

Analisis Sentimen Twitter Tentang Wisata di Kota Solo

Twitter Sentiment Analysis About Tourism in the City of Solo

Siti Fatimah¹, Eko Purwanto², Hanifah Permatasari³

^{1,2,3}Studi Sistem Informasi, Fakultas Ilmu Komputer, Universitas Duta Bangsa Surakarta

E-mail: ¹190101056@fikom.udb.ac.id, ²eko_purwanto@udb.ac.id,

³hanifah_permatasari@udb.ac.id

Abstrak

Analisis sentimen digunakan untuk memahami pandangan dan pendapat masyarakat terhadap suatu hal, serta untuk mengidentifikasi tren topik pembicaraan. Dalam upaya merespon keluhan dan meningkatkan pelayanan kepada masyarakat, Dinas Pariwisata kota Solo aktif memanfaatkan berbagai platform media sosial seperti Twitter, Instagram, Facebook, web ULAS, dan Halo Mas Wali sebagai sarana komunikasi interaktif dengan masyarakat. Melalui media sosial ini, dapat mempermudah pegawai dalam proses pengumpulan sentimen masyarakat, yang nantinya akan menjadi dasar untuk menetapkan kebijakan dan strategi pembangunan. Oleh karena itu, diperlukan aplikasi analisis sentimen yang mampu mengklasifikasikan setiap tweet menjadi tiga kategori, yakni positif, netral dan negatif. Algoritma yang digunakan yaitu Naive Bayes. Naive Bayes adalah algoritma klasifikasi yang menggunakan probabilitas fitur untuk memprediksi label data, dengan asumsi fitur saling independen. Algoritma tersebut cocok untuk klasifikasi big data karena efisien pada data besar dan beragam. Hasil dari pembuatan aplikasi analisis sentimen ini adalah menunjukkan bahwa aplikasi dapat secara otomatis melabeli data menggunakan metode Naive Bayes berbasis PHP dan MySQL. Proses *crawling* data dari Twitter dilakukan melalui Google Colab menggunakan bahasa Python. Dalam implementasinya, aplikasi ini dapat membantu dalam melakukan klasifikasi opini masyarakat terhadap tempat wisata di kota Solo. Dilakukan pengujian validasi data untuk menguji kesamaan hasil perhitungan sistem dan manual menggunakan 3 data *training* dan 1 data *testing* yang didapatkan hasil klasifikasi data *testing* positif. Berdasarkan pengujian yang telah dilakukan dengan 500 data menggunakan metode naive bayes didapatkan hasil klasifikasi netral berjumlah 122 data, positif 302 data dan negatif 76 data dengan akurasi sebesar 70,2% %, presisi 67% dan *recall* 100%.

Kata kunci: Analisis sentimen, aplikasi, naive bayes, twitter, wisata solo

Abstract

Sentiment analysis is used to understand people's views and opinions on something and to identify trends in conversation topics. To respond to complaints and improve services to the community, the Solo City Tourism Office actively utilizes social media platforms such as Twitter, Instagram, Facebook, ULAS web, and Halo Mas Wali as a means of interactive communication with the community. This social media can make it easier for employees to collect public sentiment, which will later become the basis for determining development policies and strategies. Therefore, a sentiment analysis application is needed to classify each tweet into three categories: positive, neutral, and negative. The algorithm used is Naive Bayes. Naive Bayes is a classification algorithm that uses feature probabilities to predict data labels, assuming the features are independent. This algorithm is suitable for big data classification because it efficiently uses large and diverse data. The results of creating this sentiment analysis application show that the application can automatically label data using the Naive Bayes method based on PHP and MySQL. The crawling data from Twitter is carried out through Google Colab using Python. In its implementation, this application can help classify public opinion regarding tourist attractions in Solo. Data validation testing was carried out to test the similarity of system and manual calculation results using three training data and one testing data, which resulted in positive testing data classification results. Based on tests with 500 data using the Naive Bayes method,

the results obtained were 122 neutral classification data, 302 positive data, and 76 negative data with an accuracy of 70.2%, precision of 67%, and recall of 100%.

Keywords: Sentiment analysis, apps, naive bayes, twitter, solo travel

1. PENDAHULUAN

Twitter merupakan *platform* media sosial dimana banyak orang dari dapat mengungkapkan pendapat mereka melalui *tweet* yang ditulis [1]. Pengguna *Twitter* telah meningkat pada tahun-tahun terakhir dan telah menjadi aplikasi yang penting bagi pebisnis dan masyarakat untuk berinteraksi secara luas. *Twitter* juga dapat digunakan untuk tujuan survei, karena *tweet* yang dihasilkan dapat memberikan informasi tentang sentimen yang dipandang oleh masyarakat.

Kota Solo atau Surakarta adalah salah satu Kotamadya dengan populasi sebanyak 503,421 penduduk dan masuk golongan kota kelas menengah yang terus mengalami perkembangan [2]. Kota Solo merupakan salah satu destinasi pariwisata di Indonesia menawarkan beragam jenis pengalaman, mulai dari keindahan alam, warisan sejarah, kekayaan budaya, maupun buatan [3]. Kota Solo terkenal dengan tempat-tempat bersejarah, beberapa tempat wisata kota Solo yang populer dengan tempat bersejarah yaitu Keraton Surakarta Hadiningrat, Gedung Loji, Benteng Vastenburg dan Mangkunegaran. Selain itu Kota Solo juga mempunyai destinasi wisata lainnya seperti wisata budaya, wisata kuliner, wisata kampung batik, dan masih banyak lainnya.

Sektor pariwisata di Kota Solo terus meningkat, pada awal tahun 2023 Kota Solo sudah meresmikan beberapa tujuan pariwisata yang menarik perhatian pengunjung untuk dikunjungi. Mulai dari Masjid Raya Sheikh Zayed, Taman Pracimara Twin yang berada di Pura Mangkunegaran, hingga Taman Jurug yang sekarang menjadi Solo Safari. Jumlah pengunjung wisata di Solo sebelum terjadi wabah covid-19 masih normal dan terus meningkat, namun usai wabah covid-19 jumlah pengunjung mengalami penurunan drastis pada tahun 2020 [4].

Dibutuhkannya pendapat dan opini dari wisatawan yang berkunjung untuk meningkatkan daya tarik pengunjung agar banyak yang berkunjung selain untuk meningkatkan daya tarik dapat juga meningkatkan kualitas tempat wisata di Kota Solo. Hal ini menimbulkan pentingnya pemerintah kota Solo untuk mengetahui sentimen masyarakat terhadap wisata di kota Solo. Namun untuk memahami pandangan masyarakat tentang wisata di kota Solo, dibutuhkan lebih dari sekedar menganalisis *tweet*. Sebuah analisis sentimen *twitter* juga perlu dilakukan. Analisis sentimen *Twitter* merupakan proses ekstraksi pendapat publik mengenai suatu topik tertentu, produk, atau layanan yang di dalamnya terdapat teks-teks yang tidak terstruktur. [5].

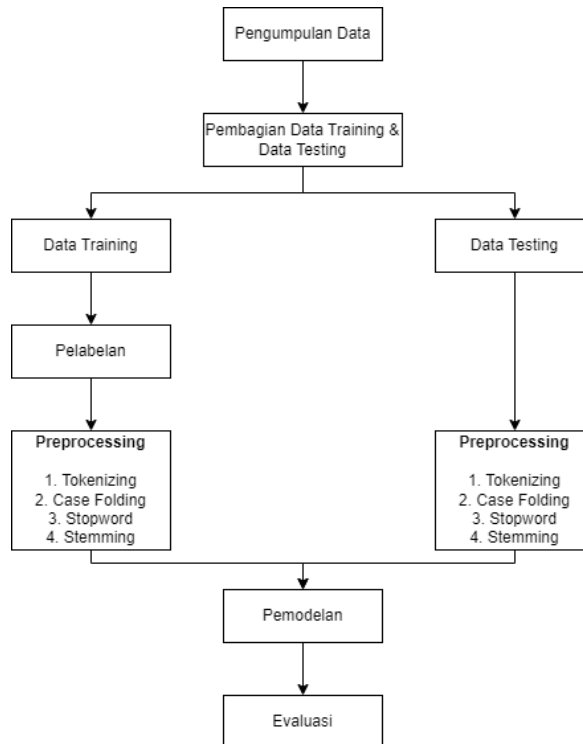
Analisis sentimen *twitter* dapat membantu untuk mengetahui bagaimana tanggapan masyarakat tentang kota Solo. Dengan demikian, pemerintah atau para pembuat kebijakan di kota Solo dapat mengambil keputusan yang tepat berdasarkan pandangan masyarakat. Saat ini, belum diketahui seberapa banyak *tweet* di *Twitter* yang memiliki opini positif, negatif, dan netral. Oleh karena itu, diperlukan sebuah sistem yang mampu mengolah data dengan menerapkan analisis sentimen. [6].

Analisis pada penelitian sebelumnya menggunakan metode Valence Aware Dictionary and Sentiment Reasoner (VADER) [7], metode Support Vector Machine (SVM) [8], metode Naïve Bayes dan K-Nearest Neighbor (KNN) [9], sedangkan analisis pada penelitian ini dilakukan menggunakan metode Naïve Bayes serta menggunakan SNScrape dengan python.

Tujuan dari penelitian ini adalah untuk membangun aplikasi web sentimen analisis *twitter* tentang pandangan masyarakat terhadap wisata di Kota Solo menggunakan metode *Naïve Bayes*, yang diharapkan dapat memudahkan pemerintah kota Solo atau para pembuat kebijakan di kota Solo dalam mengambil keputusan tersebut.

2. METODE PENELITIAN

Tahapan yang dilakukan dalam penelitian ini mencakup beberapa tahap sebagai berikut:



Gambar 1. Tahapan Penelitian

2.1 Pengumpulan Data

Pada penelitian ini dataset diambil dari data *tweet* berupa komentar masyarakat mengenai wisata di Kota Solo. Pengumpulan data dilakukan melalui proses *scrapping* data dari *tweet* menggunakan *library snsrape* dengan *keyword* ‘wisata solo’.

2.2 Pembagian *Data Training* & *Data Testing*

Membagi dataset menjadi dua bagian yaitu data *training* (data latih) dan data *testing* (data uji) dengan perbandingan 80:20.

2.3 Pelabelan

Labelling merupakan teknik untuk memberikan label sentimen terhadap masing masing data *tweet*. Teknik ini diterapkan dengan memilih orientasi data apakah masuk dalam sentimen positif, netral, atau negatif. Proses labelling ini dilakukan menggunakan bantuan Indonesia Sentiment Lexicon (InSet), yang merupakan kamus leksikon berbahasa Indonesia yang khusus digunakan untuk analisis sentimen pada teks, terutama teks yang berbasis di sosial media.

2.4 *Preprocessing*

Text Processing adalah menyiapkan dokumen dalam format teks yang mulanya tidak terstruktur menjadi format terstruktur sehingga dapat diolah dan diproses[10].

Adapun proses dalam *text processing* yaitu [11]:

a. *Tokenizing*

Tokenisasi adalah memotong atau memisahkan kalimat menjadi kata.

b. *Case Folding*

Case folding yaitu mengubah seluruh karakter huruf besar maupun kecil menjadi huruf kecil semua.

c. *Stopwords*

Stopword adalah kata-kata non-deskriptif yang dapat dihilangkan dari sebuah kalimat. Dapat dipahami bahwa kata pada *stopword* merupakan kata yang tidak mempunyai arti tersendiri. Contoh *stopwords* adalah “dan”, “yang”, “adalah”, “di”, “ke”, “dari” dan lain sebagainya.

d. *Stemming*

Stemming merupakan pemotongan imbuhan yang melekat pada kata. Kata-kata dalam kalimat akan menjadi kata dasar. Tujuan dari *stemming* adalah mengelompokkan kata-kata yang mempunyai kata dasar yang sama dan makna yang sama, tetapi mempunyai bentuk yang berbeda karena adanya imbuhan yang berbeda.

2.5 Pemodelan

Pemodelan dilakukan dengan metode pembobotan TF-IDF dan metode Klasifikasi Naive Bayes.

Term Frequency dan *Invers Document Frequency* (Tf-Idf) adalah metode untuk menghitung bobot setiap kata pada seluruh dokumen [12].

Rumus IDF yaitu [12]:

$$IDF = \log \frac{D}{D_{fi}}$$

Keterangan:

IDF = Invers dari nilai DFi

D = Jumlah *tweet* dalam dataset

Dfi = Jumlah *tweet* pada dataset yang berisi kata ke-i

Contoh table perhitungan IDF seperti berikut :

Tabel 1 Tabel IDF

term	LOG N/DF
indah	0,7781513
banget	0,7781513
wisata	0
religi	0,7781513
bulan	0,7781513
ramadhan	0,7781513
masjid	0,7781513
sheikh	0,7781513
zayed	0,7781513
solo	0
bikin	0,7781513

Nilai TF-IDF adalah:

$$TFIDF = TF \times IDF$$

Keterangan:

TF-IDF = Bobot kata dalam suatu kumpulan data

TF = Jumlah seberapa sering sebuah kata muncul pada *tweet*

IDF = Invers jumlah kata muncul dalam sebuah dataset

Contoh tabel perhitungan TFIDF adalah sebagai berikut :

Tabel 2. TF - IDF

term	D1	D2	D3	D4	D5	D6
	positif	netral	negatif	positif	netral	negatif
indah	0,7781513	0	0	0	0	0
banget	0,7781513	0	0	0	0	0
wisata	0	0	0	0	0	0
religi	0,7781513	0	0	0	0	0
bulan	0,7781513	0	0	0	0	0
ramadhan	0,7781513	0	0	0	0	0
masjid	0,7781513	0	0	0	0	0
sheikh	0,7781513	0	0	0	0	0
zayed	0,7781513	0	0	0	0	0
solo	0	0	0	0	0	0
bikin	0,7781513	0	0	0	0	0

Naïve Bayes merupakan metode klasifikasi probabilistik yang sederhana, yang berdasarkan Teorema Bayes dengan anggapan bahwa tidak ada ketergantungan yang signifikan antar kondisi atau kejadian [13].

Adapun Rumus Naive Bayes yang digunakan yaitu sebagai berikut [14]:

- 1) Persamaan dari teorema Bayes

$$P(c|x) = \frac{P(c)P(c)}{P(x)} \tag{1}$$

Dimana:

- x : sample data yang memiliki sentimen (label) yang tidak diketahui
- c : hipotesa bahwa X adalah data kelas(label)
- P(c) : peluang dari hipotesa
- c P(x|c) : peluang dari data sampel X bila diasumsikan bahwa hipotesa benar

- 2) Perhitungan Probabilitas Prior setiap kelas, dengan rumus:

$$P_c = \frac{N_c}{N} \tag{2}$$

Keterangan:

- N_c = Jumlah label klasifikasi c pada semua dokumen
- N = Jumlah keseluruhan dokumen yang ada

- 3) Perhitungan Probabilitas Kata ke-n, dengan rumus:

$$P = \frac{W_{ct} + 1}{N_c + Total\ Dok\ Term} \tag{3}$$

Keterangan:

- W_{ct} = Total nilai pembobotan TF IDF
- N_c = Jumlah label klasifikasi c pada semua dokumen
- $Total\ Dok\ Term$ = Jumlah kata pada semua dokumen

- 4) Perhitungan Probabilitas Dokumen menentukan kelas, dengan rumus:

$$P \frac{c}{term\ dokumen\ d} = P(c) x P \frac{t_1}{c} x P \frac{t_2}{c} x \dots P \frac{t_n}{c} \tag{4}$$

Keterangan:

- $P(c/term\ dokumen\ d)$ = probabilitas dokumen yang tergolong label c

$P(c)$ = probabilitas dokumen label c
 t_n = probabilitas kata pada dokumen ke-n
 $P(t_n/C)$ = probabilitas kemunculan kata ke-n dari label c

2.6 Evaluasi

Tahap selanjutnya dalam penelitian ini yaitu melakukan evaluasi. Evaluasi dilakukan dengan menghitung akurasi, presisi dan *recall* menggunakan *Confusion Matrix*..

3. HASIL DAN PEMBAHASAN

Hasil dari penelitian ini berdasarkan tahapan penelitian pada Gambar 1 dijelaskan sebagai berikut.

3.1 Pengumpulan Data

Pengumpulan data dilakukan melalui proses *scrapping* data dari *tweet* menggunakan *library ns scrape* dengan *keyword* 'wisata solo'. Adapun kode program sebagai berikut:

```
tweet_list = []

for i,tweet in enumerate(sntwitter.TwitterSearchScrapper('wisata solo since:2022-12-01
until:2023-06-01 lang:id').get_items()):
    if i>1000:
        break
    tweet_list.append([tweet.date, tweet.content])

#membuat dataframe dengan nama yg sesuai yang ditemukan
dataset = pd.DataFrame(tweet_list, columns=['datetime','tweet'])
dataset

#simpan kedalam csv
dataset.to_csv('crawling_1000.csv', sep=',', index=False)
```

Data yang diambil sebanyak 1000 tweet dengan kata kunci 'wisata solo' didapatkan 1000 data tweet dengan periode waktu 23 Februari 2023 – 31 Mei 2023.

3.2 Pembagian *Data Training & Data Testing*

Kemudian data yang sudah didapatkan dibagi menjadi data *training* (data latih) dan data *testing* (data uji) dengan perbandingan 80:20. Adapun kode program sebagai berikut:

```
# Membaca file CSV
data = pd.read_csv('crawling_1000.csv')

# Menghitung jumlah data untuk pelatihan dan pengujian
total_data = len(data)
train_size = int(0.8 * total_data)
test_size = total_data - train_size

# Membagi data menjadi data pelatihan dan data pengujian
train_data = data[:train_size]
test_data = data[train_size:]
```

3.3 Pelabelan

Pelabelan dilakukan pada *data training* yang terdiri dari label positif, negatif dan netral. Berikut sampel *data training* yang diberi label.

Tabel 3. Contoh Pelabelan Data Training

Dok	Tweet	Label
D1	solo riding ke pantai indrayanti lewat jls part travel motovlog wisata pantai gunung kidul	netral
D2	bakso jalan kalo solo raya banyak rb tp kadang gak enak apalagi tempat wisata tp bakso warung masih ada ribu malah lebih enak	negatif
D3	indah banget wisata religi bulan ramadhan masjid sheikh zayed solo bikin hati jadiadem	positif

3.4 Preprocessing

Tahapan selanjutnya adalah *preprocessing*, Tahap *preprocessing* yang dilakukan diantaranya *tokenizing*, *case folding*, *stopword*, *stemming*. Tabel 2 menunjukkan hasil dari *preprocessing* data.

Tabel 4. Hasil Preprocessing

Tweet Mentah	Preprocessing	Hasil
@Dany2Oka @tanyakanrl Bakso jalanan kalo Solo Raya juga banyak yg 10rb tp kadang gak enak, apalagi di tempat wisata.. tp bakso warung masih ada yang 5ribu malah lebih enak..	Tokenizing	'Bakso', 'jalanan', 'kalo', 'Solo', 'Raya', 'juga', 'banyak', 'yg', 'rb', 'tp', 'kadang', 'gak', 'enak', 'apalagi', 'di', 'tempat', 'wisata', 'tp', 'bakso', 'warung', 'masih', 'ada', 'yang', 'ribu', 'malah', 'lebih', 'enak'
	Case Folding	bakso jalanan kalo solo raya juga banyak yg rb tp kadang gak enak apalagi di tempat wisata tp bakso warung masih ada yang ribu malah lebih enak
	Stopword	bakso jalanan kalo solo raya banyak rb tp kadang gak enak apalagi tempat wisata tp bakso warung masih ada ribu malah lebih enak
	Stemming	bakso jalan kalo solo raya banyak rb tp kadang gak enak apalagi tempat wisata tp bakso warung masih ada ribu malah lebih enak

3.5 Pemodelan

Proses pemodelan melibatkan penggunaan metode TF-IDF untuk memberikan bobot pada kata dan penggunaan metode Naive Bayes untuk melakukan klasifikasi. Adapun sampel data terdapat pada tabel 3 berikut.

Tabel 5. Sampel Data

Dok	Tweet	Label
Data Training		
D1	solo riding ke pantai indrayanti lewat jls part travel motovlog wisata pantai gunung kidul	netral
D2	bakso jalan kalo solo raya banyak rb tp kadang gak enak apalagi tempat wisata tp bakso warung masih ada ribu malah lebih enak	negatif
D3	indah banget wisata religi bulan ramadhan masjid sheikh zayed solo bikin hati jadiadem	positif
Data Testing		
D4	masjid raya syekh zayed solo jadi tempat wisata religi baru kota surakarta	?

Proses Selanjutnya adalah menghitung total TF-IDF dengan rumus:

$$W_{at} = tf_{at} \times idf_{at} = tf_{at} \times \log(N/df_t)$$

Tabel 6. Hasil Perhitungan TF-IDF

Term	Doc1	Doc2	Doc3
solo	0	0	0
riding	0.477	0	0
ke	0.477	0	0
pantai	0.954	0	0
indrayanti	0.477	0	0
lewat	0.477	0	0
jls	0.477	0	0
part	0.477	0	0
travel	0.477	0	0
motovlog	0.477	0	0
wisata	0	0	0
gunung	0.477	0	0
bakso	0	0.954	0
jalan	0	0.477	0
kalo	0	0.477	0
raya	0	0.477	0
banyak	0	0.477	0
rb	0	0.477	0
tp	0	0.954	0
kadang	0	0.477	0
gak	0	0.477	0
enak	0	0.954	0
apalagi	0	0.477	0
tempat	0	0.477	0
warung	0	0.477	0
masih	0	0.477	0
ada	0	0.477	0
ribu	0	0.477	0
malah	0	0.477	0
lebih	0	0.477	0
indah	0	0	0.477
banget	0	0	0.477
religi	0	0	0.477
bulan	0	0	0.477
ramadhan	0	0	0.477
masjid	0	0	0.477
sheikh	0	0	0.477
zayed	0	0	0.477
bikin	0	0	0.477
hati	0	0	0.477

Tabel 4 merupakan hasil perhitungan TF-IDF dari setiap kata. Proses selanjutnya yaitu menghitung nilai prior dari setiap klasifikasi pada data *training* pada tabel 3. Jumlah data training ada 3 yang berlabel positif ada 1 data, netral ada 1 data dan negatif ada 1 data. Nilai prior dihitung dengan persamaan (1) hasil nilai prior terdapat pada tabel 5.

Tabel 7. Nilai Prior setiap klasifikasi

Klasifikasi	P(ci)
Netral	0.3333333333333333
Negatif	0.3333333333333333

Klasifikasi	P(ci)
Positif	0.33333333333333

Selanjutnya menghitung nilai probabilitas setiap kata pada data *training*, dihitung menggunakan algoritma Naive Bayes dengan persamaan (2) didapatkan nilai probabilitas pada tabel 6 berikut.

Tabel 8. Probabilitas data training setiap kata

Klasifikasi	Netral	Negatif	Positif
solo	0.0243	0.0243	0.0243
riding	0.0360	0.0243	0.0243
ke	0.0360	0.0243	0.0243
pantai	0.0476	0.0243	0.0243
indrayanti	0.0360	0.0243	0.0243
lewat	0.0360	0.0243	0.0243
jls	0.0360	0.0243	0.0243
part	0.0360	0.0243	0.0243
travel	0.0360	0.0243	0.0243
motovlog	0.0360	0.0243	0.0243
wisata	0.0243	0.0243	0.0243
gunung	0.0360	0.0243	0.0243
bakso	0.0243	0.0476	0.0243
jalan	0.0243	0.0360	0.0243
kalo	0.0243	0.0360	0.0243
raya	0.0243	0.0360	0.0243
banyak	0.0243	0.0360	0.0243
rb	0.0243	0.0360	0.0243
tp	0.0243	0.0476	0.0243
kadang	0.0243	0.0360	0.0243
gak	0.0243	0.0360	0.0243
enak	0.0243	0.0476	0.0243
apalagi	0.0243	0.0360	0.0243
tempat	0.0243	0.0360	0.0243
warung	0.0243	0.0360	0.0243
masih	0.0243	0.0360	0.0243
ada	0.0243	0.0360	0.0243
ribu	0.0243	0.0360	0.0243
malah	0.0243	0.0360	0.0243
lebih	0.0243	0.0360	0.0243
indah	0.0243	0.0243	0.0360
banget	0.0243	0.0243	0.0360
religi	0.0243	0.0243	0.0360
bulan	0.0243	0.0243	0.0360
ramadhan	0.0243	0.0243	0.0360
masjid	0.0243	0.0243	0.0360
sheikh	0.0243	0.0243	0.0360
zayed	0.0243	0.0243	0.0360
bikin	0.0243	0.0243	0.0360
hati	0.0243	0.0243	0.0360

Dari hasil perhitungan probabilitas pada Tabel 8 selanjutnya akan digunakan untuk memprediksi data *testing*, dengan data testing seperti pada tabel 9. Selanjutnya dihitung bobot *term data testing* berdasarkan kata dari *data training*.

Tabel 9. Bobot term data testing

Dokumen	D4
solo	1
riding	0
ke	0
pantai	0
indrayanti	0
lewat	0
jls	0
part	0
travel	0
motovlog	0
wisata	1
gunung	0
bakso	0
jalan	0
kalo	0
raya	1
banyak	0
rb	0
tp	0
kadang	0
gak	0
enak	0
apalagi	0
tempat	1
warung	0
masih	0
ada	0
ribu	0
malah	0
lebih	0
indah	0
banget	0
religi	1
bulan	0
ramadhan	0
masjid	1
sheikh	0
zayed	1
bikin	0
hati	0

Selanjutnya dihitung probabilitas data *testing* berdasarkan pada probabilitas data *training* di Tabel 8 dan juga bobot *term* data *testing* di Tabel 9.

Selanjutnya dihitung nilai probabilitas setiap kata.

Tabel 10. Probabilitas Data *Testing*

Label	probabilitas	indah	banget	wisata	religi	bulan	ramadhan
positif	0,333333333	1	1	0,017857	0,031753	1	1
netral	0,333333333	1	1	0,017857	0,017857	1	1
negatif	0,333333333	1	1	0,017857	0,017857	1	1

Label	probabilitas	masjid	sheikh	zayed	solo	bikin	hati
positif	0,333333333	0,031753	1	0,031753	0,017857	1	1
netral	0,333333333	0,017857	1	0,017857	0,017857	1	1
negatif	0,333333333	0,017857	1	0,017857	0,017857	1	1

Label	probabilitas	riding	ke	pantai	indrayanti	lewat	jls
positif	0,333333333	1	1	1	1	1	1
netral	0,333333333	1	1	1	1	1	1
negatif	0,333333333	1	1	1	1	1	1

Label	probabilitas	part	travel	motovlog	gunung	bakso	jalan
positif	0,333333333	1	1	1	1	1	1
netral	0,333333333	1	1	1	1	1	1
negatif	0,333333333	1	1	1	1	1	1

Label	probabilitas	kalo	raya	banyak	rb	tp	kadang
positif	0,333333333	1	0,017857	1	1	1	1
netral	0,333333333	1	0,017857	1	1	1	1
negatif	0,333333333	1	0,031753	1	1	1	1

Label	probabilitas	gak	enak	apalagi	tempat	warung	masih
positif	0,333333333	1	1	1	0,017857	1	1
netral	0,333333333	1	1	1	0,017857	1	1
negatif	0,333333333	1	1	1	0,031753	1	1

Label	probabilitas	ada	ribu	malah	lebih	kuliner	daripada
positif	0,333333333	1	1	1	1	1	1
netral	0,333333333	1	1	1	1	1	1
negatif	0,333333333	1	1	1	1	1	1

Label	probabilitas	jogja	bagus	kurang	motor	ajantar	maju
positif	0,333333333	1	1	1	1	1	1
netral	0,333333333	1	1	1	1	1	1
negatif	0,333333333	1	1	1	1	1	1

Label	probabilitas	sekarang	cuma	sayang	frekuensi	berangkat	krl
positif	0,33333333	1	1	1	1	1	1
netral	0,33333333	1	1	1	1	1	1
negatif	0,33333333	1	1	1	1	1	1

Tabel 11. Probabilitas data training

Label	Netral	Negatif	Positif
Probabilitas	0,33333	0,33333	0,33333
solo	0,02439	0,02439	0,02439
riding	1	1	1
ke	1	1	1
pantai	1	1	1
indrayanti	1	1	1
lewat	1	1	1
jls	1	1	1
part	1	1	1
travel	1	1	1
motovlog	1	1	1
wisata	0,02439	0,02439	0,02439
gunung	1	1	1
bakso	1	1	1
jalan	1	1	1
kalo	1	1	1
raya	0,02439	0,0360	0,02439
banyak	1	1	1
rb	1	1	1
tp	1	1	1
kadang	1	1	1
gak	1	1	1
enak	1	1	1
apalagi	1	1	1
tempat	0,02439	0,0360	0,02439
warung	1	1	1
masih	1	1	1
ada	1	1	1
ribu	1	1	1
malah	1	1	1
lebih	1	1	1
indah	1	1	1
banget	1	1	1
religi	0,02439	0,02439	0,0360
bulan	1	1	1
ramadhan	1	1	1
masjid	0,02439	0,02439	0,0360

Label	Netral	Negatif	Positif
sheikh	1	1	1
zayed	0,02439	0,02439	0,0360
bikin	1	1	1
hati	1	1	1
Total	1,7115E-12	3,7344E-12	5,5162E-12

Data Testing			
Total			
Dokumen	Netral	Positif	Negatif
D1	1.711558502367E-12	5.5162023678802E-12	3.7344275903247E-12
Klasifikasi Positif			
No	Testing ke-	Dokumen	
1	1	masjid raya syekh zayed solo jadi tempat wisata religi baru kota surakarta	

Gambar 2. Hasil Klasifikasi sistem dengan sampel data

Pada Tabel 10 dapat dianalisis hasil dari *testing* didapatkan nilai probabilitas positif lebih besar dari nilai probabilitas netral dan negatif sehingga dapat disimpulkan bahwa data testing pada Tabel 10 tersebut termasuk dalam kategori positif.

Tabel 12. Hasil analisis sampel data testing

Dok	<i>Tweet</i>	Label
D4	masjid raya syekh zayed solo jadi tempat wisata religi baru kota surakarta	Positif

Pada Tabel 9 dan Gambar 2 disimpulkan bahwa perhitungan manual dengan perhitungan sistem menghasilkan nilai dan klasifikasi yang sama.

3.6 Evaluasi

Pengujian *confusion matrix* sebagai metode evaluasi untuk mengukur kinerja dan keakuratan hasil klasifikasi sistem. Hasil pengujian *confusion matrix* dengan mengklasifikasikan 500 data menggunakan klasifikasi naive bayes pada sistem adalah sebagai berikut.

Tabel 13. Tabel Confusion Matrix

Klasifikasi	TP	FP	TN	FN
Positif	302	149	49	0
Negatif	26	0	424	50
Netral	23	0	378	99

Total Klasifikasi	
Klasifikasi	Total
Netral	122
Positif	302
Negatif	76

Total data testing: 500
 Total Prediksi yang benar: 351
 Akurasi: 70.2 %

Statistic
 Execution Time: 48.92334318161 seconds
 Memory Usage: 222168.1875 kilo bytes

Confusion Matrix							
Klasifikasi	TP	FP	TN	FN	Accuracy	Precision	Recall
Positif	302	149	49	0	0.702	0.67	1
Negatif	26	0	424	50	0.9	1	0.342
Netral	23	0	378	99	0.802	1	0.189

Akurasi			
Aktual / Prediksi	Positif	Negatif	Netral
Positif	302	0	0
Negatif	50	26	0
Netral	99	0	23

Akurasi = 70.2%

Gambar 3. Confusion matrix pada sistem

- a. *Accuracy*
- $$= \frac{TP + TN}{TP + TN + FP + FN}$$
- $$= \frac{302 + 49}{302 + 49 + 149 + 0}$$
- $$= \frac{351}{500} \times 100\%$$
- $$= 70,2\%$$
- b. *Precision*
- $$= \frac{TP}{TP + FP}$$
- $$= \frac{302}{302 + 149}$$
- $$= \frac{302}{451} \times 100\%$$
- $$= 67\%$$
- c. *Recall*
- $$= \frac{TP}{TP + FN}$$
- $$= \frac{302}{302 + 0}$$
- $$= \frac{302}{302} \times 100\%$$
- $$= 100\%$$

Dari pengujian *confusion matrix* tersebut didapatkan akurasi sebesar 70,2 %, presisi 67% dan *recall* sebesar 100%.

4. KESIMPULAN DAN SARAN

Berdasarkan hasil penelitian yang telah dilakukan, maka dapat disimpulkan sebagai berikut:

- a. Metode analisis klasifikasi yang digunakan adalah Naive Bayes.

- b. Data yang digunakan merupakan hasil dari pengumpulan *tweet* dengan pencarian kata kunci “wisata solo” sebanyak 1000 *tweet*. Kemudian dataset dibagi menjadi dua bagian, yaitu data *training* dan data *testing*.
- c. Pengumpulan data twitter menggunakan bahasa python dengan *library* SNScrape.
- d. Data twitter diklasifikasikan menjadi 3 sentimen yaitu positif, netral dan negatif.
- e. Hasil pengujian akurasi menggunakan *Confusion Matrix* berdasarkan hasil klasifikasi sistem memperoleh nilai *accuracy* 70,2%, *precision* 67% dan *recall* 100%.
- f. Dari pengujian validasi data dengan menggunakan sampel data *training* berjumlah 3 *tweet*, dan data *testing* berjumlah 1 *tweet*, dapat disimpulkan bahwa total nilai probabilitas sentimen positif, negatif dan netral pada perhitungan sistem dengan perhitungan manual menghasilkan nilai yang sama. dengan nilai probabilitas positif tertinggi yang berarti hasil dari klasifikasi sampel data *testing* menghasilkan klasifikasi positif.

Pada penelitian ini masih memiliki kekurangan yang bisa ditambahkan untuk pengembangan penelitian berikutnya. Saran untuk pengembangan penelitian berikutnya dapat ditambahkan dataset lebih banyak agar model dapat mengenali kata, sehingga prediksi dapat semakin akurat.

DAFTAR PUSTAKA

- [1] A. Z. Malik, E. Utami, and S. Raharjo, “Analisis Sentiment Twitter Terhadap Capres Indonesia 2019 dengan Metode K-NN,” *J. Inf. Politek. Indones. Surakarta*, vol. 5, no. 2, pp. 1–7, 2019.
- [2] I. F. Kristalia, “Analisis kecelakaan lalu lintas di kota surakarta,” *Progr. Stud. Tek. Sipil, Fak. Tek. Univ. Muhammadiyah Surakarta*, 2019.
- [3] L. A. Mumtazia and A. Saputra, “City Branding Kota Surakarta: Peran Arsitektur Dalam Pembentukan Ekspresi Kota,” *Semin. Ilm. Arsit. II*, vol. 8686, pp. 363–371, 2021.
- [4] Dinas Pariwisata Kota Surakarta, “Jumlah wisatawan Nusantara / Domestik di Kota Surakarta,” 2022.
- [5] S. S. Salim and J. Mayary, “Analisis Sentimen Pengguna Twitter Terhadap Dompot Elektronik Dengan Metode Lexicon Based Dan K – Nearest Neighbor,” *J. Ilm. Inform. Komput.*, vol. 25, no. 1, pp. 1–17, 2020, doi: 10.35760/ik.2020.v25i1.2411.
- [6] D. Normawati and S. A. Prayogi, “Implementasi Naïve Bayes Classifier Dan Confusion Matrix Pada Analisis Sentimen Berbasis Teks Pada Twitter,” *J-SAKTI (Jurnal Sains Komput. dan Inform.)*, vol. 5, no. 2, pp. 697–711, 2021.
- [7] M. Php, K. Bekasi, R. Noviana, J. Margonda, R. No, and P. Cina, “Perancangan Web Sistem Analisis Sentimen Media Sosial Twitter Dengan Metode Valence Aware Dictionary And Sentimen Reasoner (Vader) Menggunakan PHP & MySQL pada Pemerintah Kota Bekasi,” *J. Ilm. Komputasi*, vol. 20, no. 1, pp. 1–14, 2021, doi: 10.32409/jikstik.20.1.369.
- [8] N. Fitriyah, B. Warsito, and D. A. I. Maruddani, “Analisis Sentimen Gojek Pada Media Sosial Twitter Dengan Klasifikasi Support Vector Machine (Svm,” *J. Gaussian*, vol. 9, no. 3, pp. 376–390, 2020, doi: 10.14710/j.gauss.v9i3.28932.
- [9] E. N. Hamdana, “Pengembangan Sistem Analisis Sentimen Berbasis Java Pada Data Twitter Terhadap Omnibus Law Menggunakan Algoritma Naïve Bayes dan K-Nearest Neighbor (K-NN),” *J. Inform. Polinema*, vol. 7, no. 2, pp. 79–84, 2021, doi: 10.33795/jip.v7i2.688.
- [10] A. Firdaus and W. I. Firdaus, “Text Mining Dan Pola Algoritma Dalam Penyelesaian Masalah Informasi : (Sebuah Ulasan),” *J. JUPITER*, vol. 13, no. 1, p. 66, 2021.
- [11] Y. T. Handika, S. Defit, and G. W. Nurcahyo, “Text Mining Dalam Membandingkan Metode Naïve Bayes Dengan C.45 Dalam Mengidentifikasi Berita Hoax Pada Media Sosial,” *Rang Tek. J.*, vol. 5, no. 1, pp. 116–123, 2022, doi: 10.31869/rtj.v5i1.2855.
- [12] A. M. F. Hulu and K. M. Lhaksana, “Analisis Sentimen Politik pada Twitter

- Menggunakan Metode Support Vector Machine (Studi Kasus : Pilpres 2019),” *e-Proceeding Eng.*, vol. 6, no. 2, pp. 2017–2020, 2019.
- [13] C. F. Hasri and D. Alita, “Penerapan Metode Naïve Bayes Classifier Dan Support Vector Machine Pada Analisis Sentimen Terhadap Dampak Virus Corona Di Twitter,” *J. Inform. dan Rekayasa Perangkat Lunak*, vol. 3, no. 2, pp. 145–160, 2022.
- [14] V. Rahmayanti, S. Basuki, and H. Hilman, “Klasifikasi sinopsis novel menggunakan metode naïve bayes classifier,” *J. Repos.*, vol. 1, no. 2, p. 125, 2019, doi: 10.22219/repositor.v1i2.799.