
COMPARISON OF ACCURACY LEVELS OF SVM, DECISION TREE AND RANDOM FOREST ALGORITHMS IN SENTIMENT ANALYSIS OF USER RESPONSES OF THE GOPAY APPLICATION

Indriani¹, Ade Davy Wiranata²

^{1,2}Informatics Engineering, Faculty of industrial technology and informatics, Universitas Muhammadiyah Prof. Dr. Hamka, Indonesia
Email: ¹indriiyy07@gmail.com, ²adedavy@uhamka.ac.id

(Article received: February 21, 2024; Revision: March 9, 2024; published: May 28, 2024)

Abstract

The development of technology from time to time makes all work or activities easier, one of which is online money transactions which are called e-wallets or digital wallets. One of the digital wallet applications that is often used is GoPay, which is a platform and tool created for making digital payments. Not long ago, GoPay was separated into one application, which previously existed in the Gojek application. However, every application certainly has a negative side, such as GoPay, where to use the application you have to be connected to the internet, which creates dependence on smartphones. Based on this problem, the company needs to know the response of users of the GoPay application which has been launched using the SVM, Decision Tree and Random algorithms. Forest. Therefore, the aim of this research is to carry out sentiment analysis on the responses of GoPay application users after being separated from Gojek and to find out the comparison of evaluation results or accuracy produced by the three algorithms. The results of this research show that of the three algorithms used, Positive sentiment is more than Negative sentiment, where in SVM Positive 89% and Negative 85%, Decision Tree class Positive 89% and Negative 76% while in Random Forest class positive 93% and Negative 86%. Apart from that, the Random Forest algorithm has a high level of accuracy, namely 90%, then the SVM algorithm 88% and the Decision Tree algorithm 84%.

Keywords: *Decision Tree, Gopay, Random Forest, Sentiment Analysis, SVM.*

PERBANDINGAN TINGKAT AKURASI ALGORITMA SVM, DECISION TREE, DAN RANDOM FOREST DALAM ANALISIS SENTIMEN TANGGAPAN PENGGUNA APLIKASI GOPAY

Abstrak

Berkembangnya teknologi dari masa ke masa membuat segala pekerjaan atau kegiatan menjadi lebih mudah, salah satunya pada transaksi uang secara online yang di sebut dengan e-wallet atau dompet digital. Salah satu aplikasi dompet digital yang sering di gunakan adalah GoPay yang merupakan platform dan alat yang di buat untuk melakukan pembayaran secara digital. Belum lama ini GoPay di buat terpisah menjadi satu aplikasi yang sebelumnya ada pada aplikasi gojek. Namun, setiap aplikasi tentunya memiliki sisi negatif seperti pada GoPay dimana untuk menggunakan aplikasi tersebut harus terkoneksi dengan internet yang membuat ketergantungan pada smartphone. Berdasarkan masalah tersebut, Perusahaan perlu mengetahui respon pengguna aplikasi GoPay yang telah di luncurkan dengan menggunakan algoritma *SVM*, *Decision Tree*, dan *Random Forest*. Oleh karena itu, tujuan dari penelitian ini adalah untuk melakukan analisis sentimen pada tanggapan pengguna aplikasi GoPay setelah terpisah dari gojek serta mengetahui komparasi hasil evaluasi atau akurasi yang di hasilkan oleh ketiga algoritma tersebut. Hasil penelitian ini menunjukkan bahwa dari ketiga algoritma yang di gunakan sentimen Positif lebih banyak dari sentimen Negatif dimana pada *SVM* Positif 89% dan Negatif 85%, *Decision Tree* class Positif 89% dan Negatif 76% sedangkan pada *Random Forest* class positif 93% dan Negatif 86%. Selain itu tingkat akurasi yang tinggi ada pada algoritma *Random Forest* yaitu 90%, kemudian algoritma *SVM* 88% dan algoritma *Decision Tree* 84%.

Kata kunci: *Analisis Sentimen, Decision Tree, Gopay, Random Forest, SVM.*

1. PENDAHULUAN

Semakin berkembangnya teknologi dari masa ke masa membuat segala pekerjaan atau kegiatan menjadi lebih mudah, salah satunya pada transaksi uang secara online yang di sebut dengan e-wallet atau dompet digital. Dompet digital adalah teknologi yang memberikan fasilitas pada para penggunanya dimana dapat menyimpan, mengirim, serta menerima uang secara elektronik. Dompet digital biasanya terhubung dengan kartu kredit, kartu debit, atau rekening bank pengguna [1]. Adanya perkembangan teknologi ini sangat memudahkan untuk melakukan berbagai transaksi uang tanpa harus membawa *cash* serta menggunakan kartu debit atau mengunjungi bank yang menyediakan mesin ATM.

Salah satu aplikasi dompet digital yang sering di gunakan adalah GoPay yang merupakan platform dan alat yang di buat untuk melakukan pembayaran secara digital [2]. GoPay awalnya merupakan bagian dari gojek yang berdiri pada tahun 2010 dan didirikan oleh Nadiem Makarim. Gojek salah satu perusahaan transportasi online yang cukup terkenal di Indonesia. Setelah berhasilnya bisnis transportasi online, gojek memperluas bisnisnya ke industry fintech dengan mengembangkan layanan GoPay. Pada tahun 2016, aplikasi Gojek secara resmi meluncurkan layanan GoPay sebagai dompet digital [3]. Namun, saat ini aplikasi GoPay dapat di akses secara terpisah yang di luncurkan pada tanggal 26 Juli 2023 [4] artinya untuk menggunakan Gopay maka tidak harus mempunyai aplikasi gojek. Seperti penelitian yang telah di lakukan oleh [5] menggunakan algoritma Naïve Bayes Classifier mendapatkan hasil bahwa NBC mampu menganalisis sentimen pengguna terhadap aplikasi GoPay dengan tingkat keakuratan yang cukup baik.

Dengan pertumbuhan jumlah pengguna E-Wallet yang semakin meningkat, tentunya terjadi peningkatan dalam jumlah komentar atau pendapat terhadap aplikasi tersebut salah satunya pada GoPay [6]. Hadirnya aplikasi GoPay tentu berdampak positif bagi semua orang dimana lebih memudahkan dalam transaksi pembayaran serta mudah diawasi. Namun, setiap aplikasi tentunya memiliki kelemahan atau sisi negatif seperti dimana beberapa pekerjaan akan menghilang serta untuk menggunakan aplikasi ini harus terkoneksi dengan internet yang mengakibatkan ketergantungan terhadap smartphone [7]. Tanggapan positif dan negatif selalu ada dari pengguna aplikasi GoPay yang di ekspresikan melalui ulasan pada google play store [8].

Berdasarkan ulasan tersebut, perusahaan perlu meningkatkan kualitas dari produk yang telah di luncurkannya. Untuk mengetahui respon serta tanggapan dari pengguna aplikasi GoPay digunakan analisis sentimen. Analisis sentimen adalah proses mengumpulkan, mengklasifikasi, atau menganalisis pendapat atau sentimen yang di berikan oleh pengguna yang bersifat positif, netral, maupun negatif

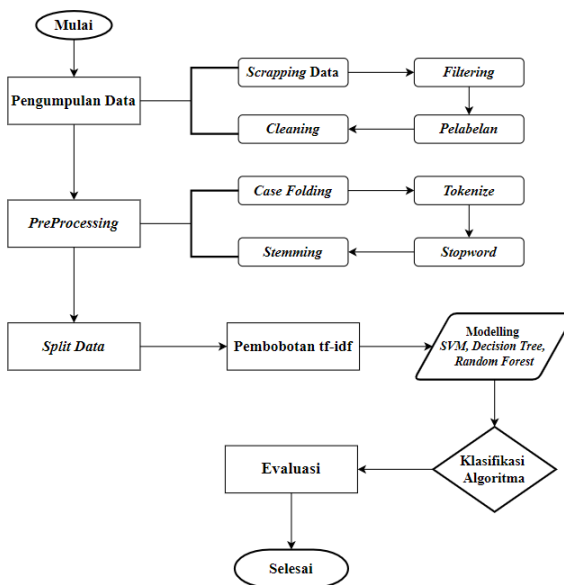
[9]. Metode klasifikasi yang di gunakan dalam penelitian analisis sentimen ini adalah *Support Vector Machine (svm)*, *Random Forest*, dan *Decision Tree*. Ketiga metode tersebut tentunya memiliki tingkat akurasi yang berbeda-beda berdasarkan klasifikasi datanya.

Tujuan dari penelitian ini adalah untuk melakukan analisis sentimen pada tanggapan pengguna aplikasi GoPay setelah terpisah dari gojek serta mengetahui perbandingan atau komparasi hasil evaluasi atau akurasi yang di dihasilkan oleh algoritma *Support Vector Machine (SVM)*, *Decision Tree*, dan *Random Forest* berdasarkan pengklasifikasian yang telah di lakukan.

2. METODE PENELITIAN

2.1. Tahap Penelitian

Tahap penelitian dapat di lihat pada Gambar 1.



Gambar 1. Tahap penelitian

Gambar 1 merupakan tahapan penelitian yang di lakukan dimulai dengan pengumpulan data sampai dengan tahap evaluasi.

2.2. Pengumpulan Data

Data yang di gunakan pada penelitian ini merupakan hasil pengumpulan ulasan dari pengguna aplikasi GoPay di Google Play Store sebanyak 1500 data yang kemudian akan di olah menggunakan Google *Collaboratory (colab)* python. Pengumpulan data di mulai dari proses sebagai berikut.

a. Scraping Data

Scraping data merupakan salah satu teknik yang dipergunakan untuk mengumpulkan informasi berupa numerik dan teks dari berbagai situs web secara otomatis dengan menggunakan sebuah perangkat lunak [10].

b. Filtering

Filtering merupakan proses mengambil data sesuai dengan kebutuhan dalam analisis sentimen yang di lakukan.

c. Labeling Data

Pelabelan adalah suatu proses untuk memberi label pada kalimat dimana label ini bisa yang berupa positif dan negatif berdasarkan teksnya.

d. Cleaning

Cleaning pada penelitian ini berfungsi untuk mengecek data atau label yang bernilai kosong yang kemudian akan di hilangkan atau pembersihan data yang tidak bernilai.

2.3. PreProcessing

PreProcessing adalah proses mengubah data menjadi format yang sesuai dengan kebutuhan. *PreProcessing* di gunakan untuk menghilangkan kata dan simbol-simbol yang tidak bermanfaat [11]. Tahap dari *PreProcessing* adalah sebagai berikut.

a. Case Folding

Case folding adalah proses mengganti huruf yang awalnya besar menjadi huruf kecil serta menghapus tanda baca pada suatu teks [12].

b. Stopword Removal

Stopword removal bertujuan untuk menghapus kata – kata yang tidak relevan atau bermanfaat dalam suatu kalimat berdasarkan daftar stopword [13].

c. Tokenizing

Proses *Tokenize* di lakukan agar teks kalimat dapat berubah menjadi bagian per kata sehingga proses analisis lebih mudah [14].

d. Stemming

Stemming merupakan suatu teknik untuk mengubah sebuah kalimat atau bentuk kata agar kembali pada kata dasarnya.

2.4. Splitting Data

Splitting data merupakan proses untuk membagi data latih (training) dan data uji (testing) yang mana data latih di gunakan untuk melatih model yang di pakai sedangkan data uji di gunakan untuk menguji akurasi dari model yang di pakai. Pembagian data biasanya menggunakan banding 80:20%, 70:30% atau 60:40% [15].

2.5. Pembobotan Tf-Idf

Proses pembobotan tf-idf adalah metode algoritmik yang di gunakan untuk menghitung bobot tiap kata yang sering digunakan. Dalam pembobotan tf-idf dapat menghitung Term Frekuensi (TF) untuk mengukur seberapa sering suatu kata muncul dalam suatu dokumen serta Inverse Document Frekuensi (IDF) untuk mengurangi bobot suatu kata jika kata tersebut muncul banyak di seluruh dokumen. Teknik ini sangat berguna dalam analisis teks dan pengelompokan dokumen karena bisa mengetahui kata yang sering muncul dalam suatu dokumen [16].

2.6. Modelling

Pemodelan adalah proses membangun model yang akan di gunakan bersarkan dataset yang telah bersih. Pada penelitian ini akan menggunakan metode klasifikasi dengan algoritma *Support Vector Machine (svm)*, *Decision Tree*, dan *Random Forest* [17].

2.7. Support Vector Machine (SVM)

Support Vector Machine (SVM) merupakan suatu metode yang di pakai untuk memprediksi sebuah kasus klasifikasi. Dalam *Support Vector Machine (SVM)*, cara kerja klasifikasi adalah dengan mengoptimalkan hyperlane, yaitu hyperplane yang membagi ruang data menjadi dua bagian dengan jarak maksimum di antara mereka. Dalam penggunaan algoritma *SVM* bisa menguji tiga kernel berbeda sebagai scenario percobaan, yaitu kernel linear, kernel polinomial, dan kernel radial basis function (RBF) [18].

2.8. Decision Tree

Decision Tree adalah sebuah metode terkenal dan efektif yang di gunakan untuk kasus prediksi maupun klasifikasi dimana bisa mengubah fakta menjadi struktur pohon keputusan yang merepresentasikan aturan-aturan yang mudah untuk di pahami. Mengubah data menjadi pohon keputusan, mengubah pohon keputusan menjadi role, dan menyederhanakan role adalah proses yang dilakukan oleh metode *Decision Tree*. Metode *Decision Tree* menggunakan pohon yang memiliki internal node, root node, dan terminal node. Internal node dan root node merepresentasikan variabel atau fitur, dan terminal node merepresentasikan label kelas. Data query akan bergerak dari internal node dan root node menuju terminal node. Label kelas untuk data query sesuai dengan label yang terdapat di internal node [19].

2.9. Random Forest

Random Forest merupakan suatu teknik ensemble learning yang memanfaatkan sejumlah besar pohon keputusan. Setiap pohon keputusan dibangun secara acak dengan menggunakan subkelompok dari data pelatihan serta subkelompok fitur yang dipilih secara acak. Selama proses pelatihan, tiap pohon keputusan akan mempelajari cara memprediksi sentimen berdasarkan ciri-ciri yang ada dalam data. Setelah model *Random Forest* terlatih, serangkaian data uji akan digunakan untuk memprediksi sentimen dari teks yang belum dilabeli. Setiap pohon keputusan dalam *Random Forest* akan memberikan prediksi sentimen, dan hasil akhirnya akan ditentukan oleh mayoritas suara dari seluruh pohon keputusan. Sentimen dengan suara terbanyak akan dianggap sebagai prediksi sentimen akhir. *Random Forest* memiliki beberapa hiperparameter

yang dapat disesuaikan untuk meningkatkan kinerja model [20].

2.10. Evaluasi

Evaluasi merupakan tahap akhir yang di gunakan untuk menghitung nilai akurasi, recall dan presisi yang di hasilkan sistem pada data testing yang tersedia. Luaran dari proses tersebut berupa class positif dan negatif sesuai dengan data uji. *Confusion Matrix* merupakan bentuk representasi dari semua nilai yang di hasilkan dalam proses evaluasi [21]. Confusion matrix memiliki bentuk seperti pada Tabel 1 [22].

Tabel 1. Confusion matrix

Confusion Matrix		Nilai sebenarnya	
		Terklarifikasi positif	Terklarifikasi negatif
Nilai Prediksi	Positif	True Positif (TP)	False Positif (FP)
	Negatif	False Negatif (FN)	True Negatif (TN)

Keterangan:

- True Positif (TP) adalah hasil positif yang nilai prediksinya di benarkan.
- False Negatif (FN) adalah menghasilkan positif namun pada data di anggap sebagai negatif.
- False Positif (FP) adalah menghasilkan negatif namun pada data di anggap positif.
- True Negatif (TN) adalah hasil negatif yang nilai prediksinya di benarkan.

Evaluasi model juga dapat di ukur dengan menggunakan persamaan sebagai berikut [23].

a. Akurasi

Akurasi merupakan indikator yang merepresentasikan seberapa efektif model klasifikasi yang di gunakan dalam memperkirakan kelas yang benar dari dataset yang dimiliki. Untuk mendapatkan nilai akurasi di gunakan rumus seperti pada persamaan (1).

$$Akurasi = \frac{TP+TN}{TP+TN+FP+FN} \times 100\% \tag{1}$$

b. Recall

Recall merupakan metrik yang menunjukkan seberapa banyak data yang berhasil diklasifikasikan secara tepat dalam suatu kategori, dibandingkan dengan total jumlah data yang sebenarnya masuk dalam kategori tersebut. Untuk mendapatkan nilai recall di gunakan rumus seperti pada persamaan (2).

$$Recall = \frac{TP}{TP+FN} \times 100\% \tag{2}$$

c. Presisi

Presisi mengacu pada tingkat keakuratan respons sistem dalam menyajikan informasi yang relevan sesuai dengan kebutuhan pengguna. Untuk mendapatkan nilai presisi di gunakan rumus seperti pada persamaan (3).

$$Precision = \frac{TP}{TP+FP} \times 100\% \tag{3}$$

3. HASIL DAN PEMBAHASAN

3.1. Hasil Pengumpulan Data

a. Scraping Data

Tahap ini akan di lakukan scrap pada ulasan yang paling relevan dari pengguna GoPay sebanyak 1500 data di Google Play Store. Proses scrap data dapat di lihat pada Gambar 2.

```
from google_play_scraper import Sort, reviews

result, continuation_token = reviews(
    'com.gojek.gopay',
    lang='id',
    country='id',
    sort=Sort.MOST_RELEVANT,
    count=1500,
    filter_score_with=None
)
```

Gambar 2. Proses scraping data

reviewId	userName	userImage	content	score	thumbsUpCount	reviewCreatedVersion	at
2073dbc8c4e7f9bcb20a0956	Deja Vu		Keran bangal epikasi ini, semua fungsi e bank...	1	4256	1.15.1	2024-02-11 04:51:10
2e4db7f2f4c09b2339a73a6f	LailiDRG		Buat admin...saya pengguna baru...pernyataan ...	1	146	1.15.1	2024-02-13 05:50:13
4901a59324802c816a9c196bfc0d	Abodaf gataumette		Semua fiturnya gampang buat digunakan, isarpria...	5	6935	1.14.0	2023-03-28 03:58:32
35a6odaic540cc82d37809a09	Syamsul Cahyadi		Saya kira aplikasinya sudah bagus, tapi fasill...	1	743	1.14.0	2023-12-31 12:55:19
44R49RL			Alhamdulillah				2024...

Gambar 3. Hasil scrap data

Gambar 3 menunjukan hasil scrap data melalui Google Colab yang menghasilkan UserName, Content, at, dan lain sebagainya.

b. Filtering

Dalam proses filtering, akan di lakukan filter terhadap data yang di peroleh dari scrapp agar sesuai dengan yang di butuhkan dimana hanya mengambil content dan score saja.

```
data = data.rename(columns={'content': 'Tanggapan', 'score': 'Rating'})
data.head()
```

	Tanggapan	Rating
1291	Bagus semua fitur berfungsi dengan baik, melay...	5
122	Aplikasi nya mudah di mengerti buat pemula sep...	5
125	Tampilannya simpel & mudah dimengerti, bebas a...	5
643	Jangan download sebelum anda menyesal, untung ...	1
937	Msih mncoba untuk prtma kali nya, mdah mudahan...	4

Gambar 4. Hasil filtering

Gambar 4 menunjukkan hasil filter pada kolom content dan score yang di ubah menjadi Tanggapan dan Rating.

c. Labeling Data

Proses untuk memberikan label berdasarkan rating yang di berikan oleh pengguna aplikasi GoPay. Dalam proses ini, sentimen yang di hasilkan berupa positif dan negatif. Script yang di gunakan dalam memberikan label ada pada Gambar 5.

```
def pelabelan(Rating):
    if Rating < 3:
        return 'Negatif'
    elif Rating == 4 :
        return 'Positif'
    elif Rating == 5 :
        return 'Positif'
data['Label'] = data ['Rating'].apply(pelabelan)
data.head()
```

Gambar 5. Proses labeling

Gambar 5 merupakan script untuk proses pelabelan dimana Rating dengan angka 3 akan di hilangkan untuk mempermudah sentimen. Rating dengan angka di bawah 3 bernilai Negatif dan untuk angka di atas 3 seperti 4 dan 5 bernilai Positif. Hasil dari pelabelan dapat di lihat pada Gambar 6.

	Tanggapan	Rating	Label
1291	Bagus semua fitur berfungsi dengan baik, melay...	5	Positif
122	Aplikasi nya mudah di mengerti buat pemula sep...	5	Positif
125	Tampilannya simpel & mudah dimengerti, bebas a...	5	Positif
643	Jangan download sebelum anda menyesal, untung ...	1	Negatif
937	Msh mncoba untuk prtma kali nya, mdah mudahan...	4	Positif

Gambar 6. Hasil labeling

Gambar 6 merupakan hasil labeling dari dataset yang dimiliki berupa label positif dan negatif berdasarkan rating yang di berikan.

d. Cleaning

Cleaning yang di maksud merupakan proses membersihkan data yang kosong atau tidak bernilai. Dalam proses ini, terlebih dahulu akan di lakukan pengecekan terhadap data yang kosong dengan *script* pada Gambar 7.

```
data.isnull().sum()

Tanggapan      0
Rating          0
Label           79
dtype: int64
```

Gambar 7. Proses cek data kosong

Gambar 7 menunjukkan bahwa pada kolom label ada 79 data yang kosong atau tidak bernilai sehingga di lakukan pembersihan data pada data kosong tersebut. Hasil setelah pembersihan data dapat di lihat pada Gambar 8.

```
data.isnull().sum()

Tanggapan      0
Rating          0
Label           0
dtype: int64

data.head()

Tanggapan  Rating  Label
0          Bagus semua fitur berfungsi dengan baik, melay...  5  Positif
1          Aplikasi nya mudah di mengerti buat pemula sep...  5  Positif
2          Tampilannya simpel & mudah dimengerti, bebas a...  5  Positif
3          Jangan download sebelum anda menyesal, untung ...  1  Negatif
4          Msh mncoba untuk prtma kali nya, mdah mudahan...  4  Positif
```

Gambar 8. Hasil cleaning data

Gambar 8 merupakan hasil dari cleaning pada label yang kosong. Setelah pembersihan data kosong tersisa 1421 data bersih yang akan di processing dari keseluruhan dataset.

3.2. Hasil PreProcessing

Data yang akan di lakukan *preprocessing* merupakan data hasil cleaning atau data yang tidak bernilai kosong. Hasil dari tahap *preprocessing* sebagai berikut.

a. Case Folding

Pada *case folding*, bukan hanya tanda baca dan huruf besar yang di hilangkan. Namun, simbol atau emoji yang ada dalam teks juga ikut di hilangkan.

index	Tanggapan	Rating	Label	text_clean
0	Bagus semua fitur berfungsi dengan baik, melayani semua media transfer	5	Positif	bagus semua fitur berfungsi dengan baik melayani semua media transfer
1	Aplikasi nya mudah di mengerti buat pemula seperti saya, cuma buat paki gopaylater nya kok tidak bisa semua transaksi atau susah apa saya yang tidak paham caranya ya?	5	Positif	aplikasi nya mudah di mengerti buat pemula seperti saya cuma buat paki gopaylater nya kok tidak bisa semua transaksi atau susah apa saya yang tidak paham caranya ya
2	Tampilannya simpel & mudah dimengerti, bebas admin transfer antar bank, transaksinya juga cepat. 🙌🙌🙌 Mantulll... Aplikasinya 🥰🥰🥰	5	Positif	tampilannya simpel mudah dimengerti bebas admin transfer antar bank transaksinya juga cepat mantulll aplikasinya
3	Jangan download sebelum anda menyesal, untung saya top up cuma sedikit, kalau top up gede haduhh gimana entarnya udah duit pas pasan saldo kagak masuk	1	Negatif	janngan download sebelum anda menyesal untung saya top up cuma sedikit kalau top up gede haduhh gimana entarnya udah duit pas pasan saldo kagak masuk
4	Msh mncoba untuk prtma kali nya, mdah mudahan bisa di gunakan dengan menu menu yg di sajikan ...	4	Positif	msh mncoba untuk prtma kali nya mdah mudahan bisa di gunakan dengan menu menu yg di sajikan

Gambar 9. Hasil case folding

Gambar 9 merupakan hasil dari penghilangan simbol atau emoji yang ada. Untuk hasil case folding bisa di lihat pada Tabel 2.

Tabel 2. Hasil case folding

Index	Tanggapan	Case Folding
0	Bagus semua fitur berfungsi dengan baik, melayani semua media transfer	bagus semua fitur berfungsi dengan baik melayani semua media transfer
2	Tampilannya simpel & mudah dimengerti, bebas admin transfer antar bank, transaksinya juga cepat. 🙌🙌🙌 Mantulll.... Aplikasinya 🥰🥰🥰	tampilannya simpel mudah dimengerti bebas admin transfer antar bank transaksinya juga cepat mantulll aplikasinya
3	Jangan download sebelum anda menyesal, untung saya top up cuma sedikit, kalau top up gede haduhh gimana entarnya udah duit pas pasan saldo kagak masuk	janngan download sebelum anda menyesal untung saya top up cuma sedikit kalau top up gede haduhh gimana entarnya udah duit pas pasan saldo kagak masuk

Tabel 2 menjelaskan perubahan signifikan dari proses case folding yang sebelumnya menggunakan

huruf kapital semua menjadi huruf kecil serta emoji yang turut menghilang.

b. *Stopword Removal*

Data yang di gunakan untuk *stopword* merupakan hasil dari *case folding*.

index	Tanggapan	Rating	Label	text_clean	text_StopWord
0	Bagus semua fitur berfungsi dengan baik, melayani semua media transfer	5	Positif	bagus semua fitur berfungsi dengan baik melayani semua media transfer	bagus fitur berfungsi melayani media transfer
1	Aplikasi nya mudah di mengerti buat pemula seperti saya, cuma buat pakai gopaylater nya kok tidak bisa semua transaksi atau susah, apa saya yang tidak paham caranya ya?	5	Positif	aplikasi nya mudah di mengerti buat pemula seperti saya cuma buat pakai gopaylater nya kok tidak bisa semua transaksi atau susah apa saya yang tidak paham caranya ya	aplikasi nya mudah mengerti pemula pakai gopaylater nya transaksi susah paham ya
2	Tampilannya simpel & mudah dimengerti, bebas acnin transfer antar bank, transaksinya juga cepet. 🙌🙌🙌 Mantulll... Aplikasinya 🙌🙌🙌	5	Positif	tampilannya simpel mudah dimengerti bebas admin transfer antar bank transaksinya juga cepet mantulll aplikasinya	tampilannya simpel mudah dimengerti bebas admin transfer bank transaksinya cepet mantulll aplikasinya
3	Jangan download sebelum anda menyesal, untung saya top up cuma sedikit, kalau top up gede haduhh gimana entarnya udah duit pas pasan saldo kagak masuk	1	Negatif	janngan download sebelum anda menyesal, untung saya top up cuma sedikit, kalau top up gede haduhh gimana entarnya udah duit pas pasan saldo kagak masuk	download menyesal untung top up top up gede haduhh gimana entarnya udah duit pas pasan saldo kagak masuk
4	Masih mencoba untuk prima kali nya, mdah mudahan bisa di gunakan dngan menu menu yg di sajikan ...	4	Positif	masih mencoba untuk prima kali nya mdah mudahan bisa di gunakan dngan menu menu yg di sajikan	masih mencoba prima kali nya mdah mudahan dngan menu menu yg di sajikan

Gambar 10. Hasil *stopword*

Gambar 10 merupakan hasil *text stopwords* setelah di lakukan *case folding*. Untuk perubahan lebih jelas dapat di lihat pada Tabel 3.

Tabel 3. Hasil perubahan *stopword*

Index	Tanggapan	Stopword
0	bagus semua fitur berfungsi dengan baik melayani semua media transfer	bagus fitur berfungsi melayani media transfer
2	tampilannya simpel mudah dimengerti bebas admin transfer antar bank transaksinya juga cepat mantulll aplikasinya	tampilannya simpel mudah dimengerti bebas admin transfer antar bank transaksinya cepat mantulll aplikasinya
3	janngan download sebelum anda menyesal untung saya top up cuma sedikit kalau top up gede haduhh gimana entarnya udah duit pas pasan saldo kagak masuk	download menyesal untung top up top up gede haduhh gimana entarnya udah duit pas pasan saldo kagak masuk

Tabel 3 merupakan hasil dari *stopword* dimana beberapa kata yang tidak relevan atau bermanfaat di hilangkan.

c. *Tokenizing*

Hasil dari *tokenizing* dapat di lihat pada Gambar 11.

Tanggapan	Rating	Label	text_clean	text_StopWord	text_tokens	
0	Bagus semua fitur berfungsi dengan baik, melay...	5	Positif	bagus semua fitur berfungsi dengan baik melay...	bagus semua fitur berfungsi dengan baik melay...	[bagus, semua, fitur, berfungsi, dengan, baik, ...]
1	Aplikasi nya mudah di mengerti buat pemula sep...	5	Positif	aplikasi nya mudah di mengerti buat pemula sep...	aplikasi nya mudah mengerti pemula pakai gopay...	[aplikasi, nya, mudah, di, mengerti, buat, pem...
2	Tampilannya simpel & mudah dimengerti, bebas a...	5	Positif	tampilannya simpel mudah dimengerti bebas adm...	tampilannya simpel mudah dimengerti bebas admi...	[tampilannya, simpel, mudah, dimengerti, bebas...
3	Jangan download sebelum anda menyesal, untung ...	1	Negatif	janngan download sebelum anda menyesal untung s...	download menyesal untung top up top up gede he...	[janngan, download, sebelum, anda, menyesal, untung, top, up, top, up, gede, he...

Gambar 11. Hasil *Tokenize*

Gambar 11 merupakan script dari *tokenize* beserta hasilnya. *Text tokens* berdasarkan *text clean* atau hasil *case folding* yang telah di lakukan. Pada hasil tersebut terjadi perubahan yang dapat di lihat pada Tabel 4.

Tabel 4. Hasil perubahan *tokenize*

Index	Tanggapan	Tokenize
0	bagus semua fitur berfungsi dengan baik melayani semua media transfer	[bagus, semua, fitur, berfungsi, dengan, baik, melayani, semua, media, transfer]
2	tampilannya simpel mudah dimengerti bebas admin transfer antar bank transaksinya juga cepat mantulll aplikasinya	[tampilannya, simpel, mudah, dimengerti, bebas, admin, transfer, antar, bank, transaksinya, juga, cepat, mantulll, aplikasinya]
3	janngan download sebelum anda menyesal untung saya top up cuma sedikit kalau top up gede haduhh gimana entarnya udah duit pas pasan saldo kagak masuk	[janngan, download, sebelum, anda, menyesal, untung, saya, top, up, cuma, sedikit, kalau, top, up, gede, haduhh, gimana, entarnya, udah, duit, pas, pasan, saldo, kagak, masuk]

Tabel 4 merupakan hasil dari *tokenize* yang mana kalimat sebelumnya pada hasil *case folding* di ubah menjadi per kata.

d. *Stemming*

Hasil *stemming* dapat di lihat pada Gambar 12.

```

4603 : tanding : tanding
4604 : csk : csk
4605 : instagram : instagram
4606 : tokennya : tokennya
4607 : munculll : munculll
4608 : melengkap : lengkap
4609 : ppob : ppob
4610 : diprint : diprint
4611 : diperbarui : baru
4612 : print : print
4613 : printer : printer
4614 : bluetooth : bluetooth
4615 : khilaf : khilaf
4616 : kopong : kopong
4617 : nambahin : nambahin
4618 : ngomongin : ngomongin
4619 : terkendala : kendala
4620 : surverisuer : surverisuer
4621 : Palembang : Palembang
4622 : mantapsangat : mantapsangat
4623 : murahcocok : murahcocok
4624 : kilatsaya : kilatsaya
4625 : myfrend : myfrend
4626 : seringlah : sering
4627 : alhadulillah : alhadulillah
4628 : bbrp : bbrp
    
```

Gambar 12. Hasil *stemming*

Gambar 12 merupakan hasil *stemming* dimana bentuk katanya akan kembali pada kata dasarnya seperti pada kata 'melengkap' : 'lengkap' dan kata 'diperbarui' : 'baru'.

3.3. Hasil *Splitting Data*

Data akan di bagi menjadi data training (*latih*) dan data testing (*uji*) dengan *script* yang ada pada Gambar 13.

```

from sklearn.model_selection import train_test_split
X_train, X_test, y_train, y_test = train_test_split(data_clean['Tanggapan'], data_clean['Label'],
                                                    test_size = 0.2,
                                                    random_state = 0)
    
```

Gambar 13. Proses *split data*

Gambar 13 merupakan proses membagi data *latih* dan data *uji* dimana *test size* 0.2 sama dengan 80:20%. 80% untuk data *train* dan 20% untuk data *test*. Hasil *splitting* data ada pada Gambar 14.

```
print (X_train.shape)
print (y_train.shape)
print (X_test.shape)
print (y_test.shape)

(1136,)
(1136,)
(285,)
(285,)
```

Gambar 14. Hasil *split* data

Gambar 14 menunjukkan bahwa data training sebanyak 1136 dan data testing 285 data.

3.4. Hasil Pembobotan Tf-Idf

Tahap pembobotan di lakukan dengan TfidfVectorizer seperti pada gambar 15.

```
from sklearn.feature_extraction.text import TfidfVectorizer
tfidf_vectorizer = TfidfVectorizer()
tfidf_train = tfidf_vectorizer.fit_transform(X_train)
tfidf_test = tfidf_vectorizer.transform(X_test)

from sklearn.feature_extraction.text import CountVectorizer
vectorizer = CountVectorizer()
vectorizer.fit(X_train)
```

Gambar 15. Proses tf-idf

Setelah proses tf-idf tersebut maka akan di lakukan normalisasi yang akan di tampilkan dalam bentuk array serta menampilkan vocabulary yang di gunakan dalam dataset seperti Gambar 16.

```
print(vectorizer.vocabulary_)
{'bunuk': 614, 'bgtr': 516, 'sumpah': 3192, 'saya': 2884, 'transfer': 3479, 'ke': 1474, 'bank': 352, 'btn': 598, 'lewat': 17}

X_train.toarray()
array([[0, 0, 0, ..., 0, 0, 0],
       [0, 0, 0, ..., 0, 0, 0],
       [0, 0, 0, ..., 0, 0, 0],
       ...,
       [0, 0, 0, ..., 0, 0, 0],
       [0, 0, 0, ..., 0, 0, 0],
       [0, 0, 0, ..., 0, 0, 0]])
```

Gambar 15. Hasil normalisasi tf-idf

3.5. Hasil Modelling

Pada tahap ini, akan di bangun model algoritma yang akan di gunakan dengan menggunakan *library* yang tersedia pada Google Colab.

```
from sklearn.svm import LinearSVC
svm = LinearSVC()
svm.fit(tfidf_train, y_train)

from sklearn.tree import DecisionTreeClassifier
tree = DecisionTreeClassifier()
tree.fit(tfidf_train, y_train)

from sklearn.ensemble import RandomForestClassifier
random = RandomForestClassifier()
random.fit(tfidf_train, y_train)
```

Gambar 16. Hasil *modelling*

Gambar 17 merupakan hasil modelling algoritma yang di gunakan yaitu *SVM*, *Decision Tree*, dan *Random Forest*.

3.6. Hasil Klasifikasi *Support Vector Machine*

Hasil Klasifikasi dengan menggunakan algoritma *Support Vector Machine* dapat di lihat pada Gambar 18.

	precision	recall	f1-score	support
Negatif	0.85	0.83	0.84	110
Positif	0.89	0.91	0.90	175
accuracy			0.88	285
macro avg	0.87	0.87	0.87	285
weighted avg	0.88	0.88	0.88	285

Gambar 17. Hasil klasifikasi *svm*

Gambar 18 merupakan hasil dari klasifikasi *SVM (Support Vector Machine)* dengan skor akurasi 88% yang di dapatkan melalui *script library* pada Gambar 19.

```
import pandas as pd
from sklearn.feature_extraction.text import TfidfVectorizer
from sklearn.svm import LinearSVC
from sklearn.metrics import accuracy_score, precision_score, recall_score, f1_score
from sklearn.model_selection import train_test_split
from sklearn.metrics import classification_report
from sklearn.metrics import confusion_matrix

clf = LinearSVC()
clf.fit(X_train, y_train)
predicted = clf.predict(X_test)

print(f'confusion_matrix:\n {confusion_matrix(y_test, predicted)}')
print('-----\n')
print(classification_report(y_test, predicted))
```

Gambar 18. *Script* klasifikasi *svm*

Gambar 19 merupakan *script* yang di gunakan untuk menghasilkan klasifikasi algoritma *SVM*.

3.7. Hasil Klasifikasi *Decision Tree*

Script yang di gunakan untuk menampilkan klasifikasi dengan algoritma *Decision Tree* dapat di lihat pada Gambar 20.

```
import pandas as pd
from sklearn.feature_extraction.text import TfidfVectorizer
from sklearn.tree import DecisionTreeClassifier
from sklearn.metrics import accuracy_score, precision_score, recall_score, f1_score
from sklearn.model_selection import train_test_split
from sklearn.metrics import classification_report
from sklearn.metrics import confusion_matrix

clf = DecisionTreeClassifier()
clf.fit(X_train, y_train)
predicted = clf.predict(X_test)

print(f'confusion_matrix:\n {confusion_matrix(y_test, predicted)}')
print('-----\n')
print(classification_report(y_test, predicted))
```

Gambar 19. Proses klasifikasi *Decision Tree*

Gambar 20 merupakan proses menampilkan hasil klasifikasi algoritma *Decision Tree*.

	precision	recall	f1-score	support
Negatif	0.76	0.83	0.79	110
Positif	0.89	0.84	0.86	175
accuracy			0.84	285
macro avg	0.83	0.83	0.83	285
weighted avg	0.84	0.84	0.84	285

Gambar 20. Hasil klasifikasi *Decision Tree*

Gambar 21 menunjukkan bahwa hasil akurasi yang di dapat adalah 84%.

3.8. Hasil Klasifikasi *Random Forest*

Library yang di gunakan untuk menampilkan hasil klasifikasi dengan algoritma *Random Forest* ada pada Gambar 22.

```
import pandas as pd
from sklearn.feature_extraction.text import TfidfVectorizer
from sklearn.ensemble import RandomForestClassifier
from sklearn.metrics import accuracy_score, precision_score, recall_score, f1_score
from sklearn.model_selection import train_test_split
from sklearn.metrics import classification_report
from sklearn.metrics import confusion_matrix

clf = RandomForestClassifier()
clf.fit(X_train, y_train)
predicted = clf.predict(X_test)

print(f'confusion_matrix:\n {confusion_matrix(y_test, predicted)}')
print('-----\n')
print(classification_report(y_test, predicted))
```

Gambar 21. Proses klasifikasi *Random Forest*

Dari proses pada gambar 22 mendapatkan hasil seperti pada Gambar 23.

	precision	recall	f1-score	support
Negatif	0.86	0.89	0.88	110
Positif	0.93	0.91	0.92	175
accuracy			0.90	285
macro avg	0.89	0.90	0.90	285
weighted avg	0.90	0.90	0.90	285

Gambar 22. Hasil klasifikasi *Random Forest*

Gambar 23 merupakan hasil klasifikasi dengan algoritma *Random Forest* yang mendapatkan skor akurasi 90%.

3.9. Hasil Evaluasi

Evaluasi dari penelitian ini akan di representasikan dengan menggunakan confusion matrix.

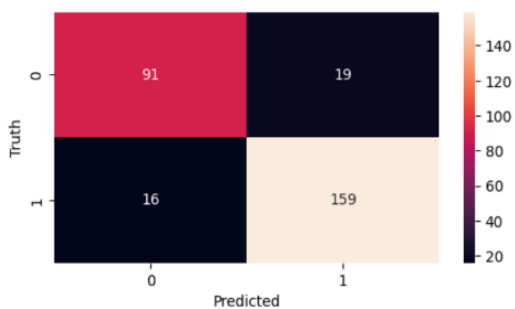
a. *Support Vector Machine (SVM)*

Evaluasi dengan confusion matrix di representasikan dengan nilai angka seperti pada Gambar 24.

```
confusion_matrix:
[[ 91  19]
 [ 16 159]]
```

Gambar 23. Hasil algoritma *svm*

Gambar 24 merupakan hasil dari algoritma *Svm* yang kemudian di buatkan visualisasi.



Gambar 24. Hasil visualisasi confusion matrix

Dari hasil tersebut dapat di jelaskan bahwa data sebenarnya di beri label Truth dan prediksi dari model

di beri label Predicted. Untuk TP: 159, TN: 91, FP: 19 dan FN: 16.

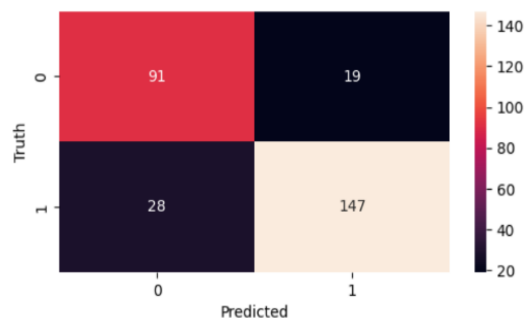
b. *Decision Tree*

Hasil dari confusion matrix dari klasifikasi algoritma *Decision Tree* dapat di lihat pada gambar berikut yang di representasikan dalam bentuk angka.

```
confusion_matrix:
[[ 91  19]
 [ 28 147]]
```

Gambar 25. Hasil algoritma *Decision Tree*

Gambar 26 merupakan hasil dari algoritma *Decision Tree* yang di visualisasikan seperti Gambar 27.



Gambar 26. Hasil visualisasi confusion matrix

Berdasarkan hasil dari Gambar 26 dan 27 dapat di jelaskan bahwa Untuk TP: 147, TN: 91, FP: 19 dan FN: 28.

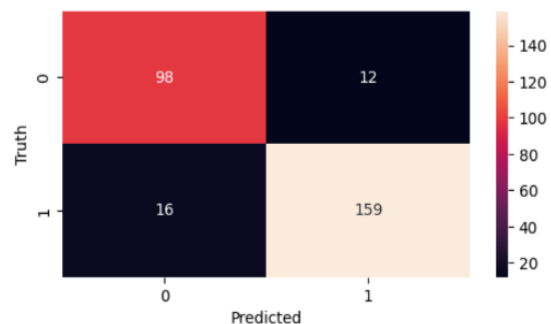
c. *Random Forest*

Hasil dari confusion matrix dari klasifikasi algoritma *Random Forest* dapat di lihat pada Gambar 28.

```
confusion_matrix:
[[ 98  12]
 [ 16 159]]
```

Gambar 27. Hasil algoritma *Decision Tree*

Gambar 28 merupakan hasil dari algoritma *Random Forest* yang di buatkan visualisasi seperti Gambar 29.



Gambar 28. Hasil visualisasi confusion matrix

Berdasarkan hasil tersebut dapat di jelaskan bahwa Untuk TP: 159, TN: 98, FP: 12 dan FN: 16.

3.10. Hasil Perbandingan Evaluasi

Perbandingan dari penggunaan ketiga algoritma klasifikasi yang di gunakan dapat di lihat pada tabel berikut.

Tabel 5. Hasil perbandingan tiga algoritma

Algoritma	SVM		Decision Tree		Random Forest	
	Pos	Neg	Pos	Neg	Pos	Neg
Sentimen Evaluasi						
Precision	0.89	0.85	0.89	0.76	0.93	0.86
Recall	0.91	0.83	0.84	0.83	0.91	0.89
F1-score	0.90	0.84	0.86	0.79	0.92	0.88
Accuracy	0.88		0.84		0.90	

4. DISKUSI

Penelitian dengan topik yang sama sudah banyak di lakukan oleh peneliti terdahulu dengan menggunakan beberapa algoritma klasifikasi seperti pada penelitian yang di lakukan oleh [24] yang tujuannya adalah untuk menentukan aplikasi e-wallet yang paling baik antara Dana dan Gopay berdasarkan pendapat pengguna aplikasinya dengan metode *Support Vector Machine (SVM)* yang mendapatkan hasil bahwa aplikasi Dana mempunyai akurasi 92% dan presisi 91%, dengan tingkat recall 92%. Sedangkan Gopay memiliki akurasi 90% dan presisi 91%, dengan tingkat recall 90%. Berdasarkan hasil tersebut menunjukkan bahwa aplikasi Gopay lebih baik dari Dana berdasarkan sentimen pengguna twitter.

Penelitian lain juga di lakukan oleh [3] yang bertujuan untuk membandingkan sentimen positif dan negatif terhadap komentar pengguna gopay pada twitter dengan menggunakan metode *lexicon Based* yang mendapatkan hasil bahwa 923 hasil positif dan 287 hasil negatif. Sementara itu, menggunakan metode SVM dengan kernel Linear menghasilkan akurasi sebesar 89,17% dan 84,38% untuk kernel Polynomial.

Hasil dari penelitian yang telah di lakukan menunjukkan bahwa sentimen Positif lebih banyak dari sentimen Negatif dimana pada SVM Positif 89% dan Negatif 85%, *Decision Tree* class Positif 89% dan Negatif 76% sedangkan pada *Random Forest* class positif 93% dan Negatif 86%. Selain itu tingkat akurasi yang tinggi ada pada algoritma *Random Forest* yaitu 90%, kemudian algoritma SVM 88% dan algoritma *Decision Tree* 84%.

5. KESIMPULAN

Berdasarkan penelitian yang telah di lakukan dari proses pengumpulan data sampai dengan tahap evaluasi mendapatkan kesimpulan sebagai berikut:

1. Dataset yang digunakan di awal adalah 1500 data yang kemudian di proses dengan di lakukan filtering dan cleaning sehingga menyisakan 1421 untuk di *preprocessing*. Dengan menggunakan Google Colab bahasa pemrograman python dapat mengambil atau

melakukan *scraping* data pada ulasan pengguna aplikasi GoPay di google play store.

2. Algoritma klasifikasi yang di gunakan adalah *Support Vector Machine (SVM)*, *Decision Tree*, dan *Random Forest* dengan hasil spliting data 80% data training (latih) dan 20% data testing (uji). Hasil klasifikasi ketiga algoritma tersebut akan di representasikan dalam bentuk evaluasi confusion matrix.
3. Algoritma *Support Vector Machine (SVM)* mendapatkan hasil precision positif 89% dan negatif 85%, recall positif 91% dan negatif 83%, f1-score positif 90% dan negatif 84% dengan score akurasi 88%.
4. Algoritma *Decision Tree* mendapatkan hasil precision positif 89% dan negatif 76%, recall positif 84% dan negatif 83%, f1-score positif 86% dan negatif 79% dengan score akurasi 84%.
5. Algoritma *Random Forest* mendapatkan hasil precision positif 93% dan negatif 86%, recall positif 91% dan negatif 89%, f1-score positif 92% dan negatif 88% dengan score akurasi 90%.

Dalam hasil yang di peroleh pada penelitian ini menunjukkan bahwa dari ketiga algoritma yang di gunakan sentimen Positif lebih besar dari sentimen Negatif. Selain itu tingkat akurasi yang tinggi ada pada *Random Forest*. Untuk penelitian selanjutnya di harapkan lebih baik lagi dengan mempertimbangkan penggunaan algoritma dan dataset serta tahap *preprocessing* datanya.

DAFTAR PUSTAKA

- [1] M. Ermanja, "Apa itu E-Wallet? Ini Pengertian dan Keuntungan Menggunakan Dompot Digital," *bayarind*, 2023. <https://www.bayarind.id/news/apa-itu-e-wallet-semua-yang-perlu-diketahui-tentang-e-wallet> (accessed Feb. 07, 2024).
- [2] C. E. Sedik, "4 Fitur GoPay Terbaru, Gratis Transfer hingga Keamanan Berlapis," *Bisnis.com*, 2023. https://m-bisnis-com.cdn.ampproject.org/v/s/m.bisnis.com/amp/read/20230727/101/1678960/4-fitur-gopay-terbaru-gratis-transfer-hingga-keamanan-berlapis?amp_gsa=1&_js_v=a9&usqp=mq331AQIUAKwASCAAgM%3D#amp_tf=Dari%25251%2524s&aoh=17073173293303&referrer=https%25 (accessed Dec. 07, 2024).
- [3] R. Mahendrajaya, G. A. Buntoro, and M. B. Setyawan, "Analisis Sentimen Pengguna Gopay Menggunakan Metode Lexicon Based Dan Support Vector Machine," *Komputek*, vol. 3, no. 2, p. 52, 2019, doi: 10.24269/jkt.v3i2.270.
- [4] D. Hariyanti, "Sehari Diluncurkan, GoPay

- Peringkat Satu Kategori Finance di App Store,” *Katadata*, 2023. https://katadata-co-id.cdn.ampproject.org/v/s/katadata.co.id/amp/dinihariyanti/digital/64c26f70069a5/sehari-diluncurkan-gopay-peringkat-satu-kategori-finance-di-app-store?amp_gsa=1&_js_v=a9&usqp=mq331AQIUAkwASCAAgM%3D#amp_tf=Dari%251%24s&aoh=170732098 (accessed Feb. 07, 2024).
- [5] A. R. R. Riskawati, Fatihanursari, Iin, “PENERAPAN METODE NAÏVE BAYES CLASSIFIER PADA ANALISIS SENTIMEN APLIKASI GOPAY,” *JATI (Jurnal Mhs. Tek. Inform.)*, vol. 8, no. 1, pp. 346–353, 2024, doi: <https://doi.org/10.36040/jati.v8i1.8699>.
- [6] N. Ambika Hapsari and A. Dwi Indriyanti, “Analisis Sentimen pada Aplikasi Dompot Digital Menggunakan Algoritma Random Forest,” *J. Emerg. Inf. Syst. Bus. Intell.*, vol. 04, no. 03, pp. 186–192, 2023.
- [7] H. Rizqiani, “Dampak Negatif dan Positif dari Penggunaan Dompot Digital,” *Ngiup*, 2021. <https://ngiup.com/2021/11/10/dampak-negatif-dan-positif-dari-penggunaan-dompot-digital/> (accessed Mar. 08, 2024).
- [8] F. A. Larasati, D. E. Ratnawati, and B. T. Hanggara, “Analisis Sentimen Ulasan Aplikasi Dana dengan Metode Random Forest,” ... *Tek. Inf. dan ...*, vol. 6, no. 9, pp. 4305–4313, 2022, [Online]. Available: <http://j-ptiik.ub.ac.id>
- [9] D. R. Alghifari, M. Edi, and L. Firmansyah, “Implementasi Bidirectional LSTM untuk Analisis Sentimen Terhadap Layanan Grab Indonesia,” *J. Manaj. Inform.*, vol. 12, no. 2, pp. 89–99, 2022, doi: [10.34010/jamika.v12i2.7764](https://doi.org/10.34010/jamika.v12i2.7764).
- [10] Aviliani, “Web Scraping: Alternatif Cari Data dengan Cepat,” *Pacmann*, 2022. <https://pacmann.io/blog/cari-data-dengan-murah-dan-cepat-menggunakan-web-scraping> (accessed Feb. 08, 2024).
- [11] V. Vamilina and R. Novita, “Analisis Sentimen E-Wallet Menggunakan Support Vector Machine Berbasis Particle Swarm Optimization,” *Buuld. Informatics, Technol. Sci.*, vol. 5, no. 1, pp. 40–48, 2023, doi: [10.47065/bits.v5i1.3526](https://doi.org/10.47065/bits.v5i1.3526).
- [12] B. Filemon, V. C. Mawardi, and N. J. Perdana, “Penggunaan Metode Support Vector Machine Untuk Klasifikasi Sentimen E-Wallet,” *J. Ilmu Komput. dan Sist. Inf.*, vol. 10, no. 1, 2022, doi: [10.24912/jiksi.v10i1.17824](https://doi.org/10.24912/jiksi.v10i1.17824).
- [13] R. Rinandyaswara, Y. A. Sari, and M. T. Furqon, “Pembentukan Daftar Stopword Menggunakan Term Based Random Sampling Pada Analisis Sentimen Dengan Metode Naïve Bayes (Studi Kasus: Kuliah Daring Di Masa Pandemi),” *J. Teknol. Inf. dan Ilmu Komput.*, vol. 9, no. 4, p. 717, 2022, doi: [10.25126/jtiik.2022934707](https://doi.org/10.25126/jtiik.2022934707).
- [14] D. Darwis, E. S. Pratiwi, and A. F. O. Pasaribu, “Penerapan Algoritma Svm Untuk Analisis Sentimen Pada Data Twitter Komisi Pemberantasan Korupsi Republik Indonesia,” *EduTic - Sci. J. Informatics Educ.*, vol. 7, no. 1, pp. 1–11, 2020, doi: [10.21107/edutic.v7i1.8779](https://doi.org/10.21107/edutic.v7i1.8779).
- [15] Oktavianus and M. Hondro, “Analisis Sentimen Pengguna Aplikasi E-Wallet Dana Melalui Postingan di Media Sosial Twitter Menggunakan Naïve Bayes,” *J. Inform.*, vol. 01, no. 01, pp. 27–31, 2023.
- [16] D. Sierra, “Algoritma TF — IDF,” *Medium*, 2019. <https://dltsierra.medium.com/algoritma-tf-idf-633e17d10a80> (accessed Nov. 01, 2023).
- [17] M. I. Fikri, T. S. Sabrila, Y. Azhar, and U. M. Malang, “Comparison of the Naïve Bayes Method and Support Vector Machine on Twitter Sentiment Analysis,” *SMATIKA J. STIKI Inform. J.*, vol. 10, no. 2, pp. 71–76, 2020.
- [18] Y. Refo, S. Rostianingsih, and L. Liliana, “Penerapan SVM untuk Klasifikasi Sentimen pada Review Comment Berbahasa Indonesia di Online Shop,” *J. Infra*, 2022, [Online]. Available: <https://publication.petra.ac.id/index.php/teknik-informatika/article/view/12813%0Ahttps://publication.petra.ac.id/index.php/teknik-informatika/article/download/12813/11113>
- [19] C. Cahyaningtyas, Y. Nataliani, and I. R. Widiyari, “Analisis Sentimen Pada Rating Aplikasi Shopee Menggunakan Metode Decision Tree Berbasis SMOTE,” *Aiti*, vol. 18, no. 2, pp. 173–184, 2021, doi: [10.24246/aiti.v18i2.173-184](https://doi.org/10.24246/aiti.v18i2.173-184).
- [20] I. A. D. Aji Susanto, “Analisis Sentimen Data Twitter Topik Ekonomi Dan Industri Dengan Metode Naive Bayes Dan Random Forest,” *J. Ilm. Wahana Pendidik.*, vol. 9, no. 20, pp. 59–65, 2023, doi: <https://doi.org/10.5281/zenodo.8398895>.
- [21] K. A. Nugraha, “Analisis Sentimen Berbasis Emoticon pada Komentar Instagram Bahasa Indonesia Menggunakan Naïve Bayes,” *J. Tek. Inform. dan Sist. Inf.*, vol. 7, no. 3, pp. 715–721, 2021, doi: [10.28932/jutisi.v7i3.4094](https://doi.org/10.28932/jutisi.v7i3.4094).
- [22] F. R. Irawan, A. Jazuli, and T. Khotimah,

- “Analisis Sentimen Terhadap Pengguna Gojek Menggunakan Metode K-Nearset Neighbors Sentiment Analysis of Gojek Users Using K-Nearest Neighbor,” *JIKO (Jurnal Inform. dan Komputer)*, vol. 5, no. 1, pp. 62–68, 2022, doi: 10.33387/jiko.
- [23] Y. Khoiruddin, A. Fauzi, and A. M. Siregar, “Analisis Sentimen Gojek Indonesia Pada Twitter Menggunakan Algoritme Naïve Bayes Dan Support Vector Machine,” *J. Ilm. Komput.*, vol. 19, pp. 391–400, 2023.
- [24] M. K. Rifa, M. H. Totohendarto, and M. R. Muttaqin, “Analisis Sentimen Pengguna E-Wallet Dana Dan Gopay Pada Twitter Menggunakan Metode Support Vector Machine (SVM),” *J. Tek.*, vol. 17, no. 2, pp. 323–332, 2023.