JITET (Jurnal Informatika dan Teknik Elektro Terapan)
Vol. 13 No. 1, pISSN: 2303-0577 eISSN: 2830-7062 http://dx.doi.org/10.23960/jitet.v13i1.5687

ANALISIS SENTIMEN PENGGUNA APLIKASI MYBLUEBIRD DENGAN ALGORITMA NAÏVE BAYES DI PLAYSTORE

Deni Prasetia[1*], Nining Rahaningsih[2], Raditya Danar Dana[3], Cep Lukman Rohmat[4]

[1,2,3,4]STMIK IKMI Cirebon; Jl. Perjuangan No. 10 B Majasem Kec. Kesambi Kota Cirebon; telp.(0231)490480

Received: 13 Desember 2024
Accepted: 14 Januari 2025
Published: 20 Januari 2025

Keywords:
Analisis Sentimen;
MyBlueBird;
Naïve Bayes;
Layanan Transportasi.

Corespondent Email:
deniprasetia063@gmail.com

Abstrak. Transportasi berbasis aplikasi digital, termasuk *MyBlueBird*, menjadi kebutuhan penting di era modern. Namun, kualitas layanan aplikasi ini sering menjadi sorotan, terutama dari ulasan pengguna yang mengindikasikan adanya keluhan. Penelitian ini bertujuan untuk menganalisis sentimen ulasan pengguna aplikasi *MyBlueBird* guna mengevaluasi kualitas layanannya. Analisis dilakukan menggunakan algoritma *Naïve Bayes* untuk mengklasifikasikan ulasan menjadi sentimen positif, negatif, dan netral. Penelitian menggunakan data sekunder dari *Google Playstore* dengan total 1.000 ulasan yang diproses melalui tahapan pengumpulan data, *preprocessing* teks, dan evaluasi model menggunakan akurasi, presisi, dan recall. Hasil menunjukkan algoritma *Naïve Bayes* mampu mengklasifikasikan sentimen dengan akurasi 92%. Sebagian besar ulasan bersifat positif, menunjukkan tingkat kepuasan tinggi, meskipun terdapat keluhan terkait keterlambatan pengemudi dan kualitas kendaraan. Analisis ini menjadi alat strategis untuk meningkatkan layanan berbasis data pengguna dan dapat diterapkan pada sektor lain.

Abstract. *Digital app-based transportation, including MyBlueBird, has become an essential necessity in the modern era. However, the service quality of these apps is often in the spotlight, especially from user reviews that indicate complaints. This research aims to analyze the sentiment of MyBlueBird app user reviews to evaluate its service quality. The analysis was conducted using the Naïve Bayes algorithm to classify reviews into positive, negative and neutral sentiments. The research used secondary data from Google Playstore with a total of 1,000 reviews processed through the stages of data collection, text preprocessing, and model evaluation using accuracy, precision, and recall. The results show that the Naïve Bayes algorithm is able to classify sentiment with 92% accuracy. Most reviews were positive, indicating a high level of satisfaction, although there were complaints related to driver tardiness and vehicle quality. This analysis is a strategic tool for improving services based on user data and can be applied to other sectors.*

1. PENDAHULUAN

Layanan transportasi berbasis aplikasi digital sudah jadi satu di antara solusi penting untuk masyarakat modern dalam memenuhi kebutuhan mobilitas harian. Aplikasi ini menawarkan kenyamanan, efisiensi, dan aksesibilitas yang tinggi, menjadikannya sangat diminati oleh pengguna di berbagai wilayah, khususnya di kota-kota besar di Indonesia. Satu di antara aplikasi yang menonjol dalam kategori ini adalah *MyBlueBird*, yang menyediakan layanan taksi online dengan berbagai fitur unggulan. Keberadaan aplikasi ini tidak hanya mempermudah proses pemesanan transportasi, tetapi juga memberikan pengalaman baru dalam layanan transportasi berbasis digital.

Namun, meskipun menjadi salah satu pilihan populer, aplikasi *MyBlueBird* juga menghadapi berbagai tantangan, terutama dalam menjaga dan meningkatkan kualitas layanan yang dirasakan oleh pengguna. Ulasan pengguna di *platform* seperti *Google Playstore* sering kali menjadi sumber data penting yang mencerminkan persepsi mereka terhadap layanan. Keluhan yang sering muncul meliputi keterlambatan pengemudi, kualitas kendaraan yang kurang memadai, dan interaksi yang tidak konsisten antara pengemudi dan pelanggan. Keluhan semacam ini, bila tidak mengelolanya dengan optimal, bisa membawa dampak negatif pada kepuasan pelanggan serta citra aplikasi secara keseluruhan.

Untuk mengatasi tantangan ini, diperlukan pendekatan yang sistematis untuk menganalisis persepsi pengguna secara objektif. Satu di antara metode yang kerap dipergunakan di penelitian ialah analisis sentimen, yaitu teknik yang mengklasifikasikan opini atau ulasan ke dalam golongan positif, negatif, atau netral. Analisis ini memanfaatkan data yang diambil dari ulasan pengguna sebagai dasar untuk mengevaluasi layanan dan mengidentifikasi area yang memerlukan perbaikan.

Sejumlah penelitian sebelumnya telah menunjukkan efektivitas analisis sentimen dalam memahami persepsi pengguna di berbagai domain. Penelitian sebelumnya melakukan analisis sentimen atas opini masyarakat mengenai pinjaman online mempergunakan algoritma *Naïve Bayes* dan berhasil mencapai akurasi yang cukup tinggi [1]. Penelitian lain mengungkap bahwa algoritma ini efektif dalam mengidentifikasi sentimen negatif terkait aplikasi media sosial dengan akurasi mencapai 95% [2]. Hal serupa juga ditunjukkan oleh yang mempergunakan algoritma *Naïve Bayes* dalam melakukan analisis mengenai ulasan aplikasi pemerintah digital, menghasilkan wawasan yang berguna bagi pengembang layanan [3].

Meskipun penelitian terdahulu memperlihatkan keberhasilan algoritma *Naïve Bayes* pada beragam konteks, penerapannya pada analisis sentimen ulasan aplikasi transportasi seperti *MyBlueBird* masih relatif jarang. Penelitian ini mempunyai tujuan dalam menutup gap tersebut melalui melakukan analisis terhadap sentimen ulasan pengguna *MyBlueBird* guna mengukur kualitas layanan yang dirasakan. Dengan mempergunakan algoritma *Naïve Bayes*, penelitian ini diharapkan dapat memberi informasi mendalam terkait kekuatan dan kelemahan layanan yang ada, serta membantu pengembangan aplikasi untuk menyusun strategi perbaikan layanan sesuai dengan data yang lebih terukur dan akurat.

2. TINJAUAN PUSTAKA

Proses *text mining* merupakan bagian dari analisis teks yang dilaksanakan secara otomatis menggunakan komputer guna menemukan informasi bermutu dari kumpulan teks yang terdapat di suatu dokumen [4].

Analisis sentimen yakni jenis penelitian berbasis teks yang memiliki tujuan guna menemukan serta mengerti emosi atau opini yang ada di teks. Jenis penelitian ini berkonsentrasi pada pembagian pernyataan berdasarkan label sentimen, seperti ulasan positif atau negatif [5].

Aplikasi *MyBluebird* dirancang untuk menghadapi gempuran era digital. Dengannya, orang dapat dengan mudah memesan dan menggunakan layanan *Bluebird* yang sudah ada. Menurut survei yang dilakukan oleh Polling Institute, orang Indonesia paling sering menggunakan taksi online pada bulan September 2022 [6].

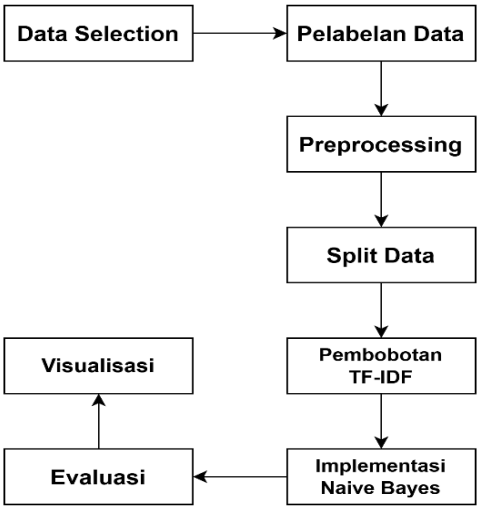
Pada penelitian ini, *Google Colab* dipergunakan untuk menjadi alat dalam melaksanakan analisis data serta pelatihan model klasifikas. *Platform* cloudnya memberi kesempatan pengguna menulis serta mengeksekusi kode *Python* secara langsung di browser serta tidak perlu konfigurasi tambahan [7].

Pada *text mining*, algoritma *naive Bayes* mempergunakan pengalaman masa lalu guna memperkirakan apa yang hendak terjadi di masa mendatang, yang dikenal sebagai teorema Bayes [8].

3. METODE PENELITIAN

Penelitian ini mempergunakan metode kuantitatif melalui desain eksperimental, menganalisis data sekunder dari ulasan pengguna aplikasi *MyBluebird* di *Google Playstore*. Data dikumpulkan dengan teknik scraping mempergunakan *Google Colaboratory* dan paket *Google Play Scraper*. Metode *data science* seperti *text mining, NLP,*

serta algoritma *Naive Bayes* dipergunakan dalam mengelompokkan sentimen menjadi positif, negatif, dan netral, guna mengukur tingkat kepuasan pengguna terhadap aplikasi *MyBluebird*. Adapun tahapan penelitian ini bisa tersaji dalam Gambar 1.



Gambar 1. Tahapan Penelitian

3. 1. Data Selection

Mengumpulkan data ulasan pengguna aplikasi *MyBlueBird* dari *platform Playstore* dengan menggunakan teknik *scraping*. Ulasan yang relevan diseleksi untuk membangun dataset penelitian.

3. 2. Pelabelan Data

Pelabelan data yakni proses memberi label atau kategori di data sehingga dapat digunakan untuk analisis, pelatihan machine learning, atau tujuan lainnya. Secara umum, label adalah informasi yang relevan dengan data tersebut, seperti kategori, kelas, atau nilai tertentu. Tahapan ini bisa dilaksanakan secara manual atau secara otomatis mempergunakan algoritma [3].

3. 4. *Preprocessing*

preprocessing adalah serangkaian tahapan pengolahan awal yang diterapkan pada teks mentah sebelum data tersebut dianalisis lebih lanjut. Dalam Teks *preprocesssing* , terdapat beberapa tahap yang biasa diterapkan, seperti *case folding*, penghapusan *stopword*, *tokenisasi*, dan *stemming* [9].

1. *Case Folding*

Case folding ialah satu di antara tahap dalam preprocessing data teks pada pengolahan bahasa alami (*Natural Language Processing*/*NLP*).

2. Penghapusan *Stopword*

Stopword yakni tahapan menghapus kata yang tidak terlalu dibutuhkan dan tidak mempunyai makna apa pun [10].

3. Tokenisasi

Tokenisasi adalah proses mengubah sekelompok kalimat - kalimat menjadi sebuah kata , diikuti dengan penghapusan simbol - simbol dan tanda baca tertentu , yang menghasilkan terciptanya kelompok kata yang unik [11].

4. *Stemming*

Stemming yakni tahapan menerjemahkan kata ke bentuk dasarnya (*Natural Language Processing*/*NLP)*. Proses ini dilakukan dengan menciptakan imbuhan seperti awalan, akhiran, atau sisipan dari kata tertentu.

3. 5. Split Data

Split Data merupakan tahapan pembagian dataset ke dalam dua komponen utama yang bertujuan untuk memastikan pengujian model pembelajaran mesin dilaksanakan secara objektif.

3. 6. Pembobotan TF-IDF

TF-IDF adalah teknik *Natural Language Processing* (NLP) yang menilai setiap kata di dokumen sesuai dengan sesering apa kata tersebut muncul serta seunik apa kata tersebut dibandingkan dengan kata lain dalam koleksi dokumen.

3. 7. Klasifikasi Naïve Bayes

Naive Bayes adalah algoritma pembelajaran mesin yang kerap dipergunakan dalam mengklasifikasikan teks. Algoritma ini memperlakukan fitur secara kondisional independen satu sama lain. Pada *naive bayes*, probabilitas kelas dokumen dihitung serta diposisikan di kelas dengan probabilitas paling tinggi.

4. HASIL DAN PEMBAHASAN

4.1. Teknik pengumpulan Data

Penelitian ini mengumpulkan 1.000 ulasan pengguna aplikasi *MyBluebird* tahun 2024 dari

Google Playstore menggunakan *library google-play-scraper*. Proses *scraping* dilakukan di platform *Google Colaboratory* dengan *Python*,hanya mengambil ulasan yang berisi konten relevan. Berikut hasil pengumpulan dataset.

	content
118	pelayanan nya luarbiasa
6	Kenapa fasilitas yang sudah ada diambil? Kami ...
414	Ok Puas Pelayanan nya
105	pelayanannya selalu terdepan. terima kasih
7	Parahh burung biru.apalagi Medan. Cas nya lamb...

Gambar 2. Hasil Pengumpulan Dataset

4.2. Pelabelan Data

Langkah berikutnya adalah pelabelan data. Data dilabelkan dengan ahli bahasa Indonesia dan menggunakan *Microsoft Excel*. Kategori pelabelan adalah Positif, Negatif, dan Netral.

4.3. Preprocessing

Setelah data selesai dilabeli, langkah berikutnya adalah melakukan *preprocessing*. *Preprocessing* merupakan serangkaian tahap pengolahan awal yang diterapkan pada teks mentah sebelum data digunakan untuk analisis lebih lanjut. Beberapa proses umum yang digunakan dalam teks *preprocessing* termasuk *case folding*, penghilangan *stopword*, tokenisasi, serta *stemming* [9].

4.3.1 Case Folding

Dalam analisis sentimen *Case folding* yakni tahapan mengubah semua huruf dalam teks jadi huruf kecil (*lowercase*). Dalam tahapan ini contohnya karakter 'A'-'Z' yang ada di data ganti menjadi karakter 'a'-'z' [12]. Berikut adalah proses case folding.

```
import re
def  clean_text(df, text_field, new_text_field_name):
    data[new_text_field_name] = data[text_field].str.lower()
    data[new_text_field_name] = data[new_text_field_name].apply(lambda elem: re.sub(r"(@[A-Za-z0-9]+)|([^
    # remove numbers
    data[new_text_field_name] = data[new_text_field_name].apply(lambda elem: re.sub(r"\d+", "", elem))
    return data
```

```
data['text_clean'] = data['content'].str.lower()
data['text_clean']
data_clean = clean_text(data, 'content', 'text_clean')
data_clean.head(10)
```

Gambar 3. Proses *Case Folding*

Pemrosesan dataset dimulai dengan tahap *Case Folding*, seperti yang ditunjukkan pada gambar di atas. Perintah pertama adalah mengimpor *library import re*, modul bawaan *Python*. Hasil dari *Case Folding* dapa dicermati melalui gambar.

	content	Label	text_clean
0	pelayanan nya luar biasa	Positif	pelayanan nya luar biasa
1	Kenapa fasilitas yang sudah ada diambil? Kami ...	Negatif	kenapa fasilitas yang sudah ada diambil kami y...
2	Ok Puas Pelayanan nya	Positif	ok puas pelayanan nya
3	pelayanannya selalu terdepan. terima kasih	Positif	pelayanannya selalu terdepan terima kasih
4	Parahh burung biru.apalagi Medan. Cas nya lamb...	Negatif	parahh burung biruapalagi medan cas nya lambat...
5	map sering tidak sesuai aplikasi	Negatif	map sering tidak sesuai aplikasi

Gambar 4. Hasil *case folding*

Dalam gambar 4. dapat dilihat perolehan dari *Case Folding* yang dimana huruf kapital pada kolom content “Kenapa fasilitas..” berubah menjadi huruf kecil pada kolom text_clean “kenapa fasilitas..” dapat dilihat juga dalam kolom seterusnya.

4.3.2 Penghapusan *Stopword*

Dalam proses pra-pemrosesan teks, istilah "penghapusan *stopword*" dipergunakan dalam menghapus kata-kata umum, atau yang dianggap tidak memiliki makna.

```
import nltk.corpus
nltk.download('stopwords')
from nltk.corpus import stopwords
stop = stopwords.words('indonesian')
data_clean['text_StopWord'] = data_clean['text_clean'].apply(lambda x: ' '.join([word for word in x.split() if word not in (stop)]))
data_clean.head(50)
```

Gambar 5. Penghapusan *stopword*

Pada gambar tersebut, langkah pertama yang dilaksanakan ialah *import nltk.corpus*, yaitu pernyataan dalam bahasa pemrograman *Python* yang berfungsi untuk mengimpor modul *nltk.corpus* dari *NLTK*. Hasil dari pengahapusan *stopword* bisa dicermati melalui gambar.

text_clean	text_StopWord
pelayanan nya luar biasa	pelayanan nya
kenapa fasilitas yang sudah ada diambil kami yang dinas atas keperluan kantor tidak bisa minta struk taksi pengemudi bilang sudah ditarik perusahaan saya gak mungkin komplain ke mereka namun gegara ini kami jadi adu argumen dulu sama bagian keuangan agar mau menerima struk yang dari email tolong agar printernya dipasang lagi terima kasih	fasilitas diambil dinas keperluan kantor struk taksi pengemudi bilang ditarik perusahaan gak komplain gegara adu argumen keuangan menerima struk email tolong printernya dipasang terima kasih
ok puas pelayanan nya	ok puas pelayanan nya
pelayanannya selalu terdepan terima kasih	pelayanannya terdepan terima kasih

Gambar 6. Hasil *Stopword*

Pada gambar diatas, dapat dilihat bahwa Output (hasil) dari pemrosesan penghapusan Stopword pada dataset. Dapat dilihat juga hasil dari pemrosesannya bahwa pada kolom *text_stopword* terdapat kata yang hilang dari suatu kalimat pada baris pertama terdapat perbedaan kalimat pada setiap kolom.

4.3.3. Tokenisasi

Dalam tahap ini melakukan tokenisasi, tokenisasi yaitu proses membagi kalimat jadi sejumlah bagian kata terpisah. proses tokenisasi bisa dicermati melalui gambar 7. Serta hasil dari tokenisasi ada dalam gambar 8.

```
import nltk
nltk.download('punkt')
from nltk.tokenize import sent_tokenize, word_tokenize
data_clean['text_tokens'] = data_clean['text_StopWord'].apply(lambda x: word_tokenize(x))
data_clean.head()
```

Gambar 7. Proses tokenisasi

text_StopWord	text_tokens
pelayanan nya	[pelayanan, nya]
fasilitas diambil dinas keperluan kantor struk...	[fasilitas, diambil, dinas, keperluan, kantor,...
ok puas pelayanan nya	[ok, puas, pelayanan, nya]
pelayanannya terdepan terima kasih	[pelayanannya, terdepan, terima, kasih]
parahh burung biruapalagi medan cas nya lambat...	[parahh, burung, biruapalagi, medan, cas, nya,...

Gambar 8. Hasil tokenisasi

Pada gambar 8. dapat dilihat hasil dari Tokenisasi pada dataset. Dapat dilihat juga hasil dari pemrosesannya terdapat pada kolom *text_tokens* terdapat kalimat yang dipisahkan secara perkata/dipisahkan secara unit – unit kecil.

4.3.4. *Stemming*

Proses penghilangan imbuhan pada sebuah kata untuk mendapatkan bentuk dasar atau akar kata dikenal sebagai *stemming*. Berikut adalah proses *stemming* yang ditunjukan di gambar 9.

```
from Sastrawi.Stemmer.StemmerFactory import StemmerFactory
factory = StemmerFactory()
stemmer = factory.create_stemmer()
```

```
#----------------STEMMING ----------------
from Sastrawi.Stemmer.StemmerFactory import StemmerFactory
#import swifter


# create stemmer
factory = StemmerFactory()
stemmer = factory.create_stemmer()

# stemmed
def stemmed_wrapper(term):
    return stemmer.stem(term)

term_dict = {}
hitung=0

for document in data_clean['text_tokens']:
    for term in document:
        if term not in term_dict:
            term_dict[term] = ' '

print(len(term_dict))
print("------------------------")
for term in term_dict:
    term_dict[term] = stemmed_wrapper(term)
    hitung+=1
    print(hitung,":",term,":" ,term_dict[term])

print(term_dict)
print("------------------------")

# apply stemmed term to dataframe
def get_stemmed_term(document):
    return [term_dict[term] for term in document]
```

Gambar 9. Proses *stemming*

Pemrosesan *stemming* dilakukan dengan menginstal *library pip_install_sastrawi* dan kemudian mengimport *library sastrawi*. Perolehan dari tahapan *stemming* bisa dicermati melalui gambar.

text_tokens	text_steamindo
[pelayanan, nya]	layan nya
[fasilitas, diambil, dinas, keperluan, kantor,...	fasilitas ambil dinas perlu kantor struk taksi...
[ok, puas, pelayanan, nya]	ok puas layan nya
[pelayanannya, terdepan, terima, kasih]	layan depan terima kasih
[parahh, burung, biruapalagi, medan, cas, nya,...	parahh burung biruapalagi medan cas nya lambat...

Gambar 10. Hasil *stemming*

hasil pemrosesan *stemming* pada dataset dapat dilihat pada gambar di atas. Hasil pemrosesan tersebut terdiri dari kalimat pada baris pertama, yang diubah menjadi stemming pada kolom *text_stemmindo* dengan kalimat (layan nya), dan juga dapat dilihat pada baris berikutnya, di mana penataan dan penguraian kata dari satu kata menjadi kata-kata dasar.

4.4. Split Data

Dalam pengembangan model pembelajaran mesin, pembagian data adalah proses membagi dataset menjadi beberapa bagian untuk tujuan pelatihan (*training*) dan pengujian (*testing*). Berikut adalah proses split data.

```
from sklearn.model_selection import train_test_split
X_train, X_test, y_train, y_test = train_test_split(data_clean['content'], data_clean['Label'],
                                                    test_size = 0.20,
                                                    random_state = 0)
```

Gambar 11. Split Data

Dari gambar diatas pembagian data dilakukan dengan 80% data latih, dan data uji 20%.

4.5. Pembobotan *TF-IDF*

Setelah dilakukan split data, selanjutnya proses pembobotan *TF-IDF* yang berguna untuk menentukan sepenting apa setiap kata pada sebuah dokumen dibandingkan dengan

semua kumpulan dokumen (*corpus*). Dalam gambar 11. Ialah tahap pembobotan *TF-IDF*.

```
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)
```

Gambar 11. Pembobotan *TF-IDF*

Pada gambar 11. *TF-IDF* dilakukan dengan *library sklearn.feature_extraction.text* dari modul *TfidfVectorizer*, yang mempunyai fungsi mengganti teks menjadi representasi *TF-IDF*.

4.6. Implementasi *Naïve Bayes*

Dalam penelitian ini digunakan algoritma *Naïve Bayes*. Dapat dilihat juga kode dari pemrosesan algoritma *Naïve Bayes* dalam gambar 12.

```
from sklearn.naive_bayes import MultinomialNB

nb = MultinomialNB()
nb.fit(tfidf_train, y_train)
```

```
MultinomialNB
MultinomialNB()
```

Gambar 12. Algoritma *Naïve Bayes*

4.7. Evaluasi

Untuk evaluasi model, metode *confusion matrix* diterapkan dalam sejumlah skenario pembagian data latih serta data uji dengan rasio 80:20. Perolehan dari pengujian memperlihatkan nilai akurasi, presisi, *recall*, serta *f1-skor* model yang diuji. Perolehan evaluasi model di tiap skenario pembagian data latih serta data uji kemudian memberi gambaran mengenai bagaimana model berfungsi pada beragam keadaan.

```
MultinomialNB Accuracy: 0.92
MultinomialNB Precision: 0.9275739644970414
MultinomialNB Recall: 0.92
MultinomialNB f1_score: 0.9017501285217054
confusion_matrix:
 [[ 30   0   7]
 [  0   1   9]
 [  0   0 153]]
==================================================

              precision    recall  f1-score   support

     Negatif       1.00      0.81      0.90        37
      Netral       1.00      0.10      0.18        10
     Positif       0.91      1.00      0.95       153

    accuracy                           0.92       200
   macro avg       0.97      0.64      0.68       200
weighted avg       0.93      0.92      0.90       200
```

Gambar 13. Hasil evaluasi model

Dalam gambar 13. ialah hasil dari penerapapan metode *Naïve Bayes* dalam aplikasi *MyBlueBird* ditunjukkan bahwasanya model *Multinomial Naive Bayes* berhasil mencapai akurasi sebesar 92% dalam mengklasifikasikan data uji. Model ini memiliki nilai rerata presisi 93%, *recall* 92%, serta *F1-score* 90%. Dalam proses evaluasi, untuk sentimen negatif, diperoleh *precision* sebesar 100%, *recall* 81%, dan *F1-score* 90%. Untuk sentimen netral, *precision* tercatat sebesar 100%, *recall* 10%, dan *F1-score* 18%. Di samping itu, bagi sentimen positif, model mencatatkan *precision* sejumlah 91%, *recall* 100%, serta *F1-score* 95%.

4.8 Visualisasi

Pada penelitian ini, visualisasi akan disajikan menggunakan *wordcloud*. Guna melihat gambaran atau informasi umum terkait data ulasan pengguna aplikasi *MyBlueBird* dalam platfrom *google playstore*.

Gambar 14. *Wordcloud*

Dari gambar 14. Bisa dicermati seperti kata “aplikasi” “bluebird” “bagus” “layan” merupakan kata yang kerap muncul dan dipergunakan dalam ulasan aplikasi *MyBluebird* di penelitian ini. Ukuran kata pada *wordcloud* yang lebih besar menunjukkan frekuensi kemunculan kata tersebut yang lebih tinggi.

5. KESIMPULAN

a. Metode *Naive Bayes* berhasil menggolongkan ulasan pengguna aplikasi *MyBluebird* ke dalam kategori sentimen netral, positif, serta negatif.. Dengan pendekatan ini, penelitian dapat mengungkap persepsi pengguna secara sistematis dan objektif, memberikan gambaran mengenai tingkat kepuasan serta

kritik terhadap layanan yang diberikan oleh aplikasi tersebut.

b. Dengan tingkat akurasi sebesar 92%, algoritma *Naive Bayes* sangat baik untuk mengklasifikasikan sentimen pengguna. Hasil ini memperlihatkan bahwasanya algoritma ini efektif untuk mengolah data teks untuk tujuan analisis sentimen, khususnya untuk aplikasi transportasi berbasis digital.

c. Tujuan penelitian ini adalah untuk menemukan dan mengukur persepsi pengguna tentang aplikasi *MyBluebird*. Hasil analisis menunjukkan bahwa sebagian besar ulasan pengguna bersifat positif, menunjukkan bahwa pengguna sangat puas dengan layanan. Namun, beberapa hal perlu diperhatikan, seperti keluhan tentang keterlambatan pengemudi dan kondisi kendaraan.

UCAPAN TERIMA KASIH

Penulis mengucapkan terima kasih sebesar-sebesarnya kepada seluruh pihak yang sudah membantu serta memberi dukungannya selama proses penelitian ini.

DAFTAR PUSTAKA

[1] M. I. Ghozali, W. H. Sugiharto, and A. F. Iskandar, "Analisis Sentimen Pinjaman Online Di Media Sosial Twitter Menggunakan Metode Naive Bayes," *KLIK Kaji. Ilm. Inform. dan Komput.*, vol. 3, no. 6, pp. 1340–1348, 2023, doi: 10.30865/klik.v3i6.936.

[2] T. I. Suari and D. Gustian, "SENTIMEN ANALISIS TERHADAP PENGGUNA APLIKASI TWITTER PADA GOOGLE PLAYSTORE MENGGUNAKAN METODE NAIVE BAYES," *SISMATIK (Seminar Nas. Sist. Inf. dan Manaj. Inform.*, 2023.

[3] D. Wijaya, R. Adi Saputra, and F. Irwiensyah, "Analisis Sentimen Ulasan Aplikasi Samsat Digital Nasional Pada Google Playstore Menggunakan Algoritma Naïve Bayes," *Kaji. Ilm. Inform. dan Komput.*, vol. 4, no. 1, pp. 2369–2380, 2024, doi: 10.30865/klik.v4i4.1738.

[4] A. Rhamadanti, A. Rifa'i, F. Dikananda, and K. Anam, "Analisis Sentimen Pada Ulasan Access By Kereta Api Indonesia Dengan K-Nearest Neighbor," *J. Inform. dan Tek. Elektro Terap.*, vol. 12, no. 1, 2024, doi: 10.23960/jitet.v12i1.3961.

[5] S. Arofah *et al.*, "Analisis Sentimen Pemakaian Sistem Absensi Berbasis Web," vol. 8, no. 3, pp. 2619–2625, 2024.

[6] Novirianto, "Analisis Sentimen Berbasis Aspek Pada UlasanAplikasi Mybluebird Dengan Implementasi N-GramDan Algoritma Logistic Regression," 2023, [Online]. Available: https://repository.uinjkt.ac.id/dspace/bitstream/123456789/76656/1/IQBAL FARIZ NOVIRIANTO-FST.pdf

[7] Wartumi, R. Kurniawan, and A. Y. Wijaya, "Analisis Data Sentimen Ulasan Pengguna Aplikasi Shopee di Google Play Store dengan Klasifikasi Algoritma Naïve Bayes," *J. Inform. dan Rekayasa Perangkat Lunak*, vol. 6, no. 1, pp. 164–170, 2024.

[8] D. Duei Putri, G. F. Nama, and W. E. Sulistiono, "Analisis Sentimen Kinerja Dewan Perwakilan Rakyat (DPR) Pada Twitter Menggunakan Metode Naive Bayes Classifier," *J. Inform. dan Tek. Elektro Terap.*, vol. 10, no. 1, pp. 34–40, 2022, doi: 10.23960/jitet.v10i1.2262.

[9] N. R. Siahaan *et al.*, "ANALISIS SENTIMEN ULASAN APLIKASI MEDIA SOSIAL WHATSAPP MENGGUNAKAN METODE NAIVE BAYES CLASSIFIER," *J. Ilm. Betrik*, vol. 14, no. 02, pp. 343–354, 2023, [Online]. Available: https://ejournal.sttpagaralam.ac.id/index.php/betrik/index

[10] M. F. El Firdaus, N. Nurfaizah, and S. Sarmini, "Analisis Sentimen Tokopedia Pada Ulasan di Google Playstore Menggunakan Algoritma Naïve Bayes Classifier dan K-Nearest Neighbor," *JURIKOM (Jurnal Ris. Komputer)*, vol. 9, no. 5, p. 1329, 2022, doi: 10.30865/jurikom.v9i5.4774.

[11] A. Hendra and F. Fitriyani, "Analisis Sentimen Review Halodoc Menggunakan Nai¨ve Bayes Classifier," *JISKA (Jurnal Inform. Sunan Kalijaga)*, vol. 6, no. 2, pp. 78–89, 2021, doi: 10.14421/jiska.2021.6.2.78-89.

[12] A. Suryana, A. Irma Purnamasari, and I. Ali, "Mengoptimalkan Kepuasan Pengguna: Analisis Sentimen Review Aplikasi Grab Di Indonesia," *JATI (Jurnal Mhs. Tek. Inform.*, vol. 8, no. 3, pp. 3396–3404, 2024, doi: 10.36040/jati.v8i3.9688.