

# Klasifikasi Sentiment Ulasan Aplikasi Sausage Man Menggunakan VADER Lexicon dan Naïve Bayes Classifier

M Ikhsan Maulana, Elvia Budianita\*, Muhammad Fikry, Febi Yanto

Fakultas Sains dan Teknologi, Program Studi Teknik Informatika, Universitas Islam Negeri Sultan Syarif Kasim, Pekanbaru Indonesia

Email: <sup>1</sup>11750114753@students.uin-suska.ac.id, <sup>2,\*</sup>elvia.budianita@uin-suska.ac.id, <sup>3</sup>muhammad.fikry@uin-suska.ac.id, <sup>4</sup>febiyanto@uin-suska.ac.id

Email Penulis Korespondensi: elvia.budianita@uin-suska.ac.id

Submitted: 20/02/2023; Accepted: 26/03/2023; Published: 31/03/2023

**Abstrak**—*Game Battle royale* merupakan permainan yang memadukan elemen petualangan dan bertahan hidup dengan mode permainan orang terakhir yang bertahan. Salah satu *game Battle royale* yang diminati adalah *game Sausage Man*. Banyaknya keluhan seperti bug, cheater, dan FPS yang terus menurun membuat permainan menjadi terganggu. Solusinya yaitu pengembang harus memperbaiki dan meningkatkan keamanan *game* agar pengguna menjadi nyaman saat bermain *game* tersebut. Banyaknya pendapat atau ulasan dari pengguna mengenai permasalahan didalam *game*, akan dilakukan analisis sentimen pada data ulasan aplikasi Sausage Man di Google play store sebagai proses untuk menghasilkan pengkategorian pendapat melalui ulasan. Tujuan peneliti melakukan analisis sentimen untuk melihat pendapat positif, netral atau negatif dari pengguna *game Sausage Man*. Tahapan yang dilakukan dalam penelitian ini yaitu pengumpulan data menggunakan *web scraping*, pelabelan data, *text preprocessing*, pembobotan dokumen, klasifikasi, dan evaluasi. Hasil pelabelan data menggunakan VADER Lexicon didapatkan hasil sentimen positif sebanyak 1089 ulasan (36.3%), untuk sentimen netral sebanyak 912 ulasan (30.4%), dan untuk sentimen negatif sebanyak 999 ulasan (33.3%). Klasifikasi menggunakan *Naïve Bayes Classifier*. Evaluasi menggunakan *Confusion matrix* dengan menggunakan pembagian 90% data latih 10% data uji menghasilkan akurasi sebesar 75%, precision 79%, dan recall 75%. Untuk pembagian 80% data latih 20% data uji menghasilkan akurasi sebesar 73%, precision 76%, dan recall 73%. Kalimat positif lebih banyak ditemukan, namun akurasi masih dibawah 80%.

**Kata Kunci:** Game; Battle Royale; Sausage Man; Analisis Sentimen; Klasifikasi

**Abstract**—Battle Royale games are games that mix adventure and survival elements with last man standing game modes. One of the most popular battle royale games is the Sausage Man game. The number of complaints such as bugs, cheaters, and FPS which continues to decrease makes the game annoying. The solution is that developers must improve and improve game security so that users feel comfortable playing the game. There are many opinions or reviews from users regarding problems in the game, sentiment analysis will be carried out on the Sausage Man application review data on the Google play store as a process to produce categorization of opinions through reviews. The purpose of the researcher is to carry out a sentiment analysis to see positive, neutral or negative opinions from Sausage Man game users. The stages carried out in this study were data collection using web scraping, data labeling, text preprocessing, document weighting, classification, and evaluation. The results of data labeling using the VADER Lexicon obtained 1089 reviews (36.3%) for positive sentiment, 912 reviews for neutral sentiment (30.4%), and 999 reviews for negative sentiment (33.3%). Classification using the Naïve Bayes Classifier. Evaluation using the Confusion Matrix by dividing 90% training data and 10% test data produces an accuracy of 75%, 79% precision, and 75% recall. For the division of 80% training data 20% of the test data produces an accuracy of 73%, 76% precision and 73% recall. Positive sentences are found more often, but the accuracy is still below 80%.

**Keywords:** Game; Battle Royale; Sausage Man; Sentiment Analysis; Classification

## 1. PENDAHULUAN

*Game online* merupakan salah satu media hiburan yang diminati semua kalangan. *Game online* dapat dimainkan bersama siapa saja dan dimana saja ini menjadi daya tarik semua orang untuk bermain *game online*. *Game online* sangat beragam diantaranya *Multiplayer Online Battle Arena* (MOBA), *First Person Shooter* (FPS), dan *Battle royale*. Dari banyaknya *game online* yang ada saat ini, *game online* bergenre *Battle royale* yang memiliki cukup banyak peminat karena dalam satu permainan dimainkan oleh banyak tim. Selain itu, *game Battle royale* juga menantang banyak pemain, dimulai dengan mencari peralatan seadanya, kemudian mencari senjata, baju besi, dan mengeliminasi semua musuh sambil menghindari terjebak diluar zona aman yang mengecil, dengan pemenangnya orang terakhir bertahan didalam permainan. *Game Battle royale* yang diminati salah satunya adalah *game Sausage Man*.

*Game Sausage Man* dirilis 29 Juni 2021 di android merupakan *game Battle royale* yang dikembangkan oleh XD Entertainment. *Game Sausage Man* juga memiliki kesamaan dengan *PUBG* dan *Fornite* yang mengamati penyempitan zona setiap beberapa menit permainan berlangsung sehingga pemain saling bertemu dan saling baku tembak. Banyaknya pendapat atau ulasan dari pengguna mengenai masalah saat bermain *game*, akan baik apabila dilakukan analisis sentimen pada data ulasan aplikasi Sausage Man di Google Play Store sebagai proses untuk menghasilkan pengkategorian pendapat melalui ulasan yang diberikan pada suatu topik menjadi berlabel positif, negatif, atau netral. Dengan adanya analisis sentimen ini diharapkan bisa membantu pengembang *game* untuk mengetahui kekurangan dari aplikasi yang mereka buat dari ulasan pengguna aplikasi tersebut. Ulasan pengguna aplikasi bisa menjadi referensi bagi pengembang untuk mengembangkan aplikasi *game* yang mereka buat.

Penelitian terkait mengenai analisis sentimen adalah tentang data ulasan *PLN Mobile* [1]. Penelitian ini menggunakan 1000 data ulasan yang dilabeli menggunakan VADER *Lexicon* dengan berdasarkan ketidaksesuaian pengguna memberikan rating dengan ulasannya adalah untuk kelas positif 67% atau sebanyak 670 ulasan, 6% sentimen netral atau sebanyak 56 ulasan, dan 27% sentimen negatif atau sebanyak 274 ulasan dengan model klasifikasi menggunakan *Naive Bayes* dengan perbandingan data uji dan latih 90:10 menggunakan *Confusion matrix* mendapatkan nilai akurasi sebesar 70%. Penelitian terkait analisis sentimen berikutnya yaitu menggunakan data twitter tentang pemindahan ibukota indonesia menggunakan VADER dengan metode *Naive Bayes* dan *Support Vector Machine* [2]. Penelitian ini menggunakan VADER menghasilkan sentimen pada datanya serta algoritma yang digunakan terbukti menghasilkan nilai paling baik yakni mencapai 76,40% dengan tingkat AUG mencapai 0,771 sehingga masuk *fair classification*. Penelitian selanjutnya yaitu menggunakan data *twitter* terkait Non-Fugible Token [3]. Pelabelan menggunakan VADER menunjukkan bahwa sentimen pengguna *twitter* terkait NFT cenderung lebih banyak ditemukan sentimen negatif sebanyak 3840 *tweet*, sedangkan sentimen positif sebanyak 3220 *tweet* setelah di klasifikasikan menggunakan metode *Multinomial Naive Bayes* data mendapatkan akurasi sebesar 84% setelah di evaluasi performansi model yang didapatkan yaitu precision 84%, recall 81%, dan f1-score 83%.

Penelitian selanjutnya yaitu menggunakan data *twitter* yang menandai akun @Indihome selama pandemi Covid-19 [4]. Pengklasifikasian menggunakan *Naive Bayes* hasil pengujian diperoleh model dengan akurasi tertinggi sebesar 76.06% dari variasi pengujian 11 fold. Terakhir penelitian terkait analisis sentimen menggunakan data komentar dari YouTube [5]. Pelabelan menggunakan VADER dan *Text Blob* menunjukkan perbedaan komentar positif, negatif, dan netral dengan kedua metode pelabelan teks tersebut namun tidak terlalu jauh. Akurasi penerapan algoritma SVM terhadap dataset yang dilabeli dengan metode VADER mengungguli pelabelan komentar dengan metode *Text Blob*.

Berdasarkan penjelasan diatas peneliti akan melakukan penelitian analisis sentimen pada salah satu aplikasi game yaitu game Sausage Man. Tujuan melakukan analisis sentimen untuk melihat pendapat positif, netral, atau negatif dari pengguna *game* Sausage Man. VADER *Lexicon* digunakan untuk memberikan label pada ulasan atau komentar berdasarkan sentimennya yaitu positif, netral, dan negatif, sedangkan *Naive Bayes Classifier* dipilih karena banyak digunakan pada proses *text mining* dan dapat mempersingkat waktu proses pengklasifikasian analisis sentimen.

## 2. METODOLOGI PENELITIAN

### 2.1 Tahapan Penelitian

Tahapan penelitian bertujuan untuk menggambarkan alur penelitian yang akan dilakukan. Ada 6 tahapan yang akan dilakukan dalam penelitian ini yaitu pengumpulan data, pelabelan data, *text preprocessing*, pembobotan dokumen, klasifikasi, dan evaluasi. Gambar 1 menunjukkan alur dalam proses penelitian.



**Gambar 1.** Gambar Tahapan Penelitian

### 2.2 Pengumpulan Data

Data yang dikumpulkan yaitu teks berbahasa indonesia sebanyak 3000 data. Pengumpulan data menggunakan teknik scraping menggunakan *library* python yaitu *google play scraper* dan disimpan dalam bentuk format *xlsx*. Data yang sudah di *scraping* akan dinormalisasi manual kata yang tidak baku menjadi kata baku. Setelah dinormalisasi lanjut ke tahapan *translate* dari teks berbahasa indonesia ke teks berbahasa inggris menggunakan *library google trans*, karena pada penelitian ini akan menggunakan VADER *Lexicon* untuk pelabelan sentimen, karena *library* tersebut menggunakan teks berbahasa inggris.

### 2.3 Pelabelan Data

Data ulasan yang sudah di *translate* ke bahasa inggris akan dilanjutkan ke proses pelabelan data menggunakan VADER *Lexicon*. VADER (Valence Aware Dictionary and Sentiment Reasoner) yaitu pendekatan yang memungkinkan untuk mengklasifikasikan informasi teks ke dalam beberapa kategori sentimen yaitu negatif, positif, dan netral. Kemampuan klasifikasi dilakukan VADER dengan cara memberikan nilai pada setiap kata dalam teks. Penilaian tersebut berdasarkan angka yang ditemukan oleh Hutto, C.J and Gilbert melalui penelitiannya kepada manusia sebagai penilai. Dasar pemilihan VADER *Lexicon* ini karena nilai disetiap kata

dalam teks murni penilaian manusia bahkan VADER menangkap makna tersirat dari tanda baca yang terdapat dalam sebuah teks [6].

#### 2.4 Text Preprocessing

Informasi yang dikumpulkan adalah informasi yang tidak terstruktur oleh karena itu *text preprocessing* sangatlah penting menjadi data yang terstruktur. Langkah-langkah yang dilakukan pada *text preprocessing* yaitu *case folding*, *remove punctuation*, *tokenisasi*, *stopword removal*, dan *stemming*.

1. *Case folding* : yaitu menyeragamkan bentuk huruf-huruf menjadi lower case [7].
2. *Remove punctuation* : yaitu bertujuan untuk menghapus karakter tanda baca karena karakter tersebut tidak digunakan dalam analisis sentimen [8].
3. *Tokenizing* : merupakan tahap pemotongan kata berdasarkan tiap kata yang menyusunnya [9].
4. *Stopword removal* : digunakan untuk menghilangkan kata atau term yang tidak memiliki arti [10].
5. *Stemming* : digunakan untuk menghilangkan imbuhan yang terdapat pada masing-masing kata sehingga menjadi kata dasar, dan bertujuan untuk membersihkan suatu kata dengan pengejaan yang kurang tepat [11].

#### 2.4 Pembobotan Dokumen

*Term frakuensi inverse document frakuensi* adalah metode pembobotan yang menggabungkan metode TF dan IDF [12]. *Term frakuensi* merupakan jumlah kemunculan sebuah *term* pada sebuah dokumen, sedangkan *inverse document frakuensi* merupakan proses untuk mengukur seberapa penting kata yang ada didalam dokumen [13].

#### 2.5 Klasifikasi Naive Bayes Classifier

Klasifikasi adalah salah satu metode penambahan cacatan yang menetapkan label kelas ke sekumpulan kasus yang tidak di klasifikasikan [14]. Pada proses klasifikasi menggunakan *Naive Bayes Classifier*. *Naive Bayes Classifier* merupakan salah satu algoritma data mining yang penggunaannya serta pemrosesannya memiliki waktu yang cepat mudah di implementasikan dengan struktur yang cukup sederhana dan memiliki tingkat efektifitas yang tinggi [15]. *Naive Bayes Classifier* merupakan sebuah metode klasifikasi yang berakar pada Teorema Bayes, serta metode pengklasifikasian menggunakan metode probabilitas statistic yaitu memprediksi peluang berdasarkan pengalaman sebelumnya (Teorema Bayes) dengan ciri utamanya adalah asumsi yang sangat kuat (naif) akan ketergantungan dari masing-masing kondisi atau kejadian [16].

#### 2.6 Evaluasi Confusion Matrix

Evaluasi pada analisis sistem ini dilakukan dengan menghitung tingkat keakuratan suatu metode dalam menganalisis opini [17]. *Confusion matrix* memberikan informasi perbandingan hasil klasifikasi yang dilakukan oleh model klasifikasi dengan hasil klasifikasi sesungguhnya [18]. Nilai rata-rata *accuracy*, *precision*, *recall* didapatkan dari tabel *Confusion matrix* seperti tabel 1 berikut.

**Tabel 1.** Confusion Matrix

	True Yes	True No
Prediction Yes	TP	FP
Prediction No	FN	FN

### 3. HASIL DAN PEMBAHASAN

#### 3.1 Pengumpulan Data

Data yang digunakan adalah data yang sudah di *scraping* dari Google Play Store yang disimpan dalam bentuk format.xlsx sebanyak 3000 data dalam 5 kolom. Data yang sudah di *scraping* dinormalisasi manual kata yang tidak baku menjadi kata yang baku. Setelah dinormalisasi lanjut ke tahapan *translate* dari teks berbahasa indonesia ke teks berbahasa inggris menggunakan *library google trans*. Berikut ini isi data dapat dilihat pada gambar berikut.



4. *Stopword removal*, merupakan proses pembuangan kata yang tidak penting dari hasil token. Pada proses ini menggunakan modul NLTK (*Natural Language Toolkit*) yang sudah disediakan oleh python. Dapat dilihat pada tabel berikut.

**Tabel 5.** Hasil Stopword removal

Sebelum	Sesudah
[wow, this, works]	[wow, works]

5. *Stemming*, merupakan proses perubahan kata yang berimbuhan menjadi kata dasar. Modul yang digunakan yaitu *Porterstemmer* dan *WordNetLemmatizer*. Dapat dilihat pada tabel berikut.

**Tabel 6.** Hasil stemming

Sebelum	Sesudah
[wow, works]	[wow, work]

### 3.4 Pembobotan Dokumen

Tahap selanjutnya adalah pembobotan kata yang ada pada dokumen. Pada penelitian ini menggunakan library sklearn. Modul yang digunakan adalah *CountVectorizer* dan *Tf-Idf vectorizer*. Dapat dilihat pada gambar berikut.

```

from sklearn.feature_extraction.text import CountVectorizer
from sklearn.feature_extraction.text import TfidfVectorizer

vectorizer = CountVectorizer()
X = vectorizer.fit_transform(data_clean['Teks_bersih']).astype('U')

tf = TfidfVectorizer()
text_tf = tf.fit_transform(data_clean['Teks_bersih']).astype('U')
print(text_tf)

```

(0, 1823)	0.17278854074659555
(0, 1923)	0.26091364398934624
(0, 1618)	0.292223912689512
(0, 1394)	0.1057770024455269
(0, 1900)	0.26660989209053365
(0, 1220)	0.4819927858242323
(0, 1621)	0.18976939455528108
(0, 1305)	0.18031202467476423
(0, 207)	0.18407314645409364
(0, 675)	0.6144126556011551
(0, 1921)	0.14891723140078506
(1, 1097)	0.49459631970396106
(1, 712)	0.8297804393707031
(1, 831)	0.25853220877302685
(2, 1126)	0.23993629624522383
(2, 848)	0.24700711544738777
(2, 1851)	0.3453306366800132
(2, 505)	0.4130113514564398

**Gambar 4.** Pembobotan dokumen

Dari gambar diatas dapat dilihat hasil pembobotan dokumen menggunakan library sklearn dengan menggunakan modul *Countvectorizer* dan *Tf-idf vectorizer*. *Counvectorizer* berfungsi untuk mengubah teks menjadi token, sedangkan *Tf-idf vectorizer* berfungsi untuk menghitung bobot frakuensi tiap kata pada dokumen. Dapat dilihat pada gambar diatas hasil pembobotan dokumen menggunakan modul *Countvectorizer* dan *Tf-idf vectorizer* hasil perhitungannya dimulai dari index 0.

### 3.5 Klasifikasi Naive Bayes Classifier

Setelah melakukan pembobotan dokumen menggunakan TF-IDF tahap berikutnya adalah melakukan klasifikasi. Klasifikasi yang digunakan yaitu *Naive Bayes Classifier*. Pada tahap klasifikasi akan dilakukan perbandingan data latih dan data uji menggunakan perbandingan 90:10 dan 80:20 di split data. Dapat dilihat pada gambar berikut.

```

[ ] import collections, numpy
from sklearn.model_selection import train_test_split
X_train, X_test, y_train, y_test = train_test_split(text_tf, data_clean['Sentiment'], test_size=0.1, random_state=13)
print("Jumlah Data Uji:", X_test.shape)
print("Jumlah Data Latih:", X_train.shape)

Jumlah Data Uji: (300, 2024)
Jumlah Data Latih: (2700, 2024)

[ ] from sklearn.naive_bayes import MultinomialNB
clasfc= MultinomialNB()
cl= clasfc.fit(X_train,y_train)
Y_pred= cl.predict(X_test)
Y_pred

```

**Gambar 5.** Perbandingan 90:10

```
[ ] import collections, numpy
from sklearn.model_selection import train_test_split
X_train, X_test, y_train, y_test = train_test_split(text_tf, data_clean['Sentiment'], test_size=0.2, random_state=13)
print("Jumlah Data Uji:", X_test.shape)
print("Jumlah Data Latih:", X_train.shape)

Jumlah Data Uji: (600, 2024)
Jumlah Data Latih: (2400, 2024)

[ ] from sklearn.naive_bayes import MultinomialNB
clasfc= MultinomialNB()
cl= clasfc.fit(X_train,y_train)
Y_pred= cl.predict(X_test)
Y_pred
```

**Gambar 6.** Perbandingan 80:20

Gambar diatas dapat dilihat proses klasifikasi menggunakan *Naive Bayes Classifier*. Untuk proses klasifikasi ini menggunakan model *MultinomialNB* dari *library sklearn* dengan memanggil masing-masing modelnya. Dari masing-masing model tersebut akan diterapkan ke data latih yang sudah terbentuk. Kemudian akan memprediksi sentimen dari ulasan pengguna aplikasi Sausage Man yang akan menjadi patokan data uji.

**3.6 Evaluasi Confusion Matrix**

Setelah melakukan klasifikasi menggunakan *Naive Bayes Classifier* tahap terakhir yaitu evaluasi menggunakan *Confusion matrix* untuk mengetahui akurasi, presisi, recall. Hasil pengujian dapat dilihat pada tabel dibawah ini.

**Tabel 7.** Hasil pengujian

Jumlah Kelas	Total Data	Data Latih	Data Uji	Akurasi	Recall	Precision
3	3000	90%	10%	75%	75%	79%
3	3000	80%	20%	73%	73%	76%

**Tabel 8.** Hasil Confusion matrix perbandingan 90:10

Prediksi Aktual	Negatif	Netral	Positif
Negatif	82	4	26
Netral	9	53	22
Positif	12	1	91

**Tabel 9.** Hasil Confusion matrix perbandingan 80:20

Prediksi Aktual	Negatif	Netral	Positif
Negatif	129	7	61
Netral	25	130	35
Positif	28	4	181

Dari pengujian yang sudah dilakukan yang sudah dilakukan dengan menghitung akurasi, *recall*, dan *precision* maka analisis dapat disimpulkan.

1. Pada tabel 7 dapat dilihat dari nilai akurasi menggunakan 90% data latih 10% data uji memiliki akurasi tertinggi 75%. Sedangkan pada pengujian 80% data latih 20% data uji memiliki akurasi terendah 73%.
2. Pada tabel 8 dapat dilihat dari *Confusion matrix* menggunakan perbandingan data 90% data latih 10% data uji di dapatkan 91 kelas terprediksi positif, 1 kelas positif terprediksi netral, 12 kelas positif terprediksi negatif, 22 kelas netral terprediksi kelas positif, 53 kelas yang terprediksi netral, 9 kelas netral terprediksi negatif, 26 kelas negatif terprediksi positif, 4 kelas negatif terprediksi netral, dan 82 kelas negatif terprediksi negatif.
3. Pada tabel 9 dapat dilihat dari *Confusion matrix* menggunakan perbandingan data 80% data latih 20% data uji dapat dilihat 181 kelas terprediksi positif, 4 kelas positif terprediksi netral, 28 kelas positif terprediksi negatif, 35 kelas terprediksi sebagai kelas positif, 130 kelas terprediksi kelas netral, 25 kelas netral terprediksi negatif, 61 kelas positif terprediksi negatif, 7 kelas negatif terprediksi netral, dan 129 kelas negatif terprediksi sebagai kelas negatif.
4. Untuk nilai akurasi tertinggi analisis sentimen dari pengujian yang sudah dilakukan terdapat pada kelas positif
5. Berdasarkan tabel 8 dan 9 dari *Confusion matrix* pengenalan kalimat negatif lebih banyak dikenali sebagai kalimat positif.
6. Hasil pengujian yang dilakukan menggunakan 2 perbandingan menghasilkan nilai akurasi, *recall*, dan *precision*. Dapat dilihat pada gambar berikut.

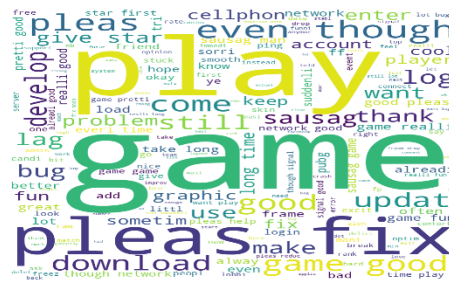
	precision	recall	f1-score	support
Negative	0.80	0.73	0.76	112
Netral	0.91	0.63	0.75	84
Positive	0.65	0.88	0.75	104
accuracy			0.75	300
macro avg	0.79	0.75	0.75	300
weighted avg	0.78	0.75	0.75	300

**Gambar 7.** Hasil akurasi, recall, dan precision 90:10

	precision	recall	f1-score	support
Negative	0.71	0.65	0.68	197
Netral	0.92	0.68	0.79	190
Positive	0.65	0.85	0.74	213
accuracy			0.73	600
macro avg	0.76	0.73	0.74	600
weighted avg	0.76	0.73	0.73	600

**Gambar 8.** Hasil akurasi, recall, dan precision 80:20

Gambar diatas menunjukkan hasil pengujian untuk metode klasifikasi *Naive Bayes Classifier* mendapatkan akurasi sebesar 75% menggunakan perbandingan 90% data latih 10% data uji, dan 73% menggunakan perbandingan 80% data latih 20% data uji. Berikut ini Hasil word cloud positif, netral, dan negatif pada analisis sentimen ini dapat dilihat pada gambar berikut.



**Gambar 9.** Word cloud sentimen positif



**Gambar 10.** Word cloud sentimen netral



**Gambar 11.** Word cloud sentimen negatif

#### 4. KESIMPULAN

Berdasarkan penelitian dan pembahasan yang sudah dilakukan yakni Klasifikasi Sentiment Ulasan Aplikasi Sausage Man Menggunakan VADER *Lexicon* dan *Naive Bayes Classifier* dapat disimpulkan bahwa hasil pelabelan menggunakan VADER *Lexicon* di dapatkan hasil sentimen positif 36.3% atau sebanyak 1089 ulasan, untuk sentimen netral 30.4% atau sebanyak 912 ulasan, dan untuk sentimen negatif 33.3% atau sebanyak 999 ulasan. Hasil pengujian menggunakan *Naive Bayes Classifier* menggunakan perbandingan 90% data latih 10% data uji menghasilkan akurasi sebesar 75%. Untuk perbandingan 80% data latih 20% data uji menghasilkan akurasi sebesar 73%. Selanjutnya dapat disimpulkan hasil dari *Confusion matrix* bahwa pengenalan kalimat negatif lebih

banyak dikenali sebagai kalimat positif. Berdasarkan hasil *Word cloud* kata good, still, pleas, dan play merupakan kata yang sering muncul selain kata game.

## REFERENCES

- [1] Y. Asri, W. N. Suliyanti, D. Kuswardani, dan M. Fajri, “Pelabelan Otomatis Lexicon Vader dan Klasifikasi Naive Bayes dalam menganalisis sentimen data ulasan PLN Mobile,” vol. 15, no. 2, hal. 264–275, 2022.
- [2] J. Watori, R. Aryanti, dan A. Junaidi, “Penggunaan Algoritma Klasifikasi Terhadap Analisa Sentimen Pindahan Ibukota Dengan Pelabelan Otomatis,” J. Inform., vol. 7, no. 1, hal. 85–90, 2020.
- [3] R. Y. Lesmana dan R. Andarsyah, “Model Klasifikasi Multinomial Naive Bayes Untuk Analisis Sentiment Terkait Non-Fungible Token,” J. Tek. Inform., vol. 14, no. 3, hal. 135–139, 2022.
- [4] S. Pamungkas dan J. B. B. Darmawan, “Klasifikasi Sentiment Tweet Pelanggan IndiHome Selama Pandemi Covid-19 Menggunakan Algoritma Multinomial Naive Bayes,” SNESTIK, hal. 339–344, 2022.
- [5] M. G. T. Akbar dan D. B. Srisulistiowati, “Analisa Sentimen Efektifitas Vaksin terhadap Varian COVID 19 Omicron Berbasis Leksikon,” J. Inf. Inf. Secur., vol. 2, no. 2, hal. 251–258, 2021.
- [6] E. A. Marwa dan A. B. Kristanto, “Analisis Sentimen Pengungkapan Informasi Manajemen : Text Mining Berbasis Metode VADER,” Own. Ris. J. Akunt., vol. 6, hal. 2973–2984, 2022.
- [7] W. Parasati, F. A. Bachtiar, dan N. Y. Setiawan, “Analisis Sentimen Berbasis Aspek pada Ulasan Pelanggan Restoran Bakso President Malang dengan Metode Naive Bayes Classifier,” J. Pengemb. Teknol. Inf. dan Ilmu Komput., vol. 4, no. 4, hal. 1090–1099, 2020.
- [8] N. S. Wardani, A. Prahutama, dan P. Kartikasari, “Analisis Sentimen Pindahan Ibu Kota Negara dengan Klasifikasi Naive Bayes untuk Model Bernoulli dan Multinomial,” J. Gaussian, vol. 9, hal. 237–246, 2020.
- [9] G. Sanjaya dan K. M. Lhaksana, “Analisis Sentimen Komentar YouTube tentang Terpilihnya Menteri Kabinet Indonesia Maju Menggunakan Lexicon Based,” e-Proceeding Eng., vol. 7, no. 3, hal. 9698–9710, 2020.
- [10] L. Ardiani, H. Sujaini, dan Tursina, “Implementasi Sentiment Analysis Tanggapan Masyarakat Terhadap Pembangunan di Kota Pontianak,” J. Sist. dan Teknol. Inf., vol. 8, no. 2, hal. 183–190, 2020, doi: 10.26418/justin.v8i2.36776.
- [11] N. Herlinawati, Y. Yuliani, S. Faizah, W. Gata, dan Samudi, “Analisis Sentimen Zoom Cloud Meetings di Play Store menggunakan Naive Bayes dan Support Vector Machine,” J. Comput. Eng. Syst. Sci., vol. 5, no. 2, hal. 293–298, 2020.
- [12] A. Deolika, Kusri, dan E. T. Luthfi, “Analisis Pembobotan Kata pada Klasifikasi Text Mining,” J. Teknol. Inf., vol. 3, no. 2, hal. 179–184, 2019.
- [13] M. R. Fahlevvi, “Analisis Sentimen Terhadap Ulasan Aplikasi Pejabat Pengelola Informasi dan Dokumentasi Kementerian Dalam Negeri Republik Indonesia di Google Playstore Menggunakan Metode Support Vector Machine,” J. Teknol. dan Komun. Pemerintah., vol. 4, no. 1, hal. 1–13, 2022.
- [14] R. A. R. Wiguna dan A. I. Rifai, “Analisis Text Clustering Masyarakat Di Twitter Mengenai Omnibus Law Menggunakan Orange Data Mining,” J. Inf. Syst. Informatics, vol. 3, no. 1, hal. 1–12, 2021.
- [15] E. Fitri, Y. Yuliani, S. Rosyida, dan W. Gata, “Analisis Sentimen Terhadap Aplikasi Ruangguru Menggunakan Algoritma Naive Bayes , Random Forest Dan Support Vector Machine,” TRANSFORMATIKA, vol. 18, no. 1, hal. 71–80, 2020.
- [16] D. N. Chandra, G. Indrawan, dan I. N. Sukajaya, “Klasifikasi Berita Lokal Radar Malang Menggunakan Metode Naive Bayes dengan Fitur N-Gram,” J. Ilmu Komput. Indones., vol. 4, no. 2, 2019.
- [17] O. S. D. Silaen, Herlawati, dan Rasim, “Analisis Sentimen Mengenai Gangguan Bipolar Pada Twitter Menggunakan Algoritma Naive Bayes,” J. Komputika (Komputasi dan Inform., vol. 6, no. 2, hal. 63–73, 2022.
- [18] T. T. Widowati, “Analisis Sentimen Twitter terhadap Tokoh Republik dengan Algoritma Naive Bayes dan Support Vector Machine,” J. SIMETRIS, vol. 11, no. 2, 2020.