

Analisis Sentimen Pada Ulasan Aplikasi Kredivo Dengan Algoritma SVM Dan NBC

Alman Muhammadin¹, Irwan Agus Sobari²

^{1,2}Universitas Nusa Mandiri, Teknik Informatika
Jalan Daan Mogot No. 31, Sukarasa Tangerang, Banten, Indonesia
e-mail: ¹muhammadinalman@gmail.com, ²irwan.igb@nusamandiri.ac.id

Abstrak - Analisis sentimen Review Aplikasi Kredivo merupakan salah satu contoh proses untuk mengaplikasikan dari pada metode algoritma *Support Vector Machine* (SVM) dan *Naive Bayes Classifier* dalam mengklasifikasi sentiment yang tujuannya adalah membandingkan kedua metode tersebut mana yang lebih baik. Data penelitian ini diambil dari website *Google Play Store*, data yang diambil yaitu data teks ulasan dengan jumlah **10000** ulasan. Data tersebut melewati proses Data Preprocessing dan menggunakan algoritma *Support Vector Machine* (SVM) dan *Naive Bayes Classifier* (NBC). Setelah itu dilakukan pengujian menggunakan kombinasi dari pembagian data latih dan data uji, serta menggunakan sistem set *validation*, dimana **80%** untuk data uji dan **20%** untuk data testing. Pengujian menggunakan algoritma *Support Vector Machine* menghasilkan akurasi **83,3%** dengan nilai presisi untuk kelas positif **77%** dan kelas negatif **87%** sedangkan nilai *recall* untuk kelas positif sebesar **89%** dan **73%** untuk kelas negatif. Kemudian untuk algoritma *Naive Bayes Classifier* sendiri menghasilkan nilai akurasi sebesar **80,8%** dengan nilai presisi untuk kelas positif sebesar **81%** dan untuk kelas negatif sebesar **87%**, sedangkan nilai *recall* untuk kelas positif sebesar **88%** dan untuk kelas negatif sebesar **79%**. Jadi untuk tingkat keseluruhan dapat dilihat dari nilai akurasi dengan algoritma SVM lebih tinggi dibanding *Naive Bayes Classifier*.

Kata Kunci: Analisis sentimen, *Support Vector Machine*, *Naive Bayes Classifier*

Abstract- *Sentiment analysis of the Kredivo Application Review is an example of a process to apply the Support Vector Machine (SVM) and Naive Bayes Classifier algorithm methods in classifying sentiment, the purpose of which is to compare the two better methods. This research data is taken from the Google Play Store website, the data taken is review text data with a total of 10000 reviews. The data goes through the Data Preprocessing process and uses the Support Vector Machine (SVM) and Naive Bayes Classifier (NBC) algorithms. After that, testing is carried out using a combination of the distribution of training data and test data, and using a set validation system, where 80% is for test data and 20% is for testing data. Tests using the Support Vector Machine algorithm resulted in an accuracy of 83.3% with a precision value of 77% for the positive class and 87% for the negative class, while the recall value for the positive class was 89% and 73% for the negative class. Then the Naive Bayes Classifier algorithm itself produces an accuracy value of 80.8% with a precision value for the positive class of 81% and for the negative class of 87%, while the recall value for the positive class is 88% and for the negative class is 79%. So for the overall level, it can be seen from the accuracy value with the SVM algorithm that it is higher than the Naive Bayes Classifier.*

Key Word: *Sentiment analysis, Support Vector Machine, Naive Bayes Classifier*

PENDAHULUAN

Di era teknologi saat ini pertumbuhan dan perkembangan teknologi khususnya aplikasi mobile semakin pesat, sekarang ini banyak perusahaan-perusahaan yang membuat aplikasi mobile online guna untuk mempermudah para penggunanya dalam bertransaksi. Banyak sekali aplikasi-aplikasi online yang sudah terdaftar di *Play Store*, salah satunya adalah KREDIVO. Aplikasi ini mulai banyak digunakan karena banyak sekali kelebihan yang bisa

membantu manusia seperti dalam hal kredit barang, peminjaman uang, pengisian pulsa, pengisian token listrik dll. Tapi ada beberapa kekurangan yang ada dalam aplikasi ini dan ini bisa jadi sangat merugikan para pengguna aplikasi tersebut seperti tidak masuk pembayaran cicilan yang telat masuk sistem atau penampilan aplikasi yang kurang menarik.

Untuk itu para pengguna harus hati-hati dalam memilih aplikasi yang aman dengan melihat kelayakan aplikasi tersebut seperti misalnya dengan melihat ulasan-ulasan atau komentar-komentar para



pengguna yang sudah pernah memakai aplikasi tersebut apakah aplikasinya aman atau tidak. Untuk mengatasi kekurangan ini *Google Play Store* menyediakan ruang ulasan untuk para pengguna dalam memberikan nilai atau *rating* serta memberikan *review* dari aplikasi yang mereka pakai. Dengan adanya *review* dan *rating* tersebut orang-orang yang menggunakan aplikasi tersebut bisa menilai apakah aplikasi tersebut aman atau tidak, karena peran opini atau pendapat seseorang disini sangat penting dalam kewaspadaan sebelum menggunakan aplikasi tersebut.

Opini adalah pendapat, ide atau pikiran untuk menjelaskan kecenderungan preferensi tertentu terhadap perspektif dan ideologi akan tetapi bersifat tidak objektif karena belum mendapatkan pemastian atau pengujian, dapat pula merupakan sebuah pernyataan tentang sesuatu yang berlaku pada masa depan dan kebenaran atau kesalahannya serta tidak dapat langsung ditentukan misalnya menurut pembuktian melalui induksi (Ratnawati, 2018).

Analisis sentimen sangat diperlukan dalam menyaring komentar-komentar dimedia sosial. Analisis sentimen pada komentar dilakukan untuk mengetahui komentar yang bersifat negatif dan komentar yang bersifat positif (Luqyana et al., 2018). Kalimat yang terkandung dalam ulasan atau komentar seseorang bisa saja bermakna negatif atau positif. Testimoni merupakan *opinion mining* atau termasuk dalam analisis sentimen, yaitu merupakan proses memahami, mengekstrak dan mengolah data tekstual secara otomatis untuk mendapatkan informasi yang terkandung dalam suatu kalimat opini (Ikasari, 2021).

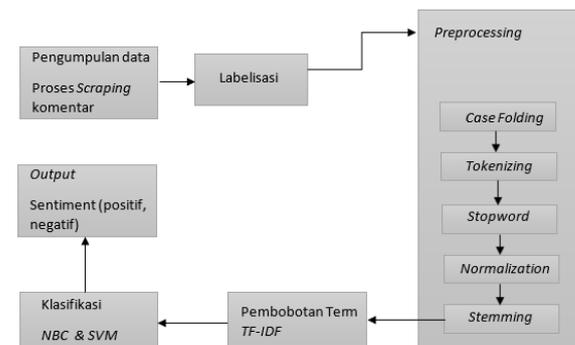
Naive Bayes Classifier (NBC) merupakan algoritma yang berfungsi dalam pengklasifikasian berdasarkan pelatihan data sudah ditraining (Ikasari, 2021). Sedangkan algoritma SVM merupakan algoritma yang mempunyai cara kerja yang sama dengan Naive Bayes namun dengan hasil yang lebih baik (Luqyana et al., 2018).

METODE PENELITIAN

Data yang digunakan pada penelitian ini merupakan data sekunder, yaitu berupa data ulasan atau *review* berbahasa Indonesia yang diberikan para pengguna Kredivo melalui situs *Google Play Store*. Jumlah populasi data yang digunakan pada penelitian ini adalah 8000 ulasan dari bulan April sampai juni 2021, isinya terdiri dari campuran ulasan yang bersifat negatif dan positif. Data ini di dapatkan dari hasil proses *Web Scraping* dari *Google Play Store*.

A	B
Komentar / Ulasan	Rating
Ini bagaimana dengan kredivo!!!! Ada penipuan mengatas bakalan kredit!!!! N	1
Kenapa saya ajukan pinjaman nda ada masuk di rekening tapi setatus nya suks	1
saya ksh bintang 3 dlu ya nnt klo bgus saya ksh bintang lg	3
Kenapa pas verifikasi wajah selalu gagal	1
Aplikasinya sangat membantu untuk orang yg membutuhkan	4
kasi 3 dulu	3
Nomor hp saya sudah mati tidak aktif lagi. Jadi saya ganti ke no hp yg baru mel	1
Apa bisa bantu, Apa yg hrus sya lakukan ,saya mau upgrade ke premium tapi say	4
mudah mudah verifikasi nya di acc..	1
kenapa ya tidak bisa isi pulsa dan tidak bisa ajukan pinjaman? padahal setiap b	1
Kecewa akun saya sudah tidak bisa dipakai, padahal gapernah bayar telat..	1
Di tolak mentah2 cih moga aja bangkrut	1
awalnya sy kasi bintang 5 karena pinjaman pertama ok,sy bayar on time. pinjan	1
Ko aneh yah tidak punya tunggakan apapun akun saya tiba tiba terblokir aplikas	1
bagus	4
aplikasi aneh.data sudah masuk..namun pas ambil foto selvie tidak bisa..giman	2
awas akun kita di sebar luaskan hat2 jngn sampe kena tipu pikir2 dulu sbilm pal	1
Jangan di download. Entar nyesel..	1
Sy heran sm kreditvo di tolak aja,seandainya sy belanja di buka lapak pakai play	3
Saya pinjem gagaln terus	4
Maaf tak kasih bintang satu karena harus upgrad ke limit yg gede, sini beraniny	1
Sistem nya ngaco liat aja sampe ada yg ketipu 700rbu gitu kalo gak becus jangar	1
Terimakasih sudah membantu ... Pelayanan baik ,,cs nya juga sopan ..untuk me	4
kenapa akun saya mau naek limit malah limit nya ilang 700rb padahal saya tidal	1
Saya minta tolong hPus data saya yang ber nama edi saya takut data saya disebi	1
ini bagaimana sih pdhal pembayaran selalu tepat waktu dan jauh? hari sebelum	1
Maaf DC lah ya ccc	1

Dataset ini merupakan data yang masih mentah dan kotor, sehingga data ini perlu diolah agar siap digunakan dalam proses klasifikasi analisis sentimen. Metode analisis data yang diajukan penulis untuk menentukan nilai akurasi dari analisis sentimen terdiri dari beberapa proses. Proses-proses yang akan dilakukan adalah sebagai berikut: kumpulan opini memasuki proses *preprocessing*, hasil dari *preprocessing* dilakukan perhitungan bobot pada setiap kata menggunakan *tf-idf*, selanjutnya dilakukan proses klasifikasi dengan algoritma *naive bayes classifier* dan *support vector machine*, pada tahap ini dilakukan untuk menentukan hasil akurasi dari analisis sentiment, Berikut adalah alur dari penelitian ini.



Adapun beberapa teori yang mendasari penelitian ini adalah sebagai berikut :

1. Analisis Sentimen

Analisis Sentimen merupan bagian dari pada proses text mining, dimana didalam mempelajari bagaimana menganalisis suatu teks, mengekstrak data tekstual seperti pendapat, sikap, evaluasi (Sodik & Kharisudin, 2021) yang tujuan adalah untuk mendapatkan informasi dari data teks untuk mengidentifikasi teks tersebut masuk kedalam kategori positif atau nefatif (Mahendrajaya et al., 2019). Inilah yang menjadikan analisis sentiment menjadi tolak ukur dalam proses pengambilan

keputusan (Arsi & Waluyo, 2021).

2. Machine Learning

Machine learning merupakan bagian dari ilmu kecerdasan buatan yang bertujuan untuk membuat alat cerdas dari proses pengetahuan berupa data sebagai bahan latih sehingga mampu membuat alat cerdas tersebut belajar sendiri dari pengetahuan atau data yang sudah diberikan.

3. Text Mining

Menurut Mustafa, text mining merupakan proses pencarian informasi pengetahuan atau sumber dalam sebuah teks, yang bertujuan untuk mendapatkan pola yang khusus dalam teks tersebut (Husada & Paramita, 2021)

4. Preprocessing

Preprocessing dilakukan untuk menghindari data yang kurang sempurna, gangguan pada data, data-data yang tidak konsisten dan data-data yang masih kotor. Adapun langkah-langkah *Preprocessing* dalam penelitian ini, yaitu *case folding*, *tokenizing*, *stopword removal*, *normalization*, *stemming*.

a. Case folding

Pada tahap ini, semua huruf akan diubah menjadi *lowercase* atau huruf kecil

b. Tokenizing

Pada tahap ini akan dilakukan pemisahan kalimat menjadi kata tunggal dan pengecekan kata dari karakter pertama sampai karakter terakhir. Apabila karakter ke-*i* bukan tanda pemisah kata seperti titik(.), koma(,), spasi dan tanda pemisah lainnya, maka akan digabungkan dengan karakter selanjutnya

c. Stopword Removal

Pada tahap ini, kumpulan kata yang telah melewati tahap *tokenizing* akan melalui tahap *stopword removal*. Setiap kata akan diperiksa, jika terdapat kata sambung, kata depan, kata ganti atau kata yang tidak ada hubungannya dengan analisis sentimen atau kata yang tidak mengandung unsur sentimen, maka kata tersebut akan dihilangkan.

d. Stemming

Pada tahap *stemming* merupakan suatu proses mengubah kata-kata yang terdapat dalam suatu dokumen ke dalam kata-kata dasarnya dengan menggunakan aturan-aturan tertentu. Proses *stemming* bahasa Indonesia dilakukan dengan menghilangkan *sufiks*, *prefix* dan *konfiks* pada dokumen.

5. TF-IDF

TF-IDF (*Term Frequency-Inverse Document Frequency*) merupakan salah satu skema atau metode untuk melakukan pembobotan kata dari proses

ekstraksi kata dengan menggunakan perhitungan kata umum di *information retrieval* Pembobotan *tf-idf* merupakan penggabungan antara *term frequency* dan *inverse document frequency* (Husada & Paramita, 2021).

$$TF - IDF_{t,d} = TF_{t,d} \times IDF_t$$
$$\text{Dimana : } IDF_t = \log \frac{N}{DF_t}$$

<i>t</i>	= Kata-kata yang dihitung
<i>d</i>	= Bobot kalimat(d)
<i>TF-IDF_{t,d}</i>	= Kalimat bobot(d) terhadap kata (t)
<i>TF_{t,d}</i>	= <i>Term Frequency</i>
<i>IDF_t</i>	= <i>Inverse Document Frequency</i>
<i>N</i>	= Jumlah kalimat
<i>DF_t</i>	= Jumlah kata yang diulang

6. Naïve Bayes Classifier

Naïve Bayes Classifier merupakan sebuah algoritma Analisa statistik, yang melakukan pengolahan data terhadap data numerik menggunakan probabilitas *Bayesian*. Klasifikasi-klasifikasi *Bayes* sendiri merupakan klasifikasi statistik yang dapat memprediksi kelas suatu anggota probabilitas.

7. Support Vector Machine

Menurut Santosa dalam penelitian (Pushpita Anna Octaviani, Yuciana Wilandari, 2014), algoritma *Support Vector Machine* (SVM) merupakan suatu Teknik yang relatif baru untuk melakukan prediksi, baik dalam kasus klasifikasi maupun regresi. Algoritma ini masuk kedalam kategori *supervised learning*, karena dalam penerapannya SVM ini perlu adanya tahap pelatihan menggunakan *sequential training* SVM dan lanjutkan tahap pengujian.

8. Ekstraksi Fitur

Setelah semua data sudah melewati tahap *preprocessing*, maka Langkah selanjutnya adalah pembuatan fitur untuk mempermudah proses klasifikasi. Pada tahap ekstraksi fitur dilakukan dua proses yaitu pembuatan *word vector* dimana dilakukan pengubahan fitur teks menjadi representasi *vector* dari pembobotan kata dengan TF-IDF.

9. Klasifikasi

Metode *naïve bayes* dan *support vector machine* merupakan contoh metode yang digunakan untuk mengklasifikasikan data komentar untuk memperoleh *sentiment analysis*. Untuk melakukan klasifikasi *sentiment* akan menggunakan data hasil proses *preprocessing* sampai dengan pembobotan kata dengan *tf-idf*. Setelah data berhasil di *training* kemudian akan dilakukan pengujian menggunakan

data test untuk menguji hasil ketepatan klasifikasi yang dilakukan.

10. Uji Mode

Pada tahap uji model ini dilakukan setelah proses training data dilakukan. Uji model sendiri dilakukan untuk mengetahui kinerja model. Jumlah data yang disajikan untuk uji model diambil dari data training sebesar 0,2% dari 0,8%. Pengambilan data dilakukan secara random atau acak dengan bantuan library dari Python. Setelah uji model dilakukan maka akan tampil seberapa besar akurasi model yang diterapkan.

11. Evaluasi Model

Evaluasi model dilakukan untuk mengetahui kinerja model. Evaluasi model dilakukan dengan cara melihat tingkat akurasi metode melalui confusion matrix dan table akurasi serta presisi untuk tiap model. Setelah data test diujikan terhadap data training, maka akan menghasilkan daftar kelas-kelas dari data test, sebut saja prediksi kelas. Kemudian prediksi kelas dibandingkan dengan kelas yang sebenarnya dari data test yang disembunyikan sebelumnya. Sehingga dapat dilihat dan dihitung nilai accuracy, precision, recall, dan f1-score.

HASIL DAN PEMBAHASAN

Pada proses klasifikasi yang digunakan, penulis menggunakan data *test* yang diambil secara acak dari 20% data *training*. Setelah proses klasifikasi dilakukan dengan menggunakan perhitungan *probabilitas* antar kalimat terhadap setiap kelas, maka disitu kita bisa mendapatkan hasil jelas prediksi dari data yang dimasukkan. Setelah semua proses itu dilakukan kita menghitung performa dari algoritma yang digunakan, yaitu *naive bayes* dan *support vector machine*.

```

# Proses TF-IDF
from sklearn.feature_extraction.text import CountVecorizer
from sklearn.feature_extraction.text import TfidfVecorizer

cv = TfidfVecorizer()
text_tf = cv.fit_transform(df['ulasan'], as_type('D'))
text_tf

<1000x753 sparse matrix of type '<class 'numpy.float64'>'
with 15450 stored elements in Compressed Sparse Row format>

# Splitting data
from sklearn.model_selection import train_test_split
X_train, X_test, y_train, y_test = train_test_split(text_tf, df['ulasan'], test_size=0.2, random_state=42)

# Performa Algoritma Naive Bayes
from sklearn.naive_bayes import MultinomialNB
from sklearn.metrics import accuracy_score, precision_score, recall_score, f1_score
from sklearn.metrics import classification_report
from sklearn.metrics import confusion_matrix

clf = MultinomialNB().fit(X_train, y_train)
# lakukan prediksi pada data test
predicted = clf.predict(X_test)
    
```

1. Uji Model

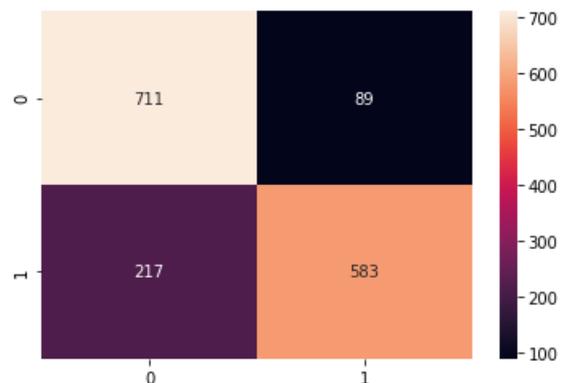
Untuk mengetahui performa dari Algoritma *Naive Bayes* dan *Support Vector Machine*, maka dilakukan pengujian terhadap model. Hasil klasifikasi akan ditampilkan dalam bentuk *confusion matrix*. Tabel *confusion matrix* terdiri dari kelas *predicted* dan kelas *actual*. Model *confusion matrix* 2x2.

Confusion Matrix

	Actual Value	
Predicted values	TP (True Positif)	FP (False Positif)
	FN (False Negatif)	TN (True Negatif)

$$accuracy = \frac{TP + TN}{TP + TN + FN + FP}$$

Dari contoh *confusion matrix* klasifikasi biner diatas maka dengan menghitung nilai accuracy dapat menjawab pertanyaan “Berapa persen sistem dalam menjawab ketepatan analisis sentiment”. Berikut adalah gambar dari *confusion matrix* dari kedua algoritma yang digunakan.



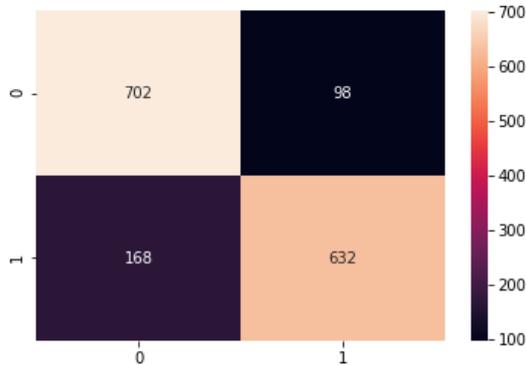
Confusion Matrix Naive Bayes

$$accuracy = \frac{TP + TN}{TP + TN + FN + FP}$$

$$accuracy = \frac{711 + 583}{711 + 583 + 217 + 89}$$

$$accuracy = \frac{1294}{1600} = 0,80875$$

$$accuracy = 0,80 \times 100\% = 80,875\%$$



Confusion Matrix SVM

$$accuracy = \frac{TP + TN}{TP + TN + FN + FP}$$

$$accuracy = \frac{702 + 632}{702 + 632 + 168 + 98}$$

$$accuracy = \frac{1334}{1600} = 0,83375$$

$$accuracy = 0,83 \times 100\% = 83,375\%$$

Dari perhitungan diatas nilai akurasi yang didapatkan adalah **0.80875** untuk *naïve bayes* dan **0.83375** untuk *support vector machine* yang dihitung berdasarkan nilai dari diagonal *confusion matrix* dibagi dengan jumlah seluruh data. Karen jumlah data tiap kelas pada data *training* tidak seimbang, maka besarnya nilai akurasi bukanlah terpenting.

2. Evaluasi Model

Evaluasi model dilakukan setelah proses uji model selesai dilakukan. Evaluasi model dilakukan untuk menghitung performa metode yang dipilih. Pada uji model yang dilakukan menghasilkan *confusion matrix* dengan dimensi 2x2, ada dua tabel *confusion matrix* yang di hasilkan dari uji model, yaitu tabel *confusion* untuk algoritma *naïve bayes* dan *confusion matrix* untuk *support vector machine*.

Untuk menghitung nilai *presisi*, *recall*, dan *f-1 score* pada sistem dapat menggunakan metode pada gambar berikut.

```
# f1_score
print("f1 score hasil prediksi adalah: ")
print(f1_score(y_test, predict, average="macro"))

# accuracy score
print("accuracy score hasil prediksi adalah: ")
print(accuracy_score(y_test, predict))

# precision score
print("precision score hasil prediksi adalah: ")
print(precision_score(y_test, predict, average="macro"))

# recall score
print("recall score hasil prediksi adalah: ")
print(recall_score(y_test, predict, average="macro"))

print(f'confusion matrix: \n{confusion_matrix(y_test, predict)}')
print("----- \n")
print(classification_report(y_test, predict, zero_division=0))
```

Code Presisi, Recall, dan f-1 score

Selanjutnya untuk performa metode klasifikasi dari setiap kelasnya dapat dilihat melalui nilai *presisi*, *recall*, *f1-score* pada setiap kelasnya. Hasil nilai *presisi*, *recall*, dan *f1-score* memiliki nilai sebesar 0 – 1. Semakin tinggi nilainya mendekati 1 maka hasilnya semakin baik. Hasil keseluruhan proses evaluasi model dapat dilihat pada gambar berikut

```
f1 score hasil prediksi adalah:
0.7921195652173912
accuracy score hasil prediksi adalah:
0.80875
precision score hasil prediksi adalah:
0.8675595238095238
recall score hasil prediksi adalah:
0.72875
confusion matrix:
[[711 89]
 [217 583]]
-----
              precision    recall  f1-score   support

   0             0.77       0.89       0.82         800
   1             0.87       0.73       0.79         800

 accuracy
 macro avg       0.82       0.81       0.81        1600
weighted avg       0.82       0.81       0.81        1600
```

presisi, recall, f1-score NBC

```
f1 score hasil prediksi adalah:
0.8334311768619622
accuracy score hasil prediksi adalah:
0.83375
precision score hasil prediksi adalah:
0.8363249881908361
recall score hasil prediksi adalah:
0.83375
confusion matrix:
[[702 98]
 [168 632]]
-----
              precision    recall  f1-score   support

   0             0.81       0.88       0.84         800
   1             0.87       0.79       0.83         800

 accuracy
 macro avg       0.84       0.83       0.83        1600
weighted avg       0.84       0.83       0.83        1600
```

presisi, recall, f1-score SVM

Tabel presisi, recall, f1-score NBC

Jenis Klasifikasi	presisi	recall	f1-score
Positif	0.77	0.89	0.82
Negatif	0.87	0.73	0.79

Hasil dari evaluasi model dapat dilihat dari nilai presisi dan recall di setiap kelasnya, dapat dikatakan tingkat kemampuan sistem dalam mencari ketepatan antara informasi yang diminta oleh pengguna (presisi) untuk kelas positif sebesar **77%**, untuk kelas negative **87%**. Sedangkan tingkat keberhasilan sistem dalam menemukan Kembali sebuah informasi (*recall*) untuk kelas positif sebesar **89%**, untuk kelas negatif sebesar **73%**.

Tabel presisi, recall, f1-score SVM

Jenis Klasifikasi	presisi	recall	f1-score
Positif	0.81	0.88	0.84
Negatif	0.87	0.79	0.83

tingkat kemampuan sistem dalam mencari ketepatan antara informasi yang diminta oleh pengguna (presisi) untuk kelas positif sebesar **81%**, untuk kelas negative **87%**. Sedangkan tingkat keberhasilan sistem dalam menemukan Kembali sebuah informasi (*recall*) untuk kelas positif sebesar **88%**, untuk kelas negatif sebesar **79%**. Dari temuan yang dihasilkan dalam penelitian ini system dapat menemukan, membandingkan dan mengklasifikasi jenis sentimen baru, berikut gambaran akhir dari penelitian ini dapat dilihat pada Gambar berikut.

```
[ ] review_positif = "respon toko kurang baik, jelek"
    review_negatif = "kualitas brand milik toko liverpool memang mantap banget pokoknya"

[ ] clf.predict(preprocess_data(review_positif))
array([0])

[ ] clf.predict(preprocess_data(review_negatif))
array([1])
```

Hasil Akhir Analisis Sentimen SVM

Dari gambar diatas bahwa sistem dengan menggunakan algoritma *Support Vector Machine* dapat menemukan dan mengkasifikasi sentimen tersebut masuk kedalam kategori positif atau negative, yaitu dengan di uji coba menggunakan variable dengan nama `review_positif` yang berisi sentimen negatif telah membuktikan hasil output berupa array dengan label angka 0 yang menunjukkan katategori Negatif. Sebaliknya untuk variabel dengan nama `review_negatif` yang berisis sentimen positif telah membuktikan hasil output berupa array dengan label angka 1 yang menunjukkan kategori Positif.

```
[ ] review_positif = "respon toko tim sebelah kok jelek banget sih"
    review_negatif = "kualitas brand milik toko liverpool memang mantap banget pokoknya"

[ ] clf.predict(preprocess_data(review_positif))
array([0])

[ ] clf.predict(preprocess_data(review_negatif))
array([1])
```

Hasil Akhir Analisis Sentimen NBC

Dan untuk sistem dengan menggunakan algoritma *Naïve Bayes* dapat menemukan dan mengkasifikasi sentimen tersebut masuk kedalam kategori positif atau negatif seperti algoritma SVM.

KESIMPULAN

Berdasarkan hasil pengujian algoritma *Naïve Bayes Classifier* dan *Support Vector Machine* yang telah dilakukan ada beberapa hal yang dihasilkan, antara lain :

1. Sistem anlisis sentiment pada penelitian sudah cukup baik dalam mengidentifikasi sebuah sentiment tersebut apakah bersifat positif atau negatif.
2. Pada penelitian ini, algoritma *Naïve Bayes Classifier* menghasilkan nilai akurasi sebesar **0.808** atau **80,8%**. Dan **0.833** atau **83.3%** untuk SVM. Dengan demikian algoritma SVM sedikit lebih baik.
3. Kelebihan pada penelitian ini adalah memiliki nilai akurasi, presisi, dan *recall* yang baik, sehingga sudah cukup untuk bisa digunakan dalam sebuah sistem.
4. Kekurangan pada penelitian ini nilai dari presisi, *recall* pada setiap kelas yang tidak merata. Hal itu disebabkan karena jumlah data jumlah data yang tidak merata antara kelas positif dan *negative*.

REFERENSI

Arsi, P., & Waluyo, R. (2021). Analisis Sentimen Wacana Pemindahan Ibu Kota Indonesia Menggunakan Algoritma Support Vector Machine (SVM). *Jurnal Teknologi Informasi Dan Ilmu Komputer*, 8(1), 147. <https://doi.org/10.25126/jtiik.0813944>

Husada, H. C., & Paramita, A. S. (2021). Analisis Sentimen Pada Maskapai Penerbangan di Platform Twitter Menggunakan Algoritma Support Vector Machine (SVM). *Teknika*, 10(1), 18–26. <https://doi.org/10.34148/teknika.v10i1.311>

Ikasari, D. (2021). *SENTIMENT ANALYSIS REVIEW NOVEL "GOODREADS" BERBAHASA INDONESIA MENGGUNAKAN NAÏVE BAYES*. 760–765.

Luqyana, W. A., Cholissodin, I., & Perdana, R. S. (2018). Analisis Sentimen Cyberbullying Pada

- Komentar Instagram dengan Metode Klasifikasi Support Vector Machine. *Jurnal Pengembangan Teknologi Informasi Dan Ilmu Komputer (J-PTIHK) Universitas Brawijaya*, 2(11), 4704–4713.
- Mahendrajaya, R., Buntoro, G. A., & Setyawan, M. B. (2019). Analisis Sentimen Pengguna Gopay Menggunakan Metode Lexicon Based Dan Support Vector Machine. *Komputek*, 3(2), 52. <https://doi.org/10.24269/jkt.v3i2.270>
- Pushpita Anna Octaviani, Yuciana Wilandari, D. I. (2014). *Penerapan Metode SVM Pada Data Akreditasi Sekolah Dasar Di Kabupaten Magelang*. 3(8), 811–820.
- Ratnawati, F. (2018). Implementasi Algoritma Naive Bayes Terhadap Analisis Sentimen Opini Film Pada Twitter. *INOVTEK Polbeng - Seri Informatika*, 3(1), 50–59. <https://doi.org/10.35314/isi.v3i1.335>
- Sodik, F., & Kharisudin, I. (2021). Analisis Sentimen dengan SVM , NAIVE BAYES dan KNN untuk Studi Tanggapan Masyarakat Indonesia Terhadap Pandemi Covid-19 pada Media Sosial Twitter. *Prisma*, 4, 628–634.