PENERAPAN METODE *NAIVE BAYES* PADA ANALISIS SENTIMEN APLIKASI MCDONALDS DI *GOOGLE PLAY STORE*

Dhyya' Rohannisa Fathwa Daud, Bambang Irawan, Agus Bahtiar
Program Studi Teknik Informatika STMIK IKMI CIREBON
Jl. Perjuangan No. 10B Majasem Kec. Kesambi Kota Cirebon Tlp. 0231-490480-490481
dhyyarohannisafathwa@gmail.com

ABSTRAK

Penelitian ini menitikberatkan pada implementasi metode Naive Bayes dalam menganalisis sentimen terhadap aplikasi McDonald's di Google Play Store dengan menggunakan ulasan pengguna sebagai sumber data. Proses analisis melibatkan langkah-langkah preprocessing, seperti tokenisasi, eliminasi stopwords, dan vektorisasi menggunakan metode Term Frequency-Inverse Document Frequency (TF-IDF). Dengan model Naive Bayes, penelitian mencapai tingkat akurasi 88%, menunjukkan keahlian model dalam mengklasifikasikan ulasan ke dalam kategori sentimen positif, negatif, atau netral dengan tingkat presisi yang tinggi. Hasil temuan menyoroti keunggulan model dalam mengidentifikasi ulasan ber sentimen negatif, dengan tingkat kepresisian 90%, recall 95%, dan nilai F1-score 93%. Namun, terdapat penurunan kinerja dalam mengklasifikasikan sentimen positif, dengan tingkat kepresisian 77%, recall 61%, dan nilai F1-score 68%. Temuan penelitian ini memberikan wawasan mendalam mengenai persepsi pengguna terhadap aplikasi McDonald's, menjadi dasar bagi pengembang untuk meningkatkan kualitas layanan berdasarkan umpan balik pengguna yang lebih akurat. Dengan akurasi tinggi, penelitian ini berpotensi mendukung pengambilan keputusan berbasis umpan balik pengguna, memungkinkan perbaikan yang lebih cermat dan efektif dalam pengembangan aplikasi berbasis sentimen. Keseluruhan, penelitian ini memberikan pemahaman mendalam mengenai respons pengguna terhadap aplikasi kuliner di era kemajuan digital, dan metodologi yang digunakan dapat diaplikasikan dalam penelitian analisis sentimen pada berbagai sektor industri di era digital, di mana aplikasi telah menjadi platform utama dalam sektor penjualan, khususnya di bidang makanan.

***Kata kunci** : Analisis Sentimen, Nive Bayes, Mcdonald's, Google Play Store, Ulasan Pengguna.*

1. PENDAHULUAN

Di era digital saat ini, sektor penjualan, termasuk makanan, pakaian, dan penginapan, semakin mengadopsi aplikasi sebagai platform utama, banyak pengguna yang kini bergantung pada aplikasi untuk berkomunikasi, memberikan ulasan, dan berbagi pengalaman mereka di *Google Play Store*. *Google* memiliki sebuah layanan yang dikenal dengan nama *Play Store* yang menyediakan konten - konten digital seperti *game*, aplikasi, film, musik, dan buku dengan kategori yang beragam. Fitur *rating* dan ulasan di *Play Store* memungkinkan pengguna produk di *platform* tersebut memberikan pendapat serta penilaian terhadap produk yang telah mereka gunakan [1]. *Rating* mencerminkan tingkat kepuasan yang dirasakan oleh pengguna setelah menggunakan produk atau jasa. [2].

Dalam konteks ini, analisis sentimen ulasan pengguna menjadi krusial. Melalui analisis ini, kita dapat memahami dengan lebih baik bagaimana pengguna merespon dan berinteraksi dengan aplikasi Mcdonal'dsSalah satu teknik yang menjanjikan untuk melakukan evaluasi sentimen adalah Metode *Naive Bayes*, yang telah terbukti berhasil dalam berbagai situasi analisis sentimen. Oleh karena itu, tujuan dari penelitian ini adalah mengimplementasikan Metode *Naive Bayes* untuk menganalisis sentimen dari ulasan pengguna Aplikasi McDonald's di Google Play Store. Salah satu strategi yang menjanjikan untuk mengevaluasi sentimen adalah menggunakan Metode *Naive Bayes*, yang telah terbukti efektif dalam berbagai konteks analisis sentimen. Dengan demikian, tujuan penelitian ini adalah menerapkan Metode *Naive Bayes* untuk menilai sentimen dari ulasan pengguna Aplikasi *McDonald's* di *Google Play Store*. Fokus utamanya adalah untuk mendapatkan pemahaman yang lebih mendalam mengenai sentimen pengguna dan memberikan wawasan yang dapat digunakan dalam pengembangan aplikasi yang lebih unggul. Pertumbuhan pesat terus terjadi dalam industri makanan, terutama di sektor makanan cepat saji yang terus berusaha memberikan pelayanan dan harga terbaik.

Pada penelitian sebelumnya, dilakukan pengujian bahwa metode *KNN* memiliki nilai yang lebih tinggi dibandingkan dengan metode yang lain dengan akurasi sebesar 71.49%, *precision* sebesar 72.86% dan *recall* sebesar 76.94% [3]. 2 penelitian lainnya, dilakukan analisis mengenai ulasan tiktok menggunakan metode *support vector machine*, *logostic regression* dan *nive bayes* dengan hasil nilai akurasi 82%, Precision 82%, *Recall* 81% dan F1 *score* 81% pada *svm*, nilai akurasi 79%, *Precision* 81%, *Recall* 77% dan F1 *score* 78% *nive bayes*, dan pada *logistic regression* dengan hasil nilai akurasi 84%, *precision* 83%, *recall* 82%, F1 *score* 83% [4] Dan penelitian lainnya Hasil dari artikel ini terlihat bahwa algoritma *Naïve Bayes Classifier* mampu melakukan prediksi kalimat positif dan negatif dengan nilai 62,5% positif dan 37,5% negatif dari 40 data

pendapat orangtua dan nilai akurasi sebesar 65% berdasarkan 100 berita positif dan 100 berita *negative* [5].

Tujuan utama dari penelitian ini adalah untuk mengoptimalkan *Nive Bayes* dalam melakukan analisis sentimen terhadap ulasan pengguna Aplikasi *Mcdonalds* di *Google Play Store*. Penelitian ini memiliki *relevansi* penting dalam menyediakan wawasan mengenai sentimen pengguna, yang dapat menjadi landasan bagi pengembang aplikasi untuk meningkatkan pengalaman pengguna dan merespons umpan balik dengan lebih efektif. Selain itu, penelitian ini juga memberikan kontribusi pada bidang Informatika dengan mengembangkan teknik analisis sentimen yang lebih unggul dan praktis, yang dapat diterapkan dalam berbagai konteks.

Penilaian sentimen merupakan suatu teknik untuk menggali informasi dari teks dengan maksud untuk mengidentifikasi apakah sentimennya cenderung positif atau negatif [6]. Oleh karena itu, akan digunakan metode *empiris* yang mencakup pengumpulan dataset ulasan pengguna Aplikasi *Mcdonald's* dari Google *Play Store*, *preprocessing* data, penerapan Metode *Nive Bayes* penanganan ketidakseimbangan kelas sentimen, evaluasi hasil, dan analisis akhir. Pendekatan ini akan memungkinkan kami untuk menemukan solusi-solusi yang relevan dalam upaya meningkatkan analisis sentimen Aplikasi *Mcdonald's* dengan menggunakan Metode *Nive Bayes.* Dengan hasil dari penelitian ini, diharapkan akan diperoleh pemahaman yang lebih mendalam mengenai sentimen pengguna terhadap Aplikasi *Mcdonalds*. Pemahaman tersebut dapat menjadi panduan bagi pengembang untuk meningkatkan aplikasi dan merespon kebutuhan pengguna dengan lebih efektif. Selain itu, penelitian ini juga memiliki potensi untuk memberikan dampak lebih besar dalam pengembangan teknik analisis sentimen di bidang Informatika dan dapat menjadi dasar untuk penelitian lebih lanjut mengenai penerapan metode analisis sentimen dalam konteks aplikasi *mobile*.

2. TINJAUAN PUSTAKA

Kalsifikasi mengenai permrosesan pengolahan datanya, dapat di deskripsiksan seperti berikut :

a. *Colecting Data*
   Pencarian permasalah yang akan di angkat, dan melakukan scrapping pengambilan data ulasan *MCDonald's.*
b. *Preprocessing*
   Melakukan pemrosesan pembersihan data yang tidak perlu
c. *Transformation*
   Melakukan pembuatan data berupa text dengan cara : *tokenizing, stopword dan stimming*
d. *Data Mining*
   Melkukan testing dan training
e. *Evaluation*
   Melakukan *confusion* matriks atau tolak ukur dengan menggunakan algoritma matriks nive bayes.

2.1. Analisis Data

Kata analisis diadaptasi dari bahasa Inggris "analysis" yang secara etimologis dari bahasa Yunani kuno "Analusis", terdiri dari dua suku kata yaitu "ana" yang artinya kembali dan "luein" yang artinya melepas atau mengurai. Bila digabungkan memiliki arti menguraikan kembali. Secara umum analisis adalah aktivitas yang terdiri dari serangkaian kegiatan seperti : mengurai, membedakan, dan memilah sesuatu untuk dikelompokkan kembali menurut kriteria tertentu kemudian dicari keitannya lalu ditafsirkan maknanya [7].

Dalam pengolahan datanya dilakukan beberapa tahapan yaitu:

a. Scrapping Data Import Data
   Scraping data atau web scraping adalah proses ekstraksi informasi dari situs web pada tahap ini, dilakukan penginstallan library yang akan digunakan serta penmilihan data yang akan digunakan. Pengambilan datanya dilakukan melalui situs web, dianalisis dan disimpan dalam format seperti spreadsheet atau database. Scraping data dilakukan menggunakan Bahasa python dan Pustaka seperti BeautifulSoup atau Selenium.
   Berikut cara kerja *scrapping* data :
   - Mengakses situs web dan mengunduh *HTML* nya
   - Mengurai *HTML* untuk mengekstrak informasi yang diinginkan sepeti teks, gambar, atau tabel.
   - Menyimpan hasil ekstraksi kedalam struktur data yang sesuai. Contoh : list, dictionary
b. Import data
   Metode pengambilan data dan pemindahan data dalam bentuk csv
c. *Pelabelan*
   Pememberikan identifikasi atau tag terhadap suatu objek, data, entitas. Pada tahap ini, dilakukan perhitungan polaritas ulasan yang telah diambil dari hasil tahap sebelumnya.
d. Preprocessing Data
   Proses mengubah data mentah kedalam bentuk yang lebih mudah dipahami
e. *Processing* Data
   Proses pengumpulan data mentah dan menrjemahkannya kedalam informasi yang dapat digunakan.
f. Pemodelan
   Proses membuat model menggunakan matriks
g. Evaluasi Hasil.
   Tahapan yang bertujuan untuk menilai sejauh mana performa algoritma klasifikasi yang digunakan.

2.2. Teknik Pengumpulan Data

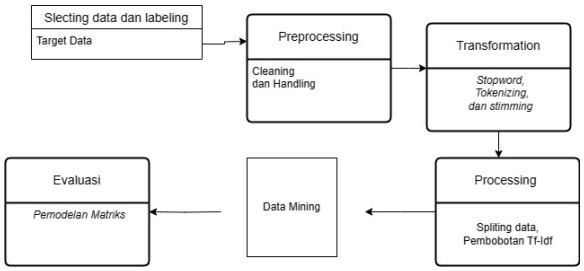
Penelitian ini akan melakukan pengumpulan data dengan memanfaatkan teknik web scraping. Proses

web scraping akan dimulai dengan mengidentifikasi URL halaman Google Play Store yang mengandung ulasan dan peringkat Aplikasi Mcdonalds. Setelahnya, alat atau library web scraping seperti BeautifulSoup atau Selenium akan digunakan untuk mengekstraksi informasi yang relevan dari halaman tersebut. Data yang diambil meliputi ulasan pengguna, peringkat, tanggal ulasan, dan komentar terkait. Setelah proses pengumpulan data selesai, langkah selanjutnya adalah melakukan pra-pemrosesan data, seperti membersihkan teks dari karakter khusus, menghapus ulasan yang duplikat, dan mengonversi peringkat menjadi kategori sentimen (positif, negatif, atau netral). Data yang telah diproses ini akan digunakan untuk melatih model analisis sentimen menggunakan metode Nive Bayes.

Naive Bayes adalah metode yang berguna dalam melakukan klasifikasi dengan menerapkan aturan-aturan Bayes menggunakan perhitungan peluang. Pada saat melakukan klasifikasi, pengelompokan dilakukan dengan memanfaatkan nilai probabilitas maksimal [8].

3. METODE PENELITIAN

Dalam Era Perkembangan Teknologi Informasi dan pengolahan data, penerapan metode *Nive Bayes* pada analisis sentiment aplikasi Mcdonald's di google play store. Penggunaan metode *Nive Bayes* sebagai bagian dari proses KDD (Knowledge Discovery in Databases) menjadi pilihan utama untuk mengawali wawasan yang berharga dari kumpulan data yang kompleks dan besar. KDD merupakan suatu pendekatan yang melibatkan ekstraksi pengetahuan yang berkualitas dari data secara sistematis.



Gambar 1. Metode Penelitian

Pada Gambar di atas merukan gambaran tahap demi tahap pemrosesan atau penerapan metode naïve bayes dilakukan sesuai dengan KDD.

4. HASIL DAN PEMBAHASAN

Pada tahap ini, dijelaskan bahwa hasil penelitian dilakukan dengan cara scrapping data. Dalam pemrosesan dan percobaannya, penelitian ini menggunakan aplikasi google collab serta menggunakan Bahasa pemrograman python. Hasil penelitian ini, akan dibagi menjadi beberapa Langkah yang sesuai dengan pendekatan metode yang digunakan.

4.1. Data

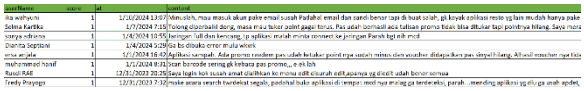
Data adalah Kumpulan fakta, angka, atau informasi yang disusun atau diorganisir sehingga memiliki arti. Dalam konteks umum, data mengacu pada sejumlah fakta atau detail yang dapat diukur, dihitung, atau diobservasi. Pada penelitian ini pengambilan data dilakukan dengan cara scrapping data dengan parameter pencarian "mcdonald's".

```
[1] #Referensi: https://www.linkedin.com/pulse/how-scrape-google-play-reviews-4-simple-steps-using-python-kundi/
    #download library google-play-scraper
    !pip install google-play-scraper

Collecting google-play-scraper
  Downloading google_play_scraper-1.2.4-py3-none-any.whl (28 kB)
Installing collected packages: google-play-scraper
Successfully installed google-play-scraper-1.2.4
```

Gambar 2. Script Data

Pada gambar ini, dilakukan *scrapping data* dengan melakukan penginstalan *library*, mencantumkan *link html* data set yang ada di google playstore, untuk pengambilan data serta penentuan jumlah dataset yang akan di ambil, di simpan dan di *upload* kedalam *google drive.*



Gambar 3. Gambar dataset

Gambar 3 merupakan tampilan dari dataset yang telah diambil, didalamnya terdapat 4 atribut yaitu *username, score, at dan conten*. Dataset yang diambil sebanyak 1000 dataset yang belum diselesaikan atau diolah, dan disimpan dalam bentuk excel.

4.2. *Labeling*

Labeling dilakukan dengan cara memberikan identifikasi atau tag terhadap suatu objek, data, entitas. Pada tahap ini, dilakukan perhitungan polaritas ulasan yang telah diambil dari hasil tahap sebelumnya. Tujuannya adalah menghasilkan dua kategori, yakni label positif dan negatif. Label netral dengan nilai 0 tidak akan mengalami proses tambahan [9].

```
def pelabelan(score):
  if score < 3:
    return 'Negatif'
  elif score == 4 :
    return 'Positif'
  elif score == 5 :
    return 'Positif'
my_df['Label'] = my_df ['score'].apply(pelabelan)
my_df.head(50)
```

81	257 poin, tukar 210 poin utk 5 ayam tapi nangu...	1	Negatif
2	Aplikasi SAMPAH. Senangnya update melulu, tp g...	1	Negatif
84	Apk ga jelas pdahal cuma mau lihat menu sama p...	2	Negatif
119	Aplikasi gak guna, ui apaan itu jelek sekali. ...	1	Negatif
8	Harusnya ada refund point klo misalnya ada gag...	1	Negatif
20	Untuk lihat promo, kenapa harus cek lokasi seg...	1	Negatif
10	Perbaiki sistem redeem poin, kadang poin langs...	1	Negatif
37	Penukaran poin di MCD Pekalongan, poin terpoto...	1	Negatif
68	Padahal udah dapat promo, tapi tidak bisa di k...	1	Negatif

Gambar 4. Labeling

Pada gambar 4 ini, menunjukan tampilan mengenai gambar hasil dari *scrapping* data tersebut. Pada tahap ini, dilakukannya pelabelan data untuk melatih model pembelajaran mesin, memvalidasi model, atau untuk tujuan analisis.

4.3. *Preprocessing*

Pada tahapan ini, dilakukan serangkaian Tindakan atau Langkah pada data mentah (*rsw* data) sebelum diolah dan di analisis. Tujannya untuk membersihkan, merapihkan, dan menyiapkan data sesuai kebutuhan dan kondisi untuk tahap selanjutnya dalam proses analisis dan pemodelannya. Berikut Langkah – langkahnya :

4.3.1. Cleaning Data

```
import pandas as pd
pd.set_option('display.max_columns', None)
my_df = pd.read_csv('/content/scrapped_data.csv')
my_df.head(50)
```

	content	score	Label
0	Aplikasi yg sangat membantuuu	5	Positif
1	Minuslah, mau masuk akun pake email susah Pada...	1	Negatif
2	Aplikasi samapah,paling gak berguna	1	Negatif
3	Aplikasi nyampah JANGAN INSTAL	1	Negatif
4	Tolong diperbaiki dong, masa mau tuker point g...	1	Negatif

Gambar 5. Gambar info

Pada gambar 5 ini, dilakuannya impor pustaka Pandas menggunakan alias 'pd', langkah selanjutnya adalah mengonfigurasi opsi tampilan Pandas dengan `pd.set_option('display.max_columns', None)`, yang bertujuan untuk menampilkan semua kolom saat mencetak DataFrame, terutama berguna ketika berurusan dengan banyak kolom. Kemudian, data dari file CSV di '/content/scrapped_data.csv' dibaca dan disimpan dalam variabel `my_df` menggunakan fungsi `read_csv` dari Pandas. Akhirnya, lima puluh baris pertama dari DataFrame `my_df` ditampilkan dengan `my_df.head(50)`, memberikan gambaran awal tentang struktur dan konten dataset yang baru dibaca.

```
my_df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 1000 entries, 0 to 999
Data columns (total 3 columns):
 #   Column   Non-Null Count  Dtype 
---  ------   --------------  ----- 
 0   content  1000 non-null   object
 1   score    1000 non-null   int64 
 2   Label    934 non-null    object
dtypes: int64(1), object(2)
memory usage: 23.6+ KB
```

Gambar 6. Gambar info

Objek data ini adalah *data frame* dari *library pandas* dengan 1000 baris dan 3 kolom. Kolom pertama, "*content*", berisi objek teks; kolom kedua, "*score*", berisi nilai *integer*; dan kolom ketiga, "*Label*", memiliki 934 nilai *non-null*, menunjukkan adanya beberapa nilai yang kosong. *Tipe* data kolom meliputi objek untuk teks dan int64 untuk nilai *integer*. Dengan estimasi penggunaan memori sekitar 23.6 KB, informasi ini memberikan gambaran lengkap tentang struktur dan karakteristik data dalam *dataframe*.

Output yang diberikan, dengan nilai `*False*` untuk kolom "content" dan "score", serta `*True*` untuk kolom "*Label*", mungkin mencerminkan hasil penerapan kondisi *boolean* pada suatu *data frame* atau *Series* di *pandas*. Pengecekan ini dapat memberikan informasi tentang keberadaan nilai-nilai yang kosong (*NaN* atau *None*) dalam kolom "*Label*", sedangkan kolom lainnya sepertinya tidak memiliki nilai yang hilang. Dengan menggunakan *output* ini sebagai mask, kita dapat mengidentifikasi dan menangani nilai-nilai yang kosong sesuai kebutuhan analisis atau pemrosesan data.

4.3.2. Handling Missing value-Ignore tuple.

```
#mencari jumlah baris data yang bernilai null
#terdapat kolom label memiliki nilai kosong
my_df.isnull().sum()
```

```
content     0
score       0
Label      66
dtype: int64
```

Gambar 7. info pencarian baris yang bernilai null

Keluaran yang diberikan menunjukkan nilai `0` pada kolom "*content*" dan "*score*", sementara nilai pada kolom *"Label"* mencapai `66`. Hal ini mungkin hasil dari penerapan suatu fungsi atau kondisi tertentu pada suatu DataFrame atau *Series* menggunakan pandas. Pada kolom "*content*" dan "*score*", nilai `0` mengindikasikan bahwa tidak ada elemen atau baris dengan nilai nol pada keduanya. Sementara itu, jumlah `66` pada kolom *"Label"* kemungkinan mencerminkan jumlah elemen atau baris yang memenuhi suatu kondisi khusus, seperti nilai yang bukan nol pada kolom tersebut. Untuk pemahaman lebih mendalam, perlu dilakukan analisis lebih lanjut mengenai bagaimana fungsi atau kondisi tersebut diterapkan pada data yang bersangkutan.

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

	content	score	Label	text_clean
0	Aplikasi yg sangat membantuuu	5	Positif	aplikasi yg sangat membantuuu
1	Minuslah, mau masuk akun pake email susah Pada...	1	Negatif	minuslah mau masuk akun pake email susah padah...
2	Aplikasi samapah,paling gak berguna	1	Negatif	aplikasi samapahpaling gak berguna
3	Aplikasi nyampah JANGAN INSTAL	1	Negatif	aplikasi nyampah jangan instal

Gambar 8. Hasil folding

Tabel yang diberikan merupakan dataset yang berisi beberapa kolom, *yaitu 'content,' 'score,' 'Label,' dan 'text_clean.' Kolom 'content'* berisi ulasan atau komentar pengguna terkait suatu layanan atau aplikasi. Kolom 'score' memuat nilai numerik yang mencerminkan skor atau penilaian yang diberikan

untuk setiap ulasan, dengan rentang skor dari 1 (Negatif) hingga 5 (Positif). Kolom *'Label'* memberikan klasifikasi sentimen, menggunakan label 'Negatif' untuk skor 1 dan 'Positif' untuk skor 5. Sementara itu, kolom 'text_clean' tampaknya berisi teks yang telah melalui proses pra-pemrosesan, seperti penghapusan karakter khusus dan pengubahan huruf menjadi huruf kecil. Data ini dapat digunakan untuk analisis sentimen, di mana ulasan pengguna dinilai dan dikategorikan berdasarkan sentimennya, dengan fokus pada aspek-aspek positif atau negatif dari pengalaman pengguna.

4.4. *Transformation*

Transformasi teks atau pembentukan atribut mencakup langkah-langkah untuk memperoleh representasi dokumen yang diperlukan. Pada tahap transformasi ini, penulis melakukan ekstraksi fitur dengan menggunakan metode *TF-IDF*[10].

4.4.1. Stopword Remova

Pada langkah ini, terjadi proses penyaringan untuk memisahkan kata-kata yang memiliki makna atau penting, sambil menghapus kata-kata imbuhan dan kata sambung[11].

```
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)
```

```
[nltk_data] Downloading package stopwords to /root/nltk_data...
[nltk_data]   Unzipping corpora/stopwords.zip.
```

	content	score	Label	text_clean	text_StopWord
0	Aplikasi yg sangat membantuuu	5	Positif	aplikasi yg sangat membantuuu	aplikasi yg membantuuu
1	Minuslah, mau masuk akun pake email susah Pada...	1	Negatif	minuslah mau masuk akun pake email susah padah...	minuslah masuk akun pake email susah email san...
2	Aplikasi samapah,paling gak berguna	1	Negatif	aplikasi samapahpaling gak berguna	aplikasi samapahpaling gak berguna

Gambar 9. Hasil stopword removal

Tabel tersebut menunjukkan hasil dari beberapa langkah pra-pemrosesan teks pada dataset. Kolom 'content' berisi ulasan atau komentar dari pengguna, 'score' memberikan nilai skor untuk setiap ulasan, dan *'Label'* memberikan klasifikasi sentimen (Positif/Negatif). Proses pra-pemrosesan diimplementasikan pada kolom *'text_clean'* dengan melakukan penghapusan karakter khusus dan konversi huruf menjadi huruf kecil. Selanjutnya, kolom *'text_StopWord'* mencerminkan teks yang telah melalui proses penghapusan kata-kata penghubung *(stopwords)*. Penghilangan *stopwords* bertujuan untuk menyederhanakan teks dan meningkatkan fokus pada kata-kata kunci yang mungkin lebih relevan dalam analisis sentimen. Data yang telah diolah ini dapat digunakan untuk langkah-langkah analisis sentimen lebih lanjut, seperti pembuatan model klasifikasi sentimen atau visualisasi pola sentimen dalam dataset.

4.4.2. Tokenizing

Tokenisasi adalah langkah untuk memecah deskripsi dari bentuk kalimat menjadi bentuk kata-kata[12]

```
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()
```

```
[nltk_data] Downloading package punkt to /root/nltk_data...
[nltk_data]   Unzipping tokenizers/punkt.zip.
```

	content	score	Label	text_clean	text_StopWord	text_tokens
0	Aplikasi yg sangat membantuuu	5	Positif	aplikasi yg sangat membantuuu	aplikasi yg membantuuu	[aplikasi, yg, membantuuu]
1	Minuslah, mau masuk akun pake email susah Pada...	1	Negatif	minuslah mau masuk akun pake email susah padah...	minuslah masuk akun pake email susah email san...	[minuslah, masuk, akun, pake, email, susah, em...
2	Aplikasi samapah,paling gak berguna	1	Negatif	aplikasi samapahpaling gak berguna	aplikasi samapahpaling gak berguna	[aplikasi, samapahpaling, gak, berguna]
3	Aplikasi nyampah JANGAN INSTAL	1	Negatif	aplikasi nyampah jangan instal	aplikasi nyampah instal	[aplikasi, nyampah, instal]

Gambar 10. Hasil Tokenizing

Pada gambar ini, menunjukkan proses pengunduhan data bahasa natural language toolkit *(NLTK)* terkait tokenisasi kata-kata. Tokenisasi adalah proses memecah teks menjadi potongan-potongan kecil yang disebut "token" atau kata-kata terpisah. Data ini diperlukan untuk memastikan bahwa algoritma atau model yang melibatkan tokenisasi dapat berfungsi dengan baik. Pada setiap baris data, kolom *'text_tokens'* menunjukkan teks yang telah melalui proses tokenisasi, di mana setiap kata atau token dipisahkan dan direpresentasikan sebagai elemen dalam daftar *(list)*. Langkah ini mempersiapkan teks untuk tahap analisis selanjutnya, seperti pembuatan model atau ekstraksi fitur dalam pemrosesan bahasa alami.

4.4.3. Stemming

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

#script ini bisa dipisah dari eksekusinya setelah pembacaaan term selesai
data_clean['text_steamindo'] = data_clean['text_tokens'].apply(lambda x:' '.join(get_stemmed_term(x)))
data_clean.head(20)
```

```
3261
------------------------
1 : aplikasi : aplikasi
2 : yg : yg
3 : membantuuu : membantuuu
4 : minuslah : minus
5 : masuk : masuk
6 : akun : akun
7 : pake : pake
8 : email : email
9 : susah : susah
```

Gambar 11. Hasil stemming

Pada proses dalam pemrosesan bahasa alami di mana kata-kata diubah menjadi bentuk dasarnya atau akar kata (stem) dengan tujuan utama menghilangkan variasi morfologis. Contohnya, kata "minuslah" dapat disederhanakan menjadi "minus".

4.5. Processing

4.5.1. Spliting 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 12. Spliting data

Kode tersebut menggunakan fungsi `train_test_split` dari pustaka Scikit-Learn untuk membagi dataset menjadi dua bagian: data pelatihan dan data pengujian. Fitur *'content'* dari *`data_clean`* digunakan sebagai variabel independen, sementara label 'Label' digunakan sebagai variabel dependen.

Dengan menetapkan *`test_size`* sebesar 0.20, 20% dari data akan dialokasikan sebagai data pengujian, dan 80% sebagai data pelatihan. Pengaturan *`random_state`* pada nilai 0 memastikan bahwa pembagian data dilakukan secara acak, tetapi memberikan reproduktibilitas hasil. Hasil dari pembagian, yaitu *`X_train`*, *`X_test`*, *`y_train`*, *dan `y_test`*, akan digunakan untuk melatih dan menguji model machine learning.

4.5.2. Pembobotan *TF-IDF*

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

```
(747,)
(747,)
(187,)
(187,)
```

Gambar 13. TF-IDF

Empat tupel angka (747,), (747,), (187,), dan (187,) kemungkinan mewakili dimensi dari data set setelah tahap pembobotan *TF-IDF*. Angka 747 pada dua tupel pertama mungkin menunjukkan bahwa ada 747 fitur atau kata kunci yang dihasilkan dari data pelatihan dan pengujian setelah penerapan pembobotan *TF-IDF*. Sementara itu, dimensi (187,) pada dua tupel terakhir dapat mengindikasikan bahwa proses pembobotan tersebut dilakukan pada subset atau validasi yang berukuran 187. Jadi, setiap angka dalam tupel tersebut mencerminkan jumlah fitur atau kata kunci yang dihasilkan oleh proses pembobotan *TF-IDF* pada data set yang bersangkutan.

4.6. Data Mining

```
from sklearn.svm import SVC
from sklearn.feature_extraction.text import TfidfVectorizer
from sklearn.metrics import accuracy_score

tfidf_vectorizer = TfidfVectorizer()

X_train_tfidf = tfidf_vectorizer.fit_transform(X_train)
X_test_tfidf = tfidf_vectorizer.transform(X_test)

svm_model = SVC()

svm_model.fit(X_train_tfidf, y_train)

y_pred = svm_model.predict(X_test_tfidf)

accuracy = accuracy_score(y_test, y_pred)
print(f'Akurasi SVM pada data pengujian: {accuracy:.2f}')
```

Gambar 14. Data Mining

Kode tersebut merupakan implementasi Support Vector Machine (SVM) untuk klasifikasi teks menggunakan pustaka scikit-learn. Pertama, dilakukan impor kelas dan fungsi yang diperlukan, seperti `SVC` untuk SVM, `TfidfVectorizer` untuk mengubah teks ke dalam matriks TF-IDF, dan `accuracy_score` untuk mengukur akurasi. Selanjutnya, objek `TfidfVectorizer` diinisialisasi untuk mengonversi data teks menjadi representasi numerik TF-IDF. Data latih dan uji kemudian diubah menggunakan objek ini. Selanjutnya, model SVM diinisialisasi dan dilatih dengan menggunakan data latih yang sudah diubah. Setelah pelatihan, model digunakan untuk memprediksi kelas dari data uji, dan akurasi prediksi diukur dengan metrik akurasi. Hasil akurasi kemudian dicetak ke layar. Secara keseluruhan, kode tersebut memberikan gambaran praktis tentang penggunaan SVM untuk klasifikasi teks dengan menerapkan representasi TF-IDF melalui pustaka scikit-learn.

4.7. Pemodelan *Nive Bayes* dan *Evaluasi* Hasil

Berikut tahapan pemodelan matriksnya :

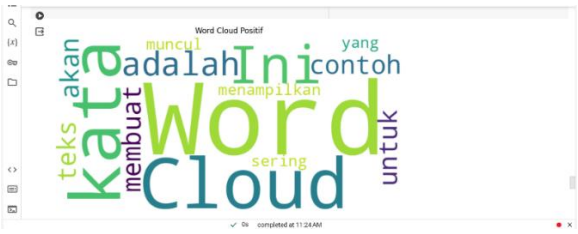
```
from sklearn.naive_bayes import MultinomialNB

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

```
MultinomialNB
MultinomialNB()
```

Gambar 15. Pemodelan matriks Naïve Bayes

Kode **from sklearn.naive_bayes import MultinomialNB** digunakan untuk mengimpor kelas **MultinomialNB** dari pustaka scikit-learn, yang menyediakan implementasi model klasifikasi Naive Bayes dengan distribusi multinomial. Selanjutnya, objek model **nb** dibuat dan diinisialisasi dengan menggunakan model multinomial. Proses pelatihan model dilakukan dengan menggunakan metode **fit**, di mana model disesuaikan dengan data latih (**tfidf_train**) dan labelnya (**y_train**). Model ini dapat digunakan untuk melakukan prediksi atau evaluasi performa pada data baru setelah proses pelatihan selesai.



Gambar 16. Word cloud positif

Pada gambar ini, di jelaskan bahwa kata yang sering muncul sesuai yang berada dalam gambar *word cloud* tersebut. Gambar ini merupakan visualisasi *word cloud* positif.

Gambar 17. Word cloud negatif

Gambar ini merupakan visualisasi dari kata yang sering dalam *word cloud negative*.

```
MultinomialNB Accuracy: 0.8877005347593583
MultinomialNB Precision: 0.8867924528301887
MultinomialNB Recall: 0.9791666666666666
MultinomialNB f1_score: 0.9306930693069307
confusion_matrix:
 [[141   3]
 [ 18  25]]
==================================================
```

	precision	recall	f1-score	support
Negatif	0.89	0.98	0.93	144
Positif	0.89	0.58	0.70	43
accuracy			0.89	187
macro avg	0.89	0.78	0.82	187
weighted avg	0.89	0.89	0.88	187

Gambar 18. Hasil akurasi

Hasil evaluasi kinerja model klasifikasi menunjukkan bahwa pada kelas "Negatif," presisi 0.89 menandakan bahwa 89% dari prediksi yang diklasifikasikan sebagai "Negatif" oleh model adalah akurat, sementara recall sebesar 0.98 menunjukkan bahwa 98% dari semua instance "Negatif" berhasil diidentifikasi. Di sisi lain, pada kelas "Positif," presisi dan recall masing-masing adalah 0.89 dan 0.58, menunjukkan bahwa model cenderung lebih baik dalam mengidentifikasi "Negatif" daripada "Positif." Skor F1 sebagai rata-rata harmonik dari presisi dan recall adalah 0.93 untuk "Negatif" dan 0.70 untuk "Positif." Jumlah instance atau support adalah 144 untuk "Negatif" dan 43 untuk "Positif." Akurasi keseluruhan model mencapai 0.89, dengan nilai rata-rata makro dan tertimbang masing-masing 0.89, 0.78, dan 0.82 untuk presisi, recall, dan skor F1. Pendekatan tertimbang memberikan perhatian lebih besar pada kelas dengan jumlah instance yang lebih besar, mencerminkan kinerja model yang lebih seimbang. Model ini memiliki presisi 0.89 untuk kedua kelas, menunjukkan keakuratan prediksi. Meskipun recall untuk kelas "Positif" rendah (0.58), skor F1 yang dihasilkan adalah 0.70. Walaupun akurasi keseluruhan model adalah 0.89, evaluasi menggunakan weighted avg menunjukkan hasil yang lebih seimbang, dengan nilai 0.89, 0.89, dan 0.88 untuk presisi, recall, dan skor F1. Untuk meningkatkan kinerja, dilakukan dengan fine-tuning model, mempertimbangkan ekstraksi fitur yang lebih baik, serta melibatkan lebih banyak data pelatihan atau menggunakan validasi silang untuk hasil yang lebih konsisten.

Evaluasi merupakan tahap penting dalam penelitian ini yang bertujuan untuk menilai sejauh mana performa algoritma klasifikasi yang digunakan. Penilaian ini menggunakan beberapa metrik, termasuk *accuracy, precision, recall, dan f-measure*Matriks konfusi digunakan untuk menyajikan analisis dalam empat aspek utama:

Accuracy: Diperlukan untuk mengukur sejauh mana prediksi model sesuai dengan nilai aktual.

Precision: Menunjukkan seberapa tepat atau akurat model dalam memprediksi nilai positif. *Precision* juga berguna dalam mengidentifikasi tingkat *False Positive* pada suatu model.

1) *Recall:* Menghitung berapa banyak nilai *Actual Positive* yang berhasil diidentifikasi sebagai *True Positive* oleh model. Recall juga digunakan sebagai metrik dalam pemilihan model terbaik saat terdapat tingkat *False Negative* yang tinggi.
2) *F-Measure*: Sebuah perbandingan rata-rata dari nilai precision dan *recall* yang diberi bobot tertentu.

Dengan menggunakan metrik-metrik ini, evaluasi dapat memberikan pemahaman yang holistik tentang kinerja algoritma klasifikasi, memungkinkan penilaian terhadap akurasi, ketepatan, serta kemampuan model dalam mengenali nilai positif dan negatif. Matriks konfusi bersama metrik evaluasi menjadi dasar untuk mengevaluasi model klasifikasi secara menyeluruh.

5. KESIMPULAN DAN SARAN

Dalam riset ini, metode pengumpulan data menggunakan *scrapping* data, *Google Colab*, dan *Python* digunakan untuk menghimpun dataset ulasan pengguna Aplikasi *McDonald's* dari Google Play *Store*. Proses pemrosesan data melibatkan instalasi library, penetapan jumlah dataset, dan penyimpanan dalam format Excel. Tahap *pelabelan* dilakukan pada data untuk melatih model mesin, sementara langkah *preprocessing* mencakup pembersihan data, penanganan nilai kosong, *folding* data, t*okenizing*, dan *stemming*. Dataset kemudian dibagi menjadi set pelatihan dan pengujian menggunakan `*train_test_split*`, dilanjutkan dengan pembobotan *TF-IDF* untuk matriks frekuensi kata-kata. Hasil evaluasi model Naive Bayes menunjukkan tingkat akurasi sebesar 88%, dengan presisi 0.90 dan recall 0.95 untuk sentimen Negatif, serta presisi 0.77 dan recall 0.61 untuk sentimen Positif. F1-score yang dihasilkan adalah 0.93 untuk sentimen Negatif dan 0.68 untuk sentimen Positif. Matriks konfusi menunjukkan 142 prediksi benar untuk sentimen Negatif dan 23 prediksi benar untuk sentimen Positif. Di samping itu, terdapat 15 prediksi salah untuk sentimen Negatif dan 7 prediksi salah untuk sentimen Positif. Evaluasi menyajikan pemahaman komprehensif tentang keunggulan dan kelemahan model dalam menganalisis sentimen ulasan Aplikasi McDonald's. Untuk sentimen Negatif, presisi sebesar 0.89, recall 0.98, dan F1-score 0.93, sementara untuk sentimen Positif, presisi 0.89, recall 0.58, dan F1-score 0.70. Dengan akurasi total sebesar 0.89, evaluasi mencakup matriks konfusi serta metrik presisi, recall, dan F1-score, memberikan pemahaman menyeluruh tentang performa model dalam mengklasifikasikan sentimen ulasan aplikasi. Evaluasi menyajikan pemahaman komprehensif tentang keunggulan dan

kelemahan model dalam menganalisis sentimen ulasan Aplikasi *McDonald's*.

DAFTAR PUSTAKA

[1] A. Nurian, "Analisis Sentimen Ulasan Pengguna Aplikasi Google Play Menggunakan Naïve Bayes," *J. Inform. dan Tek. Elektro Terap.*, vol. 11, no. 3s1, pp. 829–835, 2023, doi: 10.23960/jitet.v11i3s1.3348.

[2] U. Kusnia, F. Kurniawan, and S. Artikel, "Analisis Sentimen Review Aplikasi Media Berita Online Pada Google Play menggunakan Metode Algoritma Support Vector Machines (SVM) Dan Naive Bayes INFO ARTIKEL ABSTRAK," *J. Keilmuan dan Apl. Tek. Inform. (Explore IT)*, vol. 5, no. 36, pp. 22–28, 2022, [Online]. Available: https://doi.org/10.35891/explorit

[3] M. Z. Farhan, "Analisis Sentimen Layanan Shopeefood Pada Twitter Dengan Metode K-Nearest Neighbor, Support Vector Machine, dan Decision Tree," *J. Ilm. Inform.*, vol. 7, no. 2, pp. 95–106, 2023, doi: 10.35316/jimi.v7i2.95-106.

[4] Friska Aditia Indriyani, Ahmad Fauzi, and Sutan Faisal, "Analisis sentimen aplikasi tiktok menggunakan algoritma naïve bayes dan support vector machine," *TEKNOSAINS J. Sains, Teknol. dan Inform.*, vol. 10, no. 2, pp. 176–184, 2023, doi: 10.37373/tekno.v10i2.419.

[5] F. Sidik, I. Suhada, A. H. Anwar, and F. N. Hasan, "Analisis Sentimen Terhadap Pembelajaran Daring Dengan Algoritma Naive Bayes Classifier," *J. Linguist. Komputasional*, vol. 5, no. 1, p. 34, 2022, doi: 10.26418/jlk.v5i1.79.

[6] F. V. Sari and A. Wibowo, "Analisis Sentimen Pelanggan Toko Online Jd.Id Menggunakan Metode Naïve Bayes Classifier Berbasis Konversi Ikon Emosi," *J. SIMETRIS*, vol. 10, no. 2, pp. 681–686, 2019.

[7] W. Khofifah, D. N. Rahayu, and A. M. Yusuf, "Analisis Sentimen Menggunakan Naive Bayes Untuk Melihat Review Masyarakat Terhadap Tempat Wisata Pantai Di Kabupaten Karawang Pada Ulasan Google Maps," *J. Interkom J. Publ. Ilm. Bid. Teknol. Inf. dan Komun.*, vol. 16, no. 4, pp. 28–38, 2022, doi: 10.35969/interkom.v16i4.192.

[8] K. V. S. Toy, Y. A. Sari, and I. Cholissodin, "Analisis Sentimen Twitter menggunakan Metode Naive Bayes dengan Relevance Frequency Feature Selection (Studi Kasus: Opini Masyarakat mengenai Kebijakan New Normal)," *J. Pengemb. Teknol. Inf. dan Ilmu Komput.*, vol. 5, no. 11, pp. 5068–5074, 2021, [Online]. Available: http://j-ptiik.ub.ac.id

[9] N. Herlinawati, Y. Yuliani, S. Faizah, W. Gata, and S. Samudi, "Analisis Sentimen Zoom Cloud Meetings di Play Store Menggunakan Naïve Bayes dan Support Vector Machine," *CESS (Journal Comput. Eng. Syst. Sci.*, vol. 5, no. 2, p. 293, 2020, doi: 10.24114/cess.v5i2.18186.

[10] B. Z. Ramadhan, R. I. Adam, and I. Maulana, "Analisis Sentimen Ulasan pada Aplikasi E-Commerce dengan Menggunakan Algoritma Naïve Bayes," *J. Appl. Informatics Comput.*, vol. 6, no. 2, pp. 220–225, 2022, doi: 10.30871/jaic.v6i2.4725.

[11] J. M. Br Sembiring and H. H, "Naïve Bayes Algorithm Classification in Sentiment Analysis Covid-19 Wikipedia," *J. Tek. Inform.*, vol. 3, no. 4, pp. 869–875, 2022, doi: 10.20884/1.jutif.2022.3.4.311.

[12] M. Rezki, D. N. Kholifah, M. Faisal, P. Priyono, and R. Suryadithia, "Analisis Review Pengguna Google Meet dan Zoom Cloud Meeting Menggunakan Algoritma Naïve Bayes," *J. Infortech*, vol. 2, no. 2, pp. 264–270, 2020, doi: 10.31294/infortech.v2i2.9286.