



Perbandingan Metode *Machine Learning* dalam Analisis Sentimen Komentar Pengguna Aplikasi **InDriver** pada *Dataset* Tidak Seimbang

Sebastianus Adi Santoso Mola*, Yufridon Charisma Luttu, Dessy Nelci Rumlaklak

Program Studi Ilmu Komputer, Fakultas Sains dan Teknik, Universitas Nusa Cendana

Naskah masuk: 9 Agustus 2023; Diterima untuk publikasi: 14 Mei 2024
DOI: 10.21456/vol14iss3pp247-255

Abstract

The InDriver service is an online transportation service that has more flexibility in price and driver choice by consumers. Various comments from InDriver service users can affect people's views, so it is necessary to carry out a sentiment analysis of these comments. The purpose of this study was to identify positive, negative and neutral sentiments in user comments and to compare the performance of classification methods. The results of analysis with unbalanced datasets show that the Support Vector Machine (SVM) and Logistic Regression methods have the highest accuracy, reaching 89%. However, quality assessment is not only based on accuracy alone. In terms of the balance between precision and recall in the minority (neutral) class, the Random Forest method shows a more balanced performance with an F1-score of 55%. After balancing the dataset with the SMOTE method, performance increases significantly for the Naïve Bayes Classifier method, especially in the neutral class for recall and F1-score metrics of 57% and 52%. In conclusion, SVM and Logistic Regression have high accuracy, but to consider the balance of precision and recall in the minority class, the Random Forest method is recommended.

Keywords: Sentiment analysis; InDriver; Machine learning, Unbalanced dataset

Abstrak

Layanan InDriver merupakan layanan transportasi daring yang memiliki fleksibilitas lebih dalam harga dan pilihan pengemudi oleh konsumen. Beragam komentar pengguna layanan InDriver dapat memengaruhi pandangan masyarakat sehingga perlu dilakukan analisis sentimen terhadap komentar-komentar tersebut. Tujuan penelitian ini adalah untuk mengidentifikasi sentimen positif, negatif, dan netral dalam komentar pengguna serta membandingkan kinerja metode klasifikasi. Hasil analisis dengan *dataset* tak seimbang menunjukkan bahwa metode *Support Vector Machine* (SVM) dan *Logistic Regression* memiliki akurasi tertinggi, mencapai 89%. Namun, penilaian kualitas tidak hanya berdasarkan akurasi semata. Dalam hal keseimbangan antara *precision* dan *recall* pada kelas minoritas (netral), metode *Random Forest* menunjukkan performa lebih seimbang dengan *F1-score* sebesar 55%. Setelah dilakukan penyeimbangan *dataset* dengan metode SMOTE, peningkatan kinerja terjadi secara signifikan untuk metode *Naïve Bayes Classifier* terutama pada kelas netral untuk metrik *recall* dan *F1-score* sebesar 57% dan 52%. Kesimpulannya, SVM dan *Logistic Regression* memiliki akurasi tinggi, tetapi untuk mempertimbangkan keseimbangan *precision* dan *recall* pada kelas minoritas, metode *Random Forest* lebih disarankan.

Kata kunci: Analisis sentimen; InDriver; Pembelajaran mesin, *Dataset* tidak seimbang

1. Pendahuluan

InDriver adalah layanan transportasi daring asal Rusia yang didirikan pada tahun 2012 dan mulai beroperasi di Indonesia pada tahun 2019 (Karja, 2019). Layanan ini menawarkan konsep kebebasan bagi penumpang dan pengemudi, yang mana keduanya dapat menawar harga secara bebas hingga diperoleh harga yang disepakati. Konsep kebebasan ini menjadi ciri khas InDriver dan memberikan

keuntungan bagi penumpang dan pengemudi, seperti fleksibilitas dalam menentukan harga yang sesuai dengan kebutuhan. Namun, konsep ini juga memiliki beberapa kekurangan, yaitu pengemudi susah mendapatkan orderan, dan persaingan harga antara pengemudi cukup ketat. Selain kekurangan tersebut, penelitian (Tuwanakotta and Tanaamah, 2022) menemukan bahwa InDriver memiliki beberapa kekurangan yang sering dikeluhkan pengguna yaitu belum adanya fitur auto bid atau penetapan batas harga maksimum secara otomatis dan sering terjadi

*)Corresponding author: adimola@staf.undana.ac.id

error seperti pada peta dan *global positioning system* (GPS) yang menyebabkan fungsi aplikasi kurang maksimal.

Penggunaan aplikasi InDriver dapat dipengaruhi oleh kualitas layanan yang ditawarkan (Sianipar, 2019). Kualitas layanan ini dibagikan oleh pengguna layanan melalui komentar dan ulasan aplikasi. Oleh karena itu, diperlukan suatu metode untuk menganalisis sentimen pengguna mengenai layanan yang disediakan oleh InDriver.

Beberapa penelitian terdahulu telah menggunakan metode *machine learning* dalam melakukan analisis sentimen terhadap ulasan aplikasi transportasi daring seperti dilakukan oleh Muttaqin dan Kharisudin, (2021) pada aplikasi Gojek dengan metode *Support Vector Machine* (SVM) dan *k-Nearest Neighbor* (k-NN), Rokhman *et al.*, (2021) dengan metode SVM dan *Decision Tree*, dan Anjasmoros *et al.*, (2020) dengan metode SVM dan *Naive Bayes Classifier* (NBC). Penelitian Sitorus, (2021) secara khusus melakukan analisis sentimen ulasan aplikasi InDriver menggunakan metode SVM dan NBC dengan keunggulan metode SVM. Lebih lanjut Putri dan Kharisudin, (2022) dalam penelitiannya analisis sentimen pada ulasan *marketplace* Tokopedia menemukan bahwa kinerja SVM lebih baik dibandingkan dengan NBC dan *Logistic Regression* (LR) dengan *dataset* tidak seimbang. Sebaliknya, dengan membandingkan metode yang sama dengan Putri dan Kharisudin, (2022), penelitian Sitompul *et al.*, (2023) menemukan bahwa NBC memiliki kinerja yang lebih baik. Selanjutnya, penelitian Anbari dan Sugiantoro, (2023) menemukan bahwa LR dan SVM memiliki akurasi yang sama dalam analisis sentimen terhadap data Twitter mengenai piala dunia 2022. Terdapat juga penelitian yang menunjukkan bahwa *Random Forest* (RF) memiliki kinerja yang lebih baik dibandingkan dengan SVM seperti dalam penelitian Adrian *et al.*, (2021) serta Fitri *et al.*, (2020).

Secara umum, berdasarkan penelitian-penelitian tersebut dapat diketahui bahwa tidak ada satu pun metode klasifikasi *supervised learning* yang unggul untuk setiap domain. Oleh karena itu, dalam penelitian ini akan dilakukan perbandingan kinerja metode LR, NBC, SVM, dan RF pada *dataset* yang tidak seimbang. *Dataset* yang tidak seimbang ditandai dengan adanya ketidakseimbangan kelas. Tantangannya adalah kurang terwakilinya kelas minoritas. Walaupun model klasifikasi secara umum bagus namun untuk kelas minoritas mungkin saja tidak (Seliya *et al.*, 2010). Ketika kelas minoritas sangat penting, kelas minoritas harus mendapat perhatian.

2. Kerangka Teori

2.1. Text Preprocessing

Text preprocessing merupakan tahapan untuk membersihkan data teks dari *noise* atau memperbaiki

kualitas data teks sehingga dapat memudahkan dalam pengolahan lebih lanjut (Priyanto and Ma'arif, 2018).

Text preprocessing yang digunakan dalam penelitian ini adalah sebagai berikut:

- 1) *Cleaning*
Cleaning merupakan proses yang dilakukan untuk mengurangi *noise* dalam sebuah data atau komentar.
- 2) *Case folding*
Case folding adalah proses untuk menyamakan bentuk teks, baik menjadi *lower case* atau *upper case*.
- 3) Normalisasi
Normalisasi merupakan proses untuk mengubah kata tak baku menjadi kata baku sehingga dapat mengurangi bias dalam proses klasifikasi.
- 4) *Tokenizing*
Tokenizing adalah proses untuk memecah kalimat komentar menjadi token atau komponen-komponen individu, di mana token tersebut akan digunakan pada proses *stemming*.
- 5) *Stemming*
Stemming merupakan tahap merubah kata berimbuhan menjadi kata dasar.
- 6) *Convert negation*
Convert negation adalah proses menambahkan *dependency marker* (DM) pada kata yang mengandung negasi, agar algoritma dapat mengenali kata tersebut sebagai kata dengan makna yang berbeda dari kata yang tidak mengandung negasi.
- 7) *Stopword removal*
Stopword removal adalah tahap untuk menghapus kata-kata yang sering muncul dan bersifat umum tetapi tidak penting, seperti kata 'ada', 'untuk', 'di', 'ke', 'dari', dan sebagainya.

2.2. Inverse Document Frequency (TF-IDF)

TF-IDF adalah suatu pendekatan yang digunakan untuk mengukur sejauh mana sebuah kata (*term*) terhubung dengan sebuah dokumen dengan memberi nilai penting kepada setiap kata. Pendekatan TF-IDF menggabungkan dua ide, yakni frekuensi kemunculan suatu kata dalam suatu dokumen dan kebalikan frekuensi kemunculan kata tersebut dalam seluruh dokumen (Herwijayanti *et al.*, 2018). Untuk mendapatkan nilai TF-IDF, harus menghitung *term frequency* (TM) dengan persamaan (1), *document frequency* (DM) dengan persamaan (2), dan *invers document frequency* (IDF) dengan persamaan (4). Normalisasi bobot TF-IDF ada pada persamaan (5).

1) Term Frequency (TF)

$$TF_{(t_i,k_j)} = \frac{\sum_{i=1}^n f(t_i,k_j)}{n} \quad (1)$$

di mana:

k : komentar

m : jumlah komentar

j : indeks dari komentar (1,2,3, ..., m)
 t : *term*
 n : jumlah *term* dalam k_j
 I : indeks dari *term* (1,2, ..., n)
 $f(t_i, k_j)$: frekuensi *term* ke- i pada komentar ke- j

2) Document Frequency (DF)

$$DF(t_i) = \sum_{j=1}^m f(t_i, k_j) \quad (2)$$

di mana DF merupakan jumlah komentar yang mengandung *term* ke- t .

3) Inverse Document Frequency (IDF)

$$IDF(t_i) = \ln\left(\frac{m+1}{1+DF(t_i)}\right) + 1 \quad (3)$$

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

$$v_{(t,k)} = TF_{(t,k)} \times IDF_{(t)} \quad (4)$$

di mana $v_{(t,k)}$ adalah nilai dari bobot TF terhadap IDF.

Normalisasi bobot TF-IDF

$$v_{norm(t,k)} = \frac{v_{(t,k)}}{\sqrt{v_{(1,k)}^2 + v_{(2,k)}^2 + \dots + v_{(n,k)}^2}} \quad (5)$$

dengan $v_{norm(t,k)}$ merupakan nilai normalisasi tiap *term*.

2.3. Klasifikasi

Klasifikasi merupakan tahap terakhir dalam *text mining*, di mana dilakukan klasifikasi terhadap data teks yang telah melalui tahap *text preprocessing* dan ekstraksi fitur (Priyanto and Ma'arif, 2018). Dalam penelitian ini digunakan analisis sentimen.

Analisis sentimen adalah proses ekstraksi dan pengolahan otomatis data teks untuk memahami informasi sentimen dalam kalimat opini. Tujuannya adalah untuk memahami kecenderungan opini terhadap suatu objek atau masalah, apakah beropini positif, negatif, atau netral. Pengaruh dan manfaat analisis sentimen sangat besar, dan penelitian serta pengembangan aplikasi terkait analisis sentimen saat ini berkembang pesat (Liu, 2010).

Metode analisis sentimen yang digunakan dalam penelitian ini adalah sebagai berikut:

1) Metode Support Vector Machine (SVM)

Support Vector Machine (SVM) adalah metode supervised learning untuk klasifikasi. SVM menggunakan garis pemisah atau hyperplane untuk proses klasifikasi (Wijaya and Karyawati, 2020). Dalam penelitian ini, digunakan jenis SVM non-linear dengan kernel Radial Basis Function (RBF). Persamaan (6) dan (7) menunjukkan metode SVM

dengan RBF sedangkan persamaan (8) menunjukkan fungsi kernel RBF (Pedregosa et al., 2011).

a. Rumus SVM RBF

$$\text{Min}_{w,b,\xi} \frac{1}{2} w^T w + C \sum_{i=1}^n \xi_i \quad (6)$$

dengan syarat:

$$y_i(w^T \phi(x_i) + b) \geq 1 - \xi_i, \xi_i \geq 0 \quad (7)$$

di mana:

w : vektor bobot
 b : bias
 ϕ : fungsi kernel RBF
 C : parameter regularisasi
 ξ_i : variabel slack
 n : jumlah sample data

b. Rumus kernel RBF

$$\phi(x_i, x_j) = \exp(-\gamma \|x_i - x_j\|^2) \quad (8)$$

di mana:

γ : parameter gamma yang mengontrol lebar kernel
 $x_i - x_j$: dua vektor data
 $\|x_i - x_j\|^2$: kuadrat jarak antara x_i dan x_j

2) Metode Logistic Regression

Logistic regression (LR) merupakan metode analisis data dalam statistika, yang dibuat untuk memahami korelasi antar variabel penjelas dan respons. Sifat variabel penjelas adalah kategoris dan kontinu, sedangkan variabel respons, baik ordinal atau nominal bersifat kategorikal. Penelitian ini menggunakan variabel respons yang bersifat nominal, yaitu positif, negatif, dan netral (Novantika and Sugiman, 2022). Dalam penelitian ini, pendekatan LR yang digunakan adalah multinomial dengan L2 regularization sehingga LR dapat dihitung dengan persamaan (9) dan (10) (Pedregosa et al., 2011):

a. Menghitung probabilitas kelas

$$\hat{y}(w, x) = \frac{\exp(w_0 + w_1 x_1 + \dots + w_p x_p)}{\sum_{k=1}^K \exp(w_{k0} + w_{k1} x_1 + \dots + w_{kp} x_p)} \quad (9)$$

di mana:

\hat{y} : probabilitas kelas k (kelas target)
 w : vektor bobot yang digunakan dalam model
 x : vektor input yang berisi nilai-nilai fitur yang digunakan dalam prediksi
 p : jumlah fitur dalam vektor input x
 K : jumlah kelas dalam masalah klasifikasi

b. Menghitung fungsi objektif pada model logistic

$$\min_w -\frac{1}{n} \sum_{i=1}^n \sum_{k=1}^K y_{ik} \log(\hat{y}_{ik}) + \frac{1}{2C} \|w\|^2 \quad (10)$$

di mana:

- \min_w : meminimasi terhadap variabel w
- n : jumlah sampel dalam data training
- y_{ik} : kelas target untuk sampel ke- i dalam kelas ke- k
- \hat{y}_{ik} : prediksi probabilitas kelas target sampel ke- i dalam kelas ke- k
- C : parameter pengontrol kekuatan regularisasi
- $\|w\|^2$: norma $L2$ dari vektor bobot w

3) Metode *Random Forest*

Random forest adalah metode pengembangan dari *Classification and Regression Trees (CART)*. Pendekatan yang diterapkan metode *random forest* adalah *bagging (bootstrap aggregating)* dan *random feature selection*. Konsep dari RF adalah membuat pohon-pohon keputusan, kemudian melakukan prediksi dengan kombinasi hasil dari setiap pohon keputusan. Dari kombinasi hasil tersebut, dipilih salah satu hasil sebagai keputusan akhir dengan cara *majority vote* (Adrian *et al.*, 2021). Penelitian ini menggunakan metode RF dengan pohon keputusan impuritas Gini, sehingga *impurity node* dapat dihitung dengan persamaan (11) (Pedregosa *et al.*, 2011):

$$C(t) = \frac{n_{left}}{n_t} G(t_{left}) + \frac{n_{right}}{n_t} G(t_{right}) \quad (11)$$

di mana:

- $C(t)$ = nilai *cost* pada *split t*
- n_{left} : jumlah data yang masuk ke cabang kiri dari *split t*
- n_{right} : jumlah data yang masuk ke cabang kanan dari *split t*
- n_t : jumlah total data pada *split t*
- $G(t_{left})$: *Gini index* pada cabang kiri dari *split t*
- $G(t_{right})$: *Gini index* pada cabang kanan dari *split t*

4) Metode *Naïve Bayes Classifier (NBC)*

Metode NBC merupakan salah satu metode klasifikasi dalam *data mining* yang menggunakan perhitungan probabilitas dalam proses klasifikasi. Metode *naïve bayes* klasifikasi memiliki ciri, yaitu asumsi yang kuat terhadap independensi dari setiap kejadian atau kondisi. Metode NBC memiliki dua tahap klasifikasi, yaitu tahap pertama *training* terhadap komentar yang sudah diketahui kelas atau kategorinya, tahap kedua adalah *testing* atau klasifikasi pada komentar yang belum diketahui kelasnya (Djamaludin *et al.*, 2022). Beberapa persamaan pada tahap metode NBC adalah sebagai berikut.

a. Menghitung probabilitas *prior* $p(c_i)$

$$p(c_i) = \frac{f_d(c_i)}{|D|} \quad (12)$$

di mana:

- $p(c_i)$: probabilitas prior kelas c_i
 - $f_d(c_i)$: jumlah komentar/data yang termasuk kelas c_i
 - $|D|$: jumlah data *training*
- b. Menghitung probabilitas $p(w_k|c_i)$

$$p(w_k|c_i) = \frac{f(w_{k_i}, c_i) + 1}{f(c_i) + |W|} \quad (13)$$

di mana:

- $p(w_k|c_i)$: probabilitas kata w ke- k pada kelas c_i
 - $f(w_{k_i}, c_i)$: frekuensi kemunculan kata w_{k_i} pada kelas c_i
 - $f(c_i)$: jumlah keseluruhan kemunculan kata pada kelas c_i
 - $|W|$: jumlah keseluruhan dari kata w ke- k
- c. Menghitung V_{map}

$$V_{map} = \underset{\{pos, neg, neu\}}{argmax} [\ln(p(c)) + \sum_{1 \leq k \leq nk} \ln(p(w_k|c))] \quad (14)$$

di mana:

- V_{map} : *variant of maximum a posteriori*
- $\ln(p(c))$: nilai logaritma natural dari $p(c)$
- $\sum_{1 \leq k \leq nk} \ln(p(w_k|c))$: nilai logaritma natural dari $p(w_k|c)$
- nk : jumlah kata pada komentar k

2.4. Evaluasi

Evaluasi merupakan tahap untuk mengukur kinerja model berdasarkan hasil klasifikasi. Metode evaluasi yang digunakan adalah *confusion matrix*. *Confusion matrix* merupakan sebuah metode evaluasi yang biasanya digunakan untuk mengukur kinerja suatu model klasifikasi. Dalam tahap evaluasi, *confusion matrix* digambarkan dalam Tabel 1. yang terdiri dari jumlah data uji yang diprediksi atau diklasifikasi dengan benar dan salah oleh model klasifikasi (Wijayanto, 2015).

Tabel 1. *Confusion matrix*

		Kelas yang diprediksi		
		Kelas	Negatif	Netral
Kelas Aktual	Negatif	TP	FP	FP
	Netral	FN	TN	FN
	Positif	FN	FN	TN

Dari *confusion matrix* pada Tabel 1. dapat dilakukan perhitungan nilai Akurasi, *precision*, *Recall*, dan *F1-score*, dengan rumus (15) sampai (18).

1) Akurasi

$$Akurasi = \frac{TP+TN}{TP+FP+TN+FN} \quad (15)$$

2) Presisi

$$Precision = \frac{TP}{TP+FP} \quad (16)$$

di mana:

TP : *True Positive* (Kelas aktual positif yang diprediksi positif)

FP : *False Positive* (kelas aktual bukan positif yang diprediksi positif)

3) Recall

$$Recall = \frac{TP}{TP+FN} \quad (17)$$

di mana:

FN : *False Negative* (Kelas positif yang diprediksi bukan positif)

4) F1-Score

$$F_1 - Score = 2 \times \frac{precision \times recall}{precision + recall} \quad (18)$$

3. Metode

3.1. Alur Penelitian

Penelitian ini menggunakan alur yang terdiri dari beberapa tahapan, dan untuk membantu pemahaman terhadap tahapan-tahapan tersebut, penulis menyajikan diagram seperti yang terlihat pada Gambar 1.



Gambar 1. Alur penelitian

3.2. Jenis dan Sumber data

Jenis data yang digunakan dalam penelitian ini adalah jenis data sekunder, yaitu berupa komentar pengguna aplikasi InDriver yang dikumpulkan dengan teknik *web scraping* pada situs *Google Play Store*. Pemilihan situs *Google Play Store* sebagai sumber data karena komentar pengguna merupakan salah satu fitur ulasan pada situs *Google Play Store*, yang mana fitur ulasan sering digunakan sebagai sarana untuk mencari informasi mengenai suatu produk atau jasa sebelum menggunakannya. Kebanyakan pengguna bergantung pada *review* pengguna lain karena dianggap mampu memberikan informasi terbaru dari suatu produk atau jasa berdasarkan perspektif pengguna yang telah menggunakan produk atau jasa tersebut (Rezki *et al.*, 2020).

Total data yang diperoleh dari situs *Google Play Store* adalah sebanyak 12147 komentar. Data komentar tersebut diambil pada 04 Oktober 2022 sampai 24 Januari 2023.

4. Hasil dan Pembahasan

4.1. Hasil

4.1.1. Pengambilan Data

Tahap pertama adalah pengambilan data dari situs *Google Play Store* dengan melakukan teknik *web scraping* menggunakan program *Python* yang diakses melalui <http://bit.ly/3xYUYu8>.

4.1.2. Pelabelan

Proses pelabelan dilakukan pada setiap data komentar yang akan dijadikan sebagai *dataset*. Untuk menghindari subjektivitas dalam proses analisis data, pelabelan dilakukan oleh dosen ahli bahasa Indonesia di Universitas Nusa Cendana. Hasil pelabelan *dataset* dapat diakses pada <https://bit.ly/440zMS6>.

4.1.3. Text Preprocessing

Proses pelabelan hanya melibatkan pemberian label pada setiap data komentar, sehingga masih mungkin terdapat gangguan (*noise*) yang dapat mempengaruhi hasil analisis teks. Oleh karena itu, dilakukan *text preprocessing* guna membersihkan data sebelum dilakukan analisis. Hasil *text preprocessing* dapat dilihat pada <https://bit.ly/3KuK7i6>.

4.2. Pembahasan

4.2.1. Text Preprocessing

1) Cleaning

Proses *cleaning* dalam penelitian ini digunakan untuk membersihkan teks komentar dari tanda baca, emotikon, karakter khusus, karakter angka, *hashtag* (#), *mention* (@), URL dan *link*. Tabel 2. menunjukkan tanda baca titik (.) pada komentar pertama dihapus pada tahap *cleaning*.

Tabel 2. Hasil *cleaning*

Komentar	Cleaning
bagus dan sangat membantu	bagus dan sangat membantu
tidak membantu	tidak membantu

2) Case Folding

Case folding dalam penelitian ini adalah mengubah bentuk teks menjadi *lower case*. Pada Tabel 3. dapat dilihat bahwa setelah proses *case folding* semua komentar sudah diubah menjadi *lower case*.

Tabel 3. Hasil *case folding*

Cleaning	Case Folding
bagus dan sangat membantu	bagus dan sangat membantu
tidak membantu	tidak membantu

3) Normalisasi

Proses ini melakukan pencarian kata dalam kamus *slang words*, jika ditemukan maka akan

diubah menjadi kata baku. Daftar kata yang dijadikan sebagai kamus *slang words*, dapat diakses pada <https://bit.ly/4111w90>. Pada tahap ini kata ‘membntu’ pada komentar pertama diubah menjadi kata ‘membantu’ seperti pada tabel 4.

Tabel 4. Hasil normalisasi

Case Folding	Normalisasi
bagus dan sangat membantu	bagus dan sangat membantu
tidak membantu	tidak membantu

4) Tokenizing

Proses *tokenizing* memecah teks berdasarkan spasi menjadi kumpulan kata atau token. Hasil proses *tokenizing* dapat dilihat pada tabel 5.

Tabel 5. Hasil *tokenizing*

Normalisasi	Tokenizing
bagus dan sangat membantu	['bagus', 'dan', 'sangat', 'membantu']
tidak membantu	['tidak', 'membantu']

5) Stemming

Metode *stemming* yang digunakan pada penelitian ini adalah metode Nazief-Adriani, yang dikembangkan oleh Bobby Nazief dan Mirna Adriani (Adriani *et al.*, 2007). Hasil *stemming* ditunjukkan pada tabel 6 di mana kata ‘membantu’ diubah menjadi kata dasarnya ‘bantu’.

Tabel 6. Hasil *stemming*

Tokenizing	Stemming
['bagus', 'dan', 'sangat', 'membantu']	bagus dan sangat bantu
['tidak', 'membantu']	tidak bantu

6) Convert Negation

DM (*_neg*) ditambahkan pada kata yang mengandung negasi. Pada komentar kedua, ‘tidak bantu’ diubah menjadi ‘tidak bantu_neg’ seperti pada tabel 7.

Tabel 7. Hasil *convert negation*

Stemming	Convert Negation
bagus dan sangat bantu	bagus dan sangat bantu
tidak bantu	tidak bantu_neg

7) Stopword Removal

Stopword removal pada penelitian ini, dilakukan dengan menggunakan *library Python Sastrawi*. Daftar *stopword* dapat dilihat pada <http://bit.ly/3S2wcSQ>. Pada tahap ini, kata ‘dan’ pada komentar pertama akan dihapus karena termasuk ke dalam *stopword* seperti ditunjukkan pada tabel 8.

Tabel 8. Hasil *stopword removal*

Convert Negation	Stopword Removal
bagus dan sangat bantu	bagus sangat bantu
tidak bantu_neg	tidak bantu_neg

4.2.2. Ekstraksi Fitur Teks

Metode ekstraksi fitur digunakan untuk menghitung nilai skor dari tiap kata (*term*) dalam dokumen (komentar). Dalam hal ini, setiap kata dalam dokumen dihitung skornya melalui proses ekstraksi fitur, di mana nilai dari setiap fitur dinormalisasi agar berada dalam rentang skala antara 0 dan 1. Metode ekstraksi fitur yang digunakan adalah metode TF-IDF.

4.3. Evaluasi dan Perbandingan

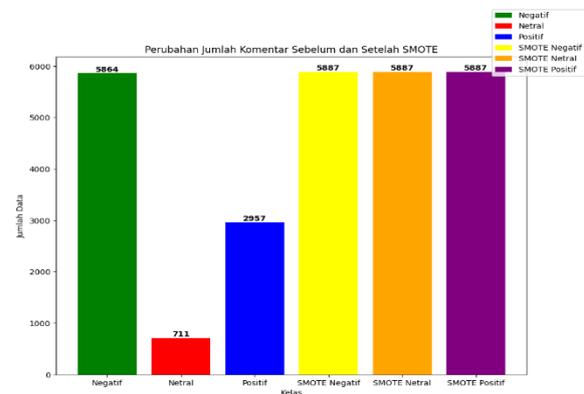
Total *dataset* awal yang dimasukkan ke dalam sistem adalah sebanyak 12147 komentar. Setelah dilakukan proses *preprocessing*, sebanyak 12 komentar dihapus karena memiliki nilai *null* atau kosong sehingga total *dataset* yang tersisa adalah sebanyak 12135 komentar. *Dataset* hasil *preprocessing* kemudian dibagi menjadi 80% menjadi komentar pelatihan model dan 20% menjadi komentar pengujian model untuk setiap kelas. Hasil pembagian *dataset* dapat dilihat pada Tabel 9.

Tabel 9. Hasil pembagian komentar

Data	Negatif	Netral	Positif
Data <i>training</i>	5828	886	2994
Data <i>testing</i>	1486	228	713
Total	7314	1114	3707

Pembagian *dataset* pada Tabel 2, digunakan untuk pengujian dan evaluasi semua metode *machine learning* yang digunakan dalam penelitian ini. Semua metode diimplementasikan dengan menggunakan *library scikit-learn python*.

Proses pengujian dan evaluasi dilakukan dengan dua skema yaitu, tanpa *Synthetic Minority Over-sampling Technique (SMOTE)* atau penanganan terhadap ketidakseimbangan kelas (*imbalanced class*) dan dengan metode SMOTE.



Gambar 2. Grafik perubahan jumlah data training

1) Skema tanpa SMOTE

Dari hasil pengujian, diperoleh hasil evaluasi untuk setiap metode seperti pada Tabel 10.

Tabel 10. Hasil evaluasi tanpa SMOTE

Metrik	Kelas	SVM	LR	NBC	RF
Precision	Negatif	88%	88%	86%	89%
	Netral	80%	74%	93%	83%
	Positif	93%	92%	92%	89%
Recall	Negatif	97%	97%	98%	96%
	Netral	39%	39%	23%	42%
	Positif	89%	88%	86%	87%
F1-score	Negatif	92%	92%	91%	92%
	Netral	53%	51%	37%	55%
	Positif	91%	90%	89%	88%
Akurasi		89%	89%	88%	88%

Jika dilihat dari akurasi setiap metode pada Tabel 10, metode SVM dan LR memperoleh akurasi tertinggi yaitu sebesar 89%, yang berarti secara keseluruhan model yang dibuat dengan kedua metode tersebut dapat mengklasifikasi sebanyak 89% dari total data *testing* yang diberikan. Namun, metrik tersebut tidak dapat dijadikan sebagai satu-satunya tolok ukur untuk menilai kualitas metode, khususnya ketika mempertimbangkan keseimbangan antara *precision* dan *recall* pada kelas minoritas (netral), metode RF lebih seimbang dibandingkan dengan metode lainnya. Hal tersebut dapat dilihat dari nilai *F1-score* pada setiap metode, yang mana *F1-score* pada *Random Forest* lebih tinggi, yaitu sebesar 55%.

2) Skema dengan SMOTE

Dari hasil pengujian setelah implementasi metode SMOTE, diperoleh hasil evaluasi seperti pada Tabel 11.

Tabel 11. Hasil evaluasi dengan SMOTE

Metrik	Kelas	SVM	LR	NBC	RF
Precision	Negatif	88%	88%	91%	88%
	Netral	80%	74%	48%	82%
	Positif	93%	92%	93%	89%
Recall	Negatif	97%	97%	92%	96%
	Netral	39%	39%	57%	41%
	Positif	89%	88%	86%	87%
F1-score	Negatif	92%	92%	92%	92%
	Netral	53%	51%	52%	55%
	Positif	91%	90%	89%	88%
Akurasi		89%	89%	87%	88%

Hasil evaluasi pada Tabel 11. menunjukkan bahwa metode SVM, LR, dan RF tetap mempertahankan performa baik setelah penyeimbangan data *training*. Akurasi yang tetap pada kisaran 89% serta nilai *F1-score* yang seimbang antara *precision* dan *recall* pada berbagai kelas, menunjukkan hasil yang konsisten. Metode SVM menghasilkan hasil yang konsisten dalam hal *precision*, *recall*, dan *F1-score* untuk semua kelas, menegaskan kemampuan model dalam mengklasifikasikan dengan baik. Prediksi dengan *precision* tinggi pada semua kelas (88% untuk negatif, 80% untuk netral, 93% untuk positif) mengindikasikan relevansi dengan kelas. *Recall* yang

tinggi pada kelas negatif (97%) dan positif (89%) menunjukkan kemampuan mengenali sebagian besar sampel kelas tersebut. Nilai *F1-score* yang tinggi untuk semua kelas (92% untuk negatif, 53% untuk netral, 91% untuk positif) menunjukkan keseimbangan yang sejalan dengan hasil evaluasi sebelumnya.

Metode LR juga mempertahankan performa baik dengan *precision* stabil pada kelas negatif (88%), netral (74%), dan positif (92%). *Recall* yang tinggi pada kelas negatif (97%) dan positif (88%) menegaskan kemampuan mengenali kelas-kelas ini. Namun, potensi peningkatan pada *recall* kelas netral (39%) masih dapat ditingkatkan. Nilai *F1-score* yang baik (92% untuk negatif, 51% untuk netral, 90% untuk positif) menunjukkan keseimbangan yang baik antara *precision* dan *recall*. Metode LR tetap andal dengan akurasi 89%.

Sementara itu, metode NBC menunjukkan variasi dalam hasil *precision*, *recall*, dan *F1-score* antara kelas-kelas. Meskipun *precision* tinggi pada kelas negatif (91%) dan positif (93%), *recall* dan *F1-score* memiliki variasi. *Recall* yang tinggi pada kelas negatif (92%) dan netral (57%) menunjukkan kemampuan model mengenali kelas-kelas ini. Namun, ada potensi peningkatan pada *recall* kelas positif (86%) dan nilai *F1-score* pada kelas netral (52%). Meskipun begitu, performa metode ini tetap baik dengan akurasi 87%. Terjadi peningkatan signifikan pada metode NBC setelah penyeimbangan data, dengan *F1-score* naik dari 37% menjadi 52%.

Metode RF tetap menunjukkan performa baik dengan *precision* stabil pada kelas negatif (88%), netral (82%), dan positif (89%). *Recall* yang tinggi pada kelas negatif (96%) tetap kuat. Meskipun demikian, ada potensi peningkatan pada *recall* kelas netral (41%). *F1-score* yang tinggi pada semua kelas (92% untuk negatif, 55% untuk netral, 88% untuk positif) menunjukkan keseimbangan yang diperoleh antara *precision* dan *recall*. Metode RF tetap dapat diandalkan dengan akurasi 88%.

5. Kesimpulan

Berdasarkan hasil analisis, dapat disimpulkan bahwa dalam eksperimen ini metode SMOTE memberikan pengaruh yang cukup signifikan terhadap metode NBC, sementara metode SVM dan LR menghasilkan akurasi tertinggi, yaitu sebesar 89%. Hasil ini menunjukkan bahwa kedua metode tersebut mampu mengklasifikasikan sekitar 89% dari total data uji dengan benar.

Namun, perlu diperhatikan bahwa akurasi saja tidak cukup untuk mengevaluasi kualitas metode klasifikasi. Dalam konteks ini, keseimbangan antara *precision* dan *recall* pada kelas minoritas (netral) juga menjadi faktor penting dalam mengevaluasi metode. Dalam hal ini, metode LR menunjukkan keseimbangan yang lebih baik dibandingkan metode

lainnya, terlihat dari nilai *F1-score* yang lebih tinggi, yaitu sebesar 55%.

Oleh karena itu, meskipun SVM dan LR memiliki akurasi tinggi, pertimbangan terhadap keseimbangan *precision* dan *recall* pada kelas minoritas harus diperhitungkan. Dalam hal ini, metode RF menunjukkan performa yang lebih seimbang.

Daftar Pustaka

- Adrian, M.R., Putra, M.P., Rafialdy, M.H., Rakhmawati, N.A., 2021. Perbandingan Metode Klasifikasi Random Forest dan SVM pada Analisis Sentimen PSBB. *Jurnal Informatika Upris*, 7(1), 36-40. <https://doi.org/10.26877/jiu.v7i1.7099>
- Adriani, M., Asian, J., Nazief, B., Tahaghoghi, S.M.M., Williams, H.E., 2007. Stemming Indonesian: A confix-stripping approach. *ACM Transactions on Asian Language Information Processing*, 6(4), 1-33. <https://doi.org/10.1145/1316457.1316459>
- Anbari, M.Z., Sugiantoro, B., 2023. Studi Komparasi Metode Analisis Sentimen Naïve Bayes, SVM, dan Logistic Regression pada Piala Dunia 2022. *Jurnal Media Informatika Budidarma*, 7(2), 688-695. <http://dx.doi.org/10.30865/mib.v7i2.5383>
- Anjasmoros, M.T., Istiadi, Marisa, F., 2020. Analisis Sentimen Aplikasi Go-jek Menggunakan Metode SVM dan NBC (Studi kasus: Komentar Pada Play Store). *Conference on Innovation and Application of Science and Technology (CIASTECH)*, 489-498. <https://doi.org/10.31328/ciastech.v3i1.1905>
- Djamaludin, M. A., Triayudi, A., Mardiani, E., 2022. Analisis Sentimen Tweet KRI Nanggala 402 di Twitter menggunakan Metode Naïve Bayes Classifier. *Jurnal JTik (Jurnal Teknologi Informasi dan Komunikasi)*, 6(2), 161-166. <https://doi.org/10.35870/jtik.v6i2.398>
- Fitri, E., Yuliani, Y., Rosyida, S., Gata, W., 2020. Analisis Sentimen Terhadap Aplikasi Ruangguru Menggunakan Algoritma Naive Bayes, Random Forest dan Support Vector Machine. *Transformatika*, 18(1), 71-80. <http://dx.doi.org/10.26623/transformatika.v18i1.2317>
- Herwijayanti, B., Ratnawati, D.E., Muflikhah, L., 2018. Klasifikasi Berita Online dengan menggunakan Pembobotan TF-IDF dan Cosine Similarity. *Jurnal Pengembangan Teknologi Informasi dan Ilmu Komputer*, 2(1), 306-312.
- Karja, 2019. Mengenal inDriver Pesaing Baru Gojek & Grab. Diakses pada 18 Juli 2023, pada <https://kumparan.com/karjaid/mengenal-indriver-pesaing-baru-gojek-and-grab-1sCtgTXz3Nn>
- Liu, B., 2010. Sentiment analysis: A multi-faceted problem. *IEEE intelligent systems*, 25, 76-80.
- Muttaqin, M.N., Kharisudin, I., 2021. Analisis Sentimen Aplikasi Gojek Menggunakan Support Vector Machine dan K Nearest Neighbor. *UNNES Journal of Mathematics*, 10(2), 22-27. <https://doi.org/10.15294/ujm.v10i2.48474>
- Novantika, A., Sugiman, S., 2022. Analisis Sentimen Ulasan Pengguna Aplikasi Video Conference Google Meet menggunakan Metode SVM dan Logistic Regression. *PRISMA, Prosiding Seminar Nasional Matematika*, 5, 808-813.
- Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M., Duchesnay, E., 2011. Scikit-learn: Machine Learning in Python. *Journal of Machine Learning Research*, 12(85), 2825-2830.
- Priyanto, A., Ma'arif, M.R., 2018. Implementasi Web Scrapping dan Text Mining untuk Akuisisi dan Kategorisasi Informasi dari Internet (Studi Kasus: Tutorial Hidroponik). *Indonesian Journal of Information Systems*, 1(1), 25-33. <https://doi.org/10.24002/ijis.v1i1.1664>
- Putri, M.I., Kharisudin, I., 2022. Analisis Sentimen Pengguna Aplikasi Marketplace Tokopedia pada Situs Google Play Menggunakan Metode Support Vector Machine (SVM), Naïve Bayes, dan Logistic Regression. *PRISMA, Prosiding Seminar Nasional Matematika*, 5, 759-766.
- Rezki, M., Kholifah, D.N., Faisal, M., Priyono, Suryadithia, R., 2020. Analisis Review Pengguna Google Meet dan Zoom Cloud Meeting Menggunakan Algoritma Naïve Bayes. *Jurnal Infortech*, 2(2), 264-270. <https://doi.org/10.31294/infortech.v2i2.9286>
- Rokhman, K.A., Berlilana, Arsi, P., 2021. Perbandingan Metode Support Vector Machine dan Decision Tree untuk Analisis Sentimen Review Komentar pada Aplikasi Transportasi Online. *Journal of Information System Management (JOISM)*, 2(2), 1-7. <https://doi.org/10.24076/joism.2021v3i1.341>
- Seliya, N., Khoshgoftaar, T.M., Van Hulse, J., 2010. Predicting Faults in High Assurance Software. *2010 IEEE 12th International Symposium on High Assurance Systems Engineering*, 26-34. <https://doi.org/10.1109/HASE.2010.29>
- Sianipar, G.J., 2019. Pengaruh Kualitas Pelayanan, Persepsi Harga dan Citra Merek Terhadap kepuasan Pelanggan Pengguna Jasa Transportasi Ojek Online (Studi pada Pelanggan GrabBike di Kota Medan). *Jurnal Manajemen dan Bisnis*, 19(2), 183-196.
- Sitompul, K.P.J., Pratama, A.R., Baihaqi, K.A., 2023. Komparasi Algoritma Naive Bayes, Support Vector Machine, dan Logistic Regression pada Analisis Sentimen Pengguna Aplikasi Transportasi Online. *Klik-Kumpulan Jurnal Ilmu Komputer*, 10(1), 27-38. <http://dx.doi.org/10.20527/klik.v10i1.616>

- Sitorus, P.R., 2021. *Analisis Sentimen Data Ulasan Aplikasi Indriver pada Situs Google Play Menggunakan Metode Naïve Bayes Classifier dan Support Vector Machine*. PhD Thesis, Universitas Sumatera Utara.
<https://repositori.usu.ac.id/handle/123456789/46908>
- Tuwanakotta, J.L., Tanaamah, A.R., 2022. Evaluation of Usability Quality Between InDriver and Maxim Applications Using Usability Scale (SUS) and Usability Testing Methods. *Sistemasi: Jurnal Sistem Informasi*, 11(3), 630-645.
<https://doi.org/10.32520/stmsi.v11i3.2001>
- Wijaya, K.D.Y., Karyawati, A.A.I.N.E., 2020. The Effects of Different Kernels in SVM Sentiment Analysis on Mass Social Distancing. *Jurnal Elektronik Ilmu Komputer Udayana*, 9(2), 161-168.
<https://doi.org/10.24843/JLK.2020.v09.i02.p01>
- Wijayanto, H., 2015. Klasifikasi Batik Menggunakan Metode K-Nearest Neighbour Berdasarkan Gray Level Co-Occurrence Matrices (GLCM). Seminar Nasional Aplikasi Teknologi Informasi. Presented at the Seminar Nasional Aplikasi Teknologi Informasi, Yogyakarta.