, kemampuan klasifikasi antar kelas, maupun konsistensi prediksi. Keunggulan ini menunjukkan bahwa *Naïve Bayes* lebih sesuai digunakan untuk analisis sentimen pada data uji yang digunakan pada penelitian ini, karena mampu memberikan hasil yang lebih stabil dan akurat dari pada *K-NN*.

## DAFTAR PUSATAKA

- Ahmed, G., Er, M. J., Fareed, M. M. S., Zikria, S., Mahmood, S., He, J., Asad, M., Jilani, S. F., & Aslam, M. (2022). DAD-Net: Classification of Alzheimer's Disease Using ADASYN Oversampling Technique and Optimized Neural Network. *Molecules*, 27(20), 7085. <https://doi.org/10.3390/molecules27207085>
- Chicco, D., Warrens, M. J., & Jurman, G. (2021). The Matthews Correlation Coefficient (MCC) is More Informative Than Cohen's Kappa and Brier Score in Binary Classification Assessment. *IEEE Access*, 9, 78368–78381. <https://doi.org/10.1109/ACCESS.2021.3084050>
- Dewi, C., Chen, R.-C., Christanto, H. J., & Cauteruccio, F. (2023). Multinomial Naïve Bayes Classifier for Sentiment Analysis of Internet Movie Database. *Vietnam Journal of Computer Science*, 10(04), 485–498. <https://doi.org/10.1142/S219688823500100>
- Dinata, R. K., & Hasdyna, N. (2025). SUPERVISED LEARNING Strategi Prediksi dan Klasifikasi Data. *Serasi Media Teknologi*.
- Halim, A. M., Dwifebri, M., & Nhita, F. (2023). Handling Imbalanced Data Sets Using SMOTE and ADASYN to Improve Classification Performance of Ecoli Data Sets. *Building of Informatics, Technology and Science (BITS)*, 5(1). <https://doi.org/10.47065/bits.v5i1.3647>
- Han, J., Kamber, M., & Pei, J. (2011). *Data Mining Concepts and Techniques* (Third Edition). Morgan Kaufmann.
- Hickman, L., Thapa, S., Tay, L., Cao, M., & Srinivasan, P. (2022). Text Preprocessing for Text Mining in Organizational Research: Review and Recommendations. *Organizational Research Methods*, 25(1), 114–146. <https://doi.org/10.1177/1094428120971683>
- Liu, B. (2022). *Sentiment analysis and opinion mining*. Springer Nature.
- Mohammed, A. J. (2020). Improving Classification Performance for a Novel Imbalanced Medical Dataset using SMOTE Method. *International Journal of Advanced Trends in Computer Science and Engineering*, 9(3), 3161–3172. <https://doi.org/10.30534/ijatcse/2020/104932020>
- Muliawan, J., & Dazki, E. (2023). SENTIMENT ANALYSIS OF INDONESIA'S CAPITAL CITY RELOCATION USING THREE ALGORITHMS: NAÏVE BAYES, KNN, AND RANDOM FOREST. *Jurnal Teknik Informatika (Jutif)*, 4(5), 1227–1236. <https://doi.org/10.52436/1.jutif.2023.4.5.1436>
- Pendrill, L. R., Melin, J., Stavelin, A., & Nordin, G. (2023). Modernising Receiver Operating Characteristic (ROC) Curves. *Algorithms*, 16(5), 253. <https://doi.org/10.3390/a16050253>
- Peraturan Menteri Dalam Negeri Nomor 7 Tahun 2019 Tentang Pelayanan Administrasi Kependudukan Secara Daring, Kementerian Dalam Negeri (2019).
- Peraturan Presiden Nomor 25 Tahun 2008 Tentang Persyaratan Dan Tata Cara Pendaftaran Penduduk Dan Pencatatan Sipil, Sekretariat Negara (2008).
- Prasetyo, S. D., Hilabi, S. S., & Nurapriani, F. (2023). Analisis Sentimen Relokasi Ibukota Nusantara Menggunakan Algoritma Naïve Bayes dan KNN. *Jurnal KomtekInfo*, 1–7. <https://doi.org/10.35134/komtekinfo.v10i1.330>
- Rao, N. K., Naseeba, B., N. P., N., & S., C. (2022). Web scraping (IMDB) using Python. *Telematique*, 21, 235–247.
- Sathyaranayanan, S., & Tantri, B. R. (2024). Confusion Matrix-Based Performance Evaluation Metrics. *African Journal of Biomedical Research*, 4023–4031. <https://doi.org/10.53555/AJBR.v27i4S.4345>
- Sofiya, W., & Setiawan, E. B. (2023). FINE-GRAINED SENTIMENT ANALYSIS IN SOCIAL MEDIA USING GATED RECURRENT UNIT WITH SUPPORT VECTOR MACHINE. *Jurnal Teknik Informatika (Jutif)*, 4(3), 511–519. <https://doi.org/10.52436/1.jutif.2023.4.3.855>
- Surohman, S., Aji, S., Rousyati, R., & Wati, F. F. (2020). Analisa Sentimen Terhadap Review Fintech Dengan Metode Naive Bayes Classifier Dan K- Nearest Neighbor. *EVOLUSI : Jurnal Sains Dan Manajemen*, 8(1). <https://doi.org/10.31294/evolusi.v8i1.7535>

Yadav, S. S., & Bhole, G. P. (2020). Learning from Imbalanced Data in Classification. *International Journal of Recent Technology and Engineering (IJRTE)*, 8(5), 1907–1016.  
<https://doi.org/10.35940/ijrte.E6286.018520>

Yuyun, Nurul Hidayah, & Supriadi Sahibu. (2021). Algoritma Multinomial Naïve Bayes Untuk Klasifikasi Sentimen Pemerintah Terhadap Penanganan Covid-19 Menggunakan Data Twitter. *Jurnal RESTI (Rekayasa Sistem Dan Teknologi Informasi)*, 5(4), 820–826.  
<https://doi.org/10.29207/resti.v5i4.3146>