

Penerapan *Multinomial Naive Bayes* untuk Analisis Sentimen *Review Skincare* Hanasui di TikTok oleh Dokter Detektif

Fadilatus Saadah¹, Busro Akramatul Umam², Hoiriyah³

^{1,2,3}Teknik Informatika, Fakultas Teknik, Universitas Islam Madura

¹fadilatussaadah8@gmail.com, ²busro.umam@gmail.com,

³hoiriyah.file.uim@gmail.com

ABSTRAK

Media sosial memiliki peran penting dalam membentuk opini konsumen terhadap produk *skincare*. Penelitian ini bertujuan untuk menganalisis sentimen komentar pengguna TikTok terhadap ulasan produk *skincare* Hanasui yang disampaikan oleh akun Dokter Detektif (DokTif). Sebanyak 1.000 komentar dikumpulkan menggunakan *Instant Data Scraper*, lalu melalui proses *pre-processing* teks yang mencakup *cleansing*, *case folding*, *tokenizing*, normalisasi, *stopword removal*, dan *stemming*. Setelah proses pembersihan data, jumlah komentar yang layak dianalisis mengalami penyusutan. Analisis sentimen dilakukan menggunakan metode *Multinomial Naive Bayes* dengan pendekatan *TF-IDF*. Dataset kemudian dibagi menjadi 563 data *training* dan 141 data *testing*. Hasil evaluasi menunjukkan bahwa model mencapai akurasi sebesar 90,07%, dengan mayoritas komentar tergolong dalam sentimen positif. Temuan ini menunjukkan bahwa ulasan yang disampaikan oleh DokTif memberikan pengaruh positif terhadap persepsi pengguna mengenai produk Hanasui. Penelitian ini memberikan manfaat bagi konsumen dalam memilih produk berdasarkan informasi berbasis data, bagi industri dalam merancang strategi pemasaran melalui analisis sentimen, serta bagi kalangan akademisi sebagai referensi penelitian sejenis. Selain itu, hasil studi ini dapat menjadi masukan bagi pemerintah dalam merumuskan regulasi terkait klaim produk *skincare* yang lebih transparan dan berbasis ilmiah.

Kata kunci: Analisis Sentimen, TikTok, *Multinomial Naive Bayes*, *TF-IDF*, *Skincare*.

ABSTRACT

This Social media plays an important role in shaping consumer opinions toward skincare products. This study analyzes the sentiment of TikTok user comments on Hanasui skincare reviews presented by Dokter Detektif (DokTif). A total of 1,000 comments were collected using Instant Data Scraper and processed through several text pre-processing stages, including cleansing, case folding, tokenizing, normalization, stopword removal, and stemming. Following the data cleaning process, the number of usable comments was reduced. Sentiment analysis was conducted using the Multinomial Naive Bayes method with a TF-IDF approach. The dataset was then split into 563 training data and 141 testing data. The evaluation results showed a model accuracy of 90.07%, with the majority of comments expressing positive sentiment. This indicates that DokTif's reviews have a positive influence on user perceptions of Hanasui. This study is beneficial for consumers in making data-driven product choices, for industries in developing sentiment-based marketing strategies, and for academics as a reference for similar research. Additionally, the findings can serve as input for policymakers in formulating regulations related to more transparent and evidence-based skincare product claims.

Keywords: *Sentiment Analysis, TikTok, Multinomial Naive Bayes, TF-IDF, Skincare.*

1. PENDAHULUAN

Perkembangan ekonomi digital di Indonesia mengalami pertumbuhan yang pesat. Industri kecantikan, khususnya produk *skincare*, memberikan kontribusi signifikan dengan tingkat pertumbuhan rata-rata sebesar 14% per tahun [1]. Kesadaran masyarakat, terutama Generasi Z, terhadap perawatan kulit semakin meningkat seiring dengan berkembangnya media sosial, khususnya TikTok [2]. TikTok telah menjadi platform pemasaran utama bagi industri *skincare*, dengan jumlah pengguna di Indonesia mencapai 157,6 juta pada tahun 2024 [3]. Strategi pemasaran yang diterapkan meliputi *viral marketing*, *influencer marketing*, serta ulasan berbasis data ilmiah [4]. Namun, fenomena pemasaran ini juga diiringi dengan adanya klaim berlebihan (*overclaim*) oleh beberapa merek [5]. Kondisi ini mendorong kemunculan figur seperti Dokter Detektif (DokTif) yang mengungkap klaim produk *skincare* menggunakan uji laboratorium seperti UPLC. Kredibilitas DokTif diakui oleh banyak dokter kecantikan dan ahli *skincare*, bahkan ia pernah hadir dalam rapat DPR untuk membahas regulasi klaim produk *skincare* [6].

Hanasui merupakan salah satu merek *skincare* lokal yang cukup dikenal, namun mendapatkan berbagai opini di media sosial. Beberapa ulasan menyoroti keunggulannya, sementara yang lain mempertanyakan klaim yang diajukan. Oleh karena itu, penelitian ini bertujuan untuk menganalisis sentimen masyarakat terhadap produk Hanasui setelah menerima ulasan dari DokTif, serta mengevaluasi efektivitas algoritma *Naïve Bayes* dalam mengklasifikasi opini.

Analisis sentimen, yang sering disebut juga sebagai *opinion mining*, adalah proses mengidentifikasi, mengekstrak, dan mengklasifikasikan pendapat yang terdapat dalam suatu teks. Metode ini sering digunakan dalam penelitian pemasaran dan media sosial untuk memahami bagaimana opini publik terhadap suatu produk atau layanan [7]. Dalam penelitian ini, analisis sentimen diterapkan untuk mengklasifikasikan opini masyarakat terhadap ulasan DokTif mengenai produk Hanasui. Algoritma *Naïve Bayes* dipilih karena kemampuannya dalam mengategorikan opini

dengan tingkat akurasi yang tinggi. S. Rasiban [8] menunjukkan bahwa metode ini dapat mencapai akurasi hingga 99,57%, sementara Rahman dan Lestari [9] mencatat akurasi sebesar 93,02%. Namun, belum ada penelitian yang secara spesifik membahas dampak ulasan dari DokTif dalam membentuk opini masyarakat terhadap suatu produk *skincare*. Oleh karena itu, penelitian ini menjadi yang pertama yang mengkaji bagaimana ulasan dari figur publik yang kredibel dapat memengaruhi persepsi konsumen.

Penelitian ini menggunakan analisis sentimen dengan algoritma *Multinomial Naïve Bayes*, dengan data yang dikumpulkan dari ulasan pengguna di TikTok terkait review DokTif terhadap produk Hanasui. *Multinomial Naïve Bayes* merupakan salah satu metode pembelajaran probabilistik yang berbasis pada teorema Bayes dan digunakan dalam lingkup *Natural Language Processing (NLP)*. Algoritma ini dikenal karena perhitungannya yang sederhana, menggunakan probabilitas gabungan dari kata dan kategori untuk memperkirakan probabilitas kategori pada suatu dokumen [10]. Algoritma *Multinomial Naïve Bayes* dipilih karena cocok untuk data teks seperti ulasan, serta memiliki keunggulan dalam kesederhanaan, kecepatan, dan akurasi dalam mengklasifikasikan sentimen. Selain itu, algoritma ini sering digunakan dalam penelitian analisis sentimen karena terbukti efektif dalam memproses data yang direpresentasikan dalam bentuk frekuensi kata atau *TF-IDF* [11].

Penelitian ini memiliki urgensi karena dapat memberikan wawasan bagi konsumen dalam memilih produk *skincare* berdasarkan informasi yang berbasis data, sehingga dapat menghindari klaim yang menyesatkan. Selain itu, hasil penelitian ini dapat membantu industri dalam menyusun strategi pemasaran yang lebih transparan dan berbasis kepercayaan konsumen. Dari sisi akademisi, penelitian ini dapat menjadi referensi dalam pengembangan kajian terkait analisis sentimen dan algoritma *Naïve Bayes*. Penelitian ini juga bermanfaat bagi pemerintah, terutama BPOM, dalam merumuskan regulasi yang lebih ketat terkait klaim produk *skincare* berdasarkan data ilmiah yang valid.

Dengan demikian, penelitian ini diharapkan dapat memberikan pemahaman yang lebih

mendalam mengenai bagaimana opini pengguna di media sosial dapat memengaruhi citra suatu produk serta membuktikan efektivitas metode *Naïve Bayes* dalam analisis sentiment.

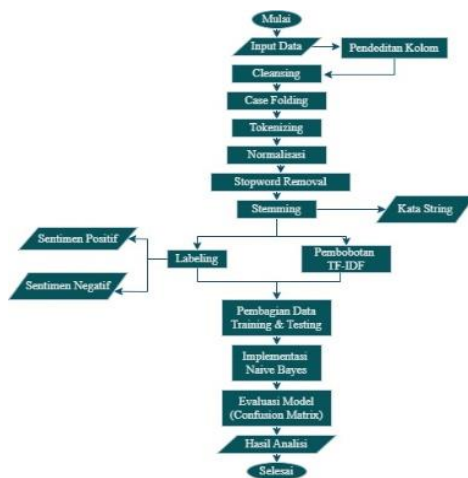
2. METODOLOGI PENELITIAN

Penelitian ini dilakukan melalui beberapa tahapan utama, yaitu pengumpulan data, pengolahan data, dan analisis hasil sentimen.

2.1 Pengumpulan Data

Data yang digunakan berupa komentar pengguna di TikTok yang membahas produk Hanasui. Namun, karena dalam video asli DokTif membandingkan Hanasui dengan produk lain, komentar yang dikumpulkan tidak diambil langsung dari video asli tersebut. Sebagai gantinya, data diperoleh dari cuplikan video yang secara khusus membahas Hanasui dan telah diunggah ulang oleh akun lain. Langkah ini dilakukan untuk memastikan bahwa opini yang dikumpulkan benar-benar relevan dengan produk Hanasui. Komentar yang berhasil dikumpulkan kemudian disimpan dalam format CSV.

2.2 Pengolahan Data



Gambar 1. Flowchart Pengolahan Data

2.2.1 Input Data

Tahap ini merupakan proses memasukkan data hasil *scraping* yang akan digunakan dalam penelitian. Data yang digunakan berupa komentar atau ulasan yang akan dianalisis lebih lanjut pada tahapan berikutnya.

2.2.2 Pre-Processing

Preprocessing adalah teknik untuk mengubah data mentah ke dalam format yang

lebih bersih dan siap untuk diproses [7]. Pada tahap ini, dilakukan beberapa langkah, yaitu:

- *Cleansing*, yaitu membersihkan teks dari karakter yang tidak diperlukan seperti URL, *mention* (@username), *hashtag* (#topik), angka, tanda baca, dan spasi berlebih.
- *Case folding*, yaitu mengubah seluruh teks menjadi huruf kecil agar tidak terjadi perbedaan analisis akibat penggunaan huruf besar dan kecil.
- *Tokenizing*, yaitu memecah teks menjadi unit kata-kata kecil (*token*) untuk memudahkan analisis lebih lanjut.
- *Normalisasi*, yaitu mengubah kata-kata tidak baku, singkatan, atau bentuk tidak standar menjadi kata yang umum dan sesuai kaidah bahasa, misalnya “gpp” menjadi “tidak apa-apa” dan “bgt” menjadi “banget”.
- *Stopword removal*, yaitu menghapus kata-kata umum yang tidak memiliki nilai informasi signifikan, seperti “yang”, “dan”, “di”, “ke”, dan sejenisnya. Daftar *stopwords* yang digunakan berasal dari *NLTK* dan Sastrawi agar hasilnya lebih akurat.
- *Stemming*, yaitu mengubah kata menjadi bentuk dasarnya menggunakan algoritma *stemming* Bahasa Indonesia dari Sastrawi, contohnya kata “berjalan” menjadi “jalan” dan “menyanyi” menjadi “nyanyi”.

2.2.3 Labeling

Setelah proses *pre-processing* selesai, data dikategorikan ke dalam dua kelas: positif dan negatif. Proses ini dilakukan menggunakan model *IndoBERT* yang telah dilatih untuk analisis sentimen dalam bahasa Indonesia. *IndoBERT* adalah model berbasis *Transformers* yang memahami konteks teks dalam bahasa Indonesia secara lebih mendalam dibandingkan metode berbasis aturan sederhana [12].

2.2.4 Pembobotan kata *TF-IDF*

Setelah data ditandai dengan label sentimen, tahap selanjutnya adalah pembobotan kata menggunakan metode *Term Frequency-Inverse Document Frequency (TF-IDF)*. Pembobotan ini bertujuan untuk memberikan nilai pada setiap kata berdasarkan seberapa sering kata tersebut muncul dalam dokumen tertentu dibandingkan dengan keseluruhan dokumen dalam dataset [13].

Rumus *TF*:

$$TF(t, d) = f_{\{t, d\}} / N_d \quad (1)$$

di mana:

- $f_{\{t, d\}}$ = jumlah kemunculan kata t dalam dokumen d
- N_d = jumlah total kata dalam dokumen d

Rumus *IDF*:

$$IDF(t) = \log(N / n_t) \quad (2)$$

di mana:

- N = jumlah total dokumen
- n_t = jumlah dokumen yang mengandung kata t

Rumus *TFIDF*:

$$TFIDF(t, d) = TF(t, d) \times IDF(t) \quad (3)$$

2.2.5 Pembagian Data

Data dibagi menjadi dua bagian yaitu 80% untuk data latih dan 20% untuk data uji. Pembagian ini dilakukan agar model dapat dilatih secara optimal menggunakan data latih, kemudian diuji menggunakan data uji untuk mengukur performa model.

2.2.6 Implementasi Algoritma *Naïve Bayes*

Naïve Bayes bekerja dengan menghitung probabilitas suatu teks termasuk dalam kategori sentimen tertentu berdasarkan distribusi kata dalam teks [14]. Keunggulan *Naïve Bayes* adalah kemampuannya dalam menangani dataset besar dan teks pendek secara efisien [15].

Pada penelitian ini, *Multinomial Naïve Bayes* diterapkan untuk melakukan klasifikasi sentimen pada komentar. *Multinomial Naïve Bayes* merupakan model pengembangan dari algoritma bayes yang cocok untuk klasifikasi teks atau dokumen. Dalam rumus *Multinomial Naïve Bayes*, kelas dokumen tidak ditentukan hanya oleh kemunculan kata, tetapi juga oleh jumlah kemunculannya [16].

Rumus *Multinomial Naïve Bayes*:

$$P(c|d) = P(c) \times P(t_1|c) \times P(t_2|c) \times \dots \times P(t_n|c) \quad (4)$$

Keterangan:

- $P(c|d)$ = Probabilitas dokumen d termasuk ke dalam kelas c
- $P(c)$ = Probabilitas awal kelas c (berapa sering kelas c muncul secara keseluruhan)
- $P(t_1|c) \times P(t_2|c) \times \dots \times P(t_n|c)$ = Probabilitas kata ke-1 sampai kata ke- n muncul dalam dokumen yang termasuk kelas c .

2.2.7 Evaluasi Model dengan *Confusion Matrix*

Langkah selanjutnya adalah mengevaluasi performa model menggunakan *Confusion Matrix*. Evaluasi ini bertujuan untuk mengukur seberapa baik model dapat membedakan antara klasifikasi yang benar atau salah [17]. Dari *Confusion Matrix* ini, beberapa metrik evaluasi dapat dihitung, seperti:

- **Akurasi:** Mengukur seberapa banyak prediksi yang benar dibandingkan dengan total data.
- **Presisi:** Mengukur seberapa tepat model dalam memprediksi kelas positif.
- **Recall:** Mengukur seberapa baik model dalam menangkap semua data positif.
- **F1-Score:** Rata-rata harmonis antara presisi dan recall. Evaluasi

2.3 Analisis Hasil

Hasil dari evaluasi model dianalisis untuk melihat apakah model sudah bekerja dengan baik atau masih memerlukan perbaikan.

3. HASIL DAN PEMBAHASAN

3.1 Pengumpulan data

Data diperoleh dari komentar pengguna TikTok terkait *review* Hanasui oleh Dokter Detektif (DokTif). Pengambilan data dilakukan menggunakan *Instant Data Scraper*, sebuah ekstensi browser untuk mengekstrak data dari halaman web secara otomatis. Dari proses ini, diperoleh 1.000 komentar.

3.2 Pengolahan Data

3.2.1 Input Data

Langkah pertama dalam pengolahan data adalah mengunggah file CSV tersebut ke *Google Colab* untuk diproses lebih lanjut. Saat data ditampilkan, ditemukan bahwa nama kolomnya kurang sesuai dengan isinya, sehingga perlu dilakukan perubahan nama kolom agar lebih jelas dan mudah digunakan dalam proses analisis berikutnya.

```
from google.colab import files
uploaded = files.upload()
df.to_csv("/content/drive/My Drive/hanasui.csv", index=False)

Tamplikan output tersembunyi

[ ] df = pd.read_csv("/content/drive/My Drive/hanasui.csv")
df.head()

Tamplikan output tersembunyi

[ ] df = pd.read_csv("/content/drive/My Drive/hanasui.csv")
df.columns = ['Link', 'Avatar', 'Username', 'Komentar', 'Waktu', 'Likes']
df.to_csv("/content/drive/My Drive/hanasuitiktok.csv", index=False)
```

Gambar 2. Kode Program Input Data dan Peneditan Kolom

Gambar tersebut menampilkan kode program Python yang digunakan dalam penelitian untuk mengolah data komentar TikTok. Tahap awalnya adalah mengunggah file CSV ke *Google Colab*, lalu membaca data tersebut ke dalam *DataFrame*.

	link-atly-focus href	css-12p3j2q-1agAvatar src	TikTokText	TikTokText 2	TikTokText 3	TikTokText 4
0	https://www.tiktok.com/@wilove06	https://p16-sign-va.tiktokcdn.com/tos-maliva-a...	Zhar	cuma pgnn tau owher narasu yg mana gess soam...	2024-10-9	358

Gambar 3. Nama Kolom Sebelum Diedit

Selanjutnya, nama-nama kolom diubah menjadi lebih informatif, seperti "*Link*", "*Avatar*", "*Username*", "*Komentar*", "*Waktu*", dan "*Likes*", dari nama default yang kurang jelas. Setelah pengeditan nama kolom ini, data disimpan kembali ke dalam file CSV baru.

3.2.2 Pre-Processing

Sebelum memasuki tahap *pre-processing* utama, dilakukan pembersihan awal terhadap data komentar untuk menghindari potensi kesalahan (*error*) selama proses analisis. Pembersihan ini meliputi penghapusan data duplikat, nilai kosong (*NaN*), serta elemen-elemen lain yang berpotensi mengganggu jalannya proses pemrosesan teks selanjutnya, seperti entri yang tidak berformat teks atau karakter-karakter yang tidak terbaca. Tahap ini penting untuk memastikan bahwa data yang digunakan berada dalam format yang sesuai dan dapat diproses secara optimal oleh algoritma pemrosesan bahasa alami.

Setelah pembersihan awal dilakukan, data kemudian melalui beberapa tahapan *pre-processing*, yaitu *cleansing*, *case folding*, *tokenizing*, normalisasi, *stopword removal*, dan *stemming*. Berikut adalah contoh komentar sebelum dan sesudah dilakukan tahap *pre-processing*:

Cleansing dan Case Folding:

```
import pandas as pd
import re
import os

df = pd.read_csv("/content/drive/My Drive/hanasuitiktobersih.csv")

# Fungsi cleansing
def clean_text(text):
    if isinstance(text, str):
        text = re.sub(r"http[s]?://www[.]*", "", text) # Hapus URL
        text = re.sub(r"@w+", "", text) # Hapus Mention
        text = re.sub(r"#w+", "", text) # Hapus Hashtag
        text = re.sub(r"d+", "", text) # Hapus Angka
        text = re.sub(r"[^w\s]", "", text) # Hapus Tanda Baca & Karakter Khusus
        text = re.sub(r"'+", "", text).strip() # Hapus Spasi Berlebih
        text = text.lower() # Ubah ke huruf kecil
    return text

df["Komentar_Clean"] = df["Komentar"].apply(clean_text)

cleaned_filename = "/content/drive/My Drive/hanasuitiktokcleaned.csv"
df.to_csv(cleaned_filename, index=False)
```

Gambar 4. Kode Program Proses Cleansing dan Case Folding

Gambar ini menunjukkan potongan kode program *Python* untuk melakukan tahap *cleansing* dan *case folding* pada data komentar TikTok. Fungsi *clean_text* dirancang untuk membersihkan teks dari karakter yang tidak diperlukan seperti *URL*, *mention* (*@username*), *hashtag* (*#topik*), angka, tanda baca, dan spasi berlebih. Selain itu, fungsi ini juga mengubah seluruh teks menjadi huruf kecil (*case folding*) untuk menyeragamkan data. Setelah proses ini selesai, data yang sudah bersih disimpan ke dalam file CSV baru bernama "*hanasuitiktokcleaned.csv*".

Tokenizing:

```
import pandas as pd
import nltk
from nltk.tokenize import word_tokenize
import os

df = pd.read_csv("/content/drive/My Drive/hanasuitiktokcleaned.csv")

# Fungsi tokenizing
def tokenize_text(text):
    if isinstance(text, str):
        return word_tokenize(text) # Tokenisasi kata
    return []

df["Komentar_Token"] = df["Komentar_Clean"].apply(tokenize_text)

tokenized_filename = "/content/drive/My Drive/hanasuitiktoktokenized.csv"
df.to_csv(tokenized_filename, index=False)
```

Gambar 5. Kode Program Proses Tokenizing

Proses ini menggunakan fungsi *word_tokenize* dari pustaka *NLTK*. Hasil tokenisasi ini kemudian disimpan dalam kolom baru bernama "*Komentar_Token*" dan diekspor ke file CSV baru bernama "*hanasuitiktoktokenized.csv*".

Normalisasi:

```
import pandas as pd
import os

df = pd.read_csv("/content/drive/My Drive/hanasuitiktoktokenized.csv")

# Normalisasi
df["Komentar_Normalized"] = df["Komentar_Token"].apply(
    lambda x: eval(x) if isinstance(x, str) and x.startswith("(") else x
).apply(normalize_tokens) # Normalisasi tetap dalam bentuk tokenized

def normalize_tokens(tokens):
    """Mengganti kata-kata tidak baku dengan kata standar"""
    return [normalization_dict.get(word, word) for word in tokens]
```

Gambar 6. Kode Program Proses Normalisasi

Proses normalisasi dilakukan dengan menggunakan kamus normalisasi (*normalization_dict*) yang berisi pasangan kata tidak baku dan padanannya yang baku, seperti "*pngn*" menjadi "*ingin*" atau "*gpp*" menjadi "*tidak apa-apa*". Setelah normalisasi, data akan disimpan dalam kolom "*Komentar_Normalized*".

Stopword Removal:

```

from nltk.corpus import stopwords
from Sastrawi.StopwordRemover.StopwordRemoverFactory import StopwordRemoverFactory
import nltk
import ast

nltk.download('stopwords')

stopwords_nltk = set(stopwords.words('indonesian'))
stopwords_sastrawi = set(StopwordRemoverFactory().get_stop_words())
all_stopwords = stopwords_nltk.union(stopwords_sastrawi)

# Fungsi hapus stopwords
def remove_stopwords(tokens):
    if isinstance(tokens, str):
        try:
            tokens = ast.literal_eval(tokens)
        except:
            tokens = tokens.split()
    return [word for word in tokens if word.lower() not in all_stopwords]

df = pd.read_csv("/content/drive/My Drive/hanasuitiktoknormalized.csv")
df['Komentar_Stopword_Removed'] = df['Komentar_Normalized'].apply(remove_stopwords)
df.to_csv("/content/drive/My Drive/hanasuitiktokstopwordremoved.csv", index=False)
    
```

Gambar 7. Kode Program Proses Stopword Removal

Untuk *Stopword Removal*, kode ini memanfaatkan pustaka *NLTK* dan *Sastrawi*; daftar *stopwords* bahasa Indonesia digabungkan dari kedua sumber tersebut untuk memastikan cakupan yang komprehensif. Sebuah fungsi (*remove_stopwords*) kemudian dibuat untuk secara iteratif memeriksa setiap kata dalam teks dan menghapusnya jika termasuk dalam daftar stopwords yang telah dikompilasi.

Stemming:

```

from Sastrawi.Stemmer.StemmerFactory import StemmerFactory

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

df = pd.read_csv("/content/drive/My Drive/hanasuitiktokstopwordremoved.csv")

# Fungsi stemming
def stemming_text(text):
    try:
        # Ubah string list ke list Python
        words = ast.literal_eval(text)
        if not isinstance(words, list):
            words = str(text).split()
    except:
        words = str(text).split() # fallback jika gagal

    stemmed_words = [stemmer.stem(word) for word in words]
    return ' '.join(stemmed_words) # hasil akhir tetap dalam bentuk string

df['Komentar_Stemmed'] = df['Komentar_Stopword_Removed'].apply(stemming_text)
    
```

Gambar 8. Kode Program Proses Stemming

Pustaka *Sastrawi* digunakan untuk membuat objek *stemmer*. Fungsi *stemming_text* yang didefinisikan akan mengambil setiap kata dari teks yang telah difilter *stopwords*, lalu memanggil metode *stem()* dari objek stemmer untuk mengonversinya ke bentuk kata dasar. Hasil dari kedua proses ini, yaitu teks yang sudah bersih dari *stopwords* dan sudah dalam bentuk dasar, kemudian akan disimpan dalam kolom *DataFrame* yang baru untuk tahapan analisis sentimen selanjutnya.

Tabel 1. Hasil Pre-Processing

Komen	Cleansi	Tokenizi	Normalisa	Stopwor	Stemmin
tar	ng &	ng	si	d	g
	Case				Removal
	Foldin				
	g				
cuma	cuma	'cuma',	'cuma',	'tau',	tau
pngn	pngn	'pngn',	'ingin',	'owner',	owner

tau	tau	'tau',	'tau',	'hanasui',	hanasui
owner	owner	'owner',	'owner',	'gess',	gess gak
hanasui	hanasui	'hanasui',	'hanasui',	'gak',	berisik
yg	yg	'yg',	'yang',	'berisik',	kaya
mana	mana	'mana',	'mana',	'kaya',	owner
gess	gess	'gess',	'gess',	'owner',	belah
soalnya	soalnya	'soalnya',	'soalnya',	'sebelah'	
gak	gak	'gak',	'gak',		
berisik	berisik	'berisik',	'berisik',		
kaya	kaya	'kaya',	'kaya',		
owner	owner	'owner',	'owner',		
sebelah	sebelah	'sebelah',	'sebelah'		

Tabel tersebut menunjukkan bagaimana sebuah komentar mentah secara bertahap diolah melalui tahapan preprocessing (*cleansing*, *tokenizing*, normalisasi, *stopword removal*, dan *stemming*) hingga siap untuk analisis.

3.2.3 Labeling

Proses *labeling* dilakukan secara otomatis menggunakan model *pre-trained* dari pustaka *Hugging Face Transformers*, yaitu model *w11wo/indonesian-roberta-base-sentiment-classifier*. Model ini merupakan hasil fine-tuning dari arsitektur *RoBERTa* untuk tugas klasifikasi sentimen dalam Bahasa Indonesia. Komentar ini kemudian diproses menggunakan *pipeline sentiment-analysis* yang akan menghasilkan label sentimen untuk masing-masing teks.

Sentimen	
positive	323
neutral	224
negative	157
Name:	count, dtype: int64

Gambar 9. Hasil Proses Labeling

Tiga label yang dihasilkan oleh model ini adalah *positive* sebanyak 323 komentar, *neutral* sebanyak 224 komentar, *negative* sebanyak 157 komentar. Distribusi ini menunjukkan bahwa mayoritas komentar memiliki sentimen positif. Jumlah total data tidak mencapai 1000 melainkan 704 data, dikarenakan telah dilakukan proses pembersihan setelah tahap *pre-processing*, seperti penghapusan data duplikat, kosong (*NaN*), dan data yang tidak relevan.

3.2.4 Pembobotan kata TF-IDF

Proses *TF-IDF* ini dilakukan melalui *TfidfVectorizer* dari *sklearn*, yang secara otomatis menghitung frekuensi kemunculan kata dalam setiap dokumen (*TF*) dan bobot kebalikannya berdasarkan frekuensi kata di seluruh korpus (*IDF*), lalu mengalikannya.

`fit_transform()` digunakan untuk data pelatihan guna mempelajari kosakata dan bobot *IDF*, sementara `transform()` digunakan pada data pengujian untuk menerapkan pembelajaran yang sama, sehingga menghasilkan matriks numerik yang merepresentasikan bobot dan pentingnya setiap kata dalam komentar.

```
hanasui    0.051709
pakai     0.048998
serum     0.028223
nya       0.018357
dok       0.016846
dtype: float64
```

Gambar 10. Kata Paling Sering Muncul

Hasil pembobotan kata dengan metode TF-IDF, di mana kata yang sering muncul dan dianggap penting memiliki nilai lebih tinggi. Kata "hanasui" memiliki bobot tertinggi, diikuti oleh "pakai" dan "serum", yang menunjukkan bahwa kata-kata ini dominan dalam data.

3.2.5 Pembagian Data

Setelah dilakukan pembagian data, terdapat 563 data untuk *training* dan 141 data untuk *testing*. Data training digunakan untuk melatih model, sedangkan data testing digunakan untuk menguji performa model.

3.2.6 Implementasi Algoritma *Naïve Bayes*

```
from imblearn.over_sampling import SMOTE
smote = SMOTE(random_state=42)
X_resampled, y_resampled = smote.fit_resample(X_train, y_train)
```

Gambar 11. Kode Program Proses *Smote*

Sebelum mengimplementasikan algoritma klasifikasi *Multinomial Naive Bayes*, dilakukan penanganan terhadap permasalahan ketidakseimbangan data (*class imbalance*). Ketidakseimbangan ini terjadi karena jumlah komentar untuk masing-masing kelas sentimen tidak merata. Untuk mengatasi hal ini, digunakan teknik *SMOTE (Synthetic Minority Over-sampling Technique)*, yaitu metode *oversampling* yang secara sintetik menghasilkan data tambahan pada kelas minoritas berdasarkan kemiripan vektor fitur. [18].

Akurasi: 0.900709219858156
Laporan Klasifikasi:

	precision	recall	f1-score	support
negative	0.79	0.97	0.87	31
neutral	0.89	0.87	0.88	45
positive	0.98	0.89	0.94	65
accuracy			0.90	141
macro avg	0.89	0.91	0.89	141
weighted avg	0.91	0.90	0.90	141

Gambar 12. Kode Program Proses dan Hasil *Modeling*

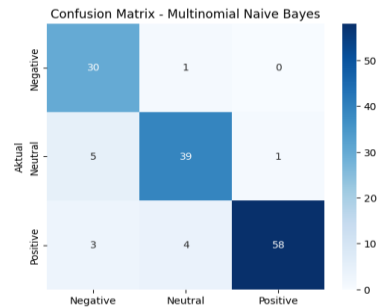
Hasil evaluasi, model menunjukkan performa yang sangat baik dengan nilai akurasi sebesar 90.07%. Detail metrik evaluasi dari

masing-masing kelas sentimen adalah sebagai berikut:

- Negatif:
Precision: 0.79, Recall: 0.97, F1-Score: 0.87, Support: 31 data
- Netral:
Precision: 0.89, Recall: 0.87, F1-Score: 0.88, Support: 45 data
- Positif:
Precision: 0.98, Recall: 0.89, F1-Score: 0.94, Support: 65 data

Secara keseluruhan model mampu mengklasifikasikan ketiga kelas dengan cukup seimbang tanpa terlalu bias pada satu kelas tertentu. Hal ini menegaskan bahwa *Multinomial Naive Bayes*, dapat menjadi pendekatan yang efektif dalam tugas analisis sentimen, terutama setelah dilakukan penanganan terhadap ketidakseimbangan data.

3.2.7 Evaluasi Model dengan *Confusion Matrix*



Gambar 13. Hasil Proses Evaluasi *Confusion Matrix*

Gambar di atas menunjukkan *Confusion Matrix* dari hasil evaluasi model klasifikasi. Berikut penjelasannya:

Kelas Negatif: Sebanyak 30 data negatif berhasil diprediksi dengan benar oleh model (*TP*). Namun, terdapat 1 data negatif yang salah diklasifikasikan sebagai *netral (FN)*, dan tidak ada yang salah diklasifikasikan sebagai positif.

Kelas Netral: Modelnya mengklasifikasikan 39 data netral dengan benar (*TP*). Namun, terdapat 5 data netral yang salah diklasifikasikan sebagai negatif (*FN*), dan 1 data netral diklasifikasikan salah sebagai positif (*FP*). Sementara itu, 1 data negatif dan 4 data positif salah diklasifikasikan sebagai netral (*FP*).

Kelas Positif: Sebanyak 58 data positif berhasil diprediksi dengan benar (*TP*). Di sisi

lain, terdapat 3 data positif yang salah diklasifikasikan sebagai negatif dan 4 data positif sebagai netral (*FN*). Selain itu, 1 data netral salah diprediksi sebagai positif (*FP*).

3.1 Analisis Hasil

Multinomial Naive Bayes menunjukkan bahwa review skincare Hanasui oleh Dokter Detektif di TikTok berpengaruh terhadap opini masyarakat. Model yang digunakan mencapai akurasi sebesar 90,07%, yang berarti cukup efektif dalam mengklasifikasikan sentimen.

Dari *confusion matrix*, model lebih banyak mengklasifikasikan sentimen dengan benar dibandingkan kesalahan prediksi, dengan 58 data positif (*TP*), 30 data negatif (*FN*), dan 39 data netral (*TN*). Jumlah total prediksi benar sebanyak 127 dari 141 data, menunjukkan performa model yang baik. Kesalahan klasifikasi paling banyak terjadi antara kelas positif dan netral. Nilai *precision*, *recall*, dan *f1-score* juga mendukung bahwa model ini mampu membedakan ketiga kelas sentimen secara cukup baik.

Kesimpulannya, *review* dari Dokter Detektif memberikan pengaruh positif terhadap persepsi masyarakat terhadap skincare Hanasui. Berdasarkan hasil analisis, mayoritas sentimen yang muncul bersifat positif, menunjukkan bahwa ulasan tersebut dapat meningkatkan kepercayaan dan minat konsumen terhadap produk ini di pasaran.

4. KESIMPULAN

Kesimpulannya, menunjukkan bahwa analisis sentimen terhadap komentar TikTok terkait *review* skincare Hanasui oleh Dokter Detektif berhasil mengungkap pola opini masyarakat. Dengan menggunakan metode *Multinomial Naïve Bayes*, model yang dikembangkan mencapai akurasi sebesar 90.07%, yang menunjukkan bahwa metode ini cukup efektif dalam mengklasifikasikan sentimen positif dan negatif.

Dari hasil analisis, sebagian besar komentar yang dikumpulkan memiliki sentimen positif, yang mengindikasikan bahwa ulasan Dokter Detektif memberikan pengaruh yang baik terhadap persepsi masyarakat terhadap Hanasui. Hal ini didukung oleh hasil *Confusion Matrix* yang menunjukkan bahwa model mampu mengidentifikasi lebih banyak komentar positif dengan benar dibandingkan

kesalahan prediksi. Selain itu, kata-kata yang memiliki bobot tertinggi dalam metode *TF-IDF*, seperti "Hanasui", "pakai", dan "serum", menunjukkan bahwa diskusi dalam komentar memang berfokus pada produk tersebut.

DAFTAR PUSTAKA

- [1] S. Kinanti and A. SAPUTRA, "Pengaruh Celebrity Endorser Brand Image dan Kualitas Produk Terhadap Keputusan Pembelian Produk Scarlett Whitening di Kota Batam," *eCo-Buss*, vol. 5, no. 3, pp. 880–893, 2023, doi: 10.32877/eb.v5i3.635.
- [2] S. Aninditya, A. Setiadarma, and A. Irawatie, "Pengaruh Electronic Word Of Mouth Di Media Sosial Tiktok Terhadap Keputusan Pembelian Produk Skincare Somethinc," *Ikon --Jurnal Ilm. Ilmu Komun.*, vol. 29, no. 2, pp. 154–162, 2023, doi: 10.37817/ikon.v29i2.3471.
- [3] Raisya Aliya Fatika, "10 Negara dengan Pengguna TikTok Terbesar, Indonesia Urutan Berapa?," *GoodStats Data (atau data.goodstats.id)*, 2024, [Online]. Available: https://data.goodstats.id/statistic/10-negara-dengan-pengguna-tiktok-terbesar-indonesia-urutan-berapa-xFOgI?utm_source=chatgpt.com
- [4] A. F. Pasaribu, T. I. Fadhila Raha, and B. Dharma, "Pengaruh Content Marketing , Viral Marketing Dan Influencer Terhadap Minat Beli Produk Skincare Pada Mahasiswa," *Ecobisma (Jurnal Ekon. Bisnis Dan Manajemen)*, vol. 10, no. 2, pp. 81–93, 2023, doi: 10.36987/ecobi.v10i2.4432.
- [5] A. A. P. Kuncoro and M. Syamsudin, "Perlindungan Konsumen terhadap Overclaim Produk Skincare," *Pros. Semin. Huk. Aktual*, vol. 2, no. 3, pp. 73–84, 2024.
- [6] K. N. Azizah, "DPR Panggil Influencer Kosmetik Bahas Perlindungan Konsumen, Ada Doktif," *DetikHealth (detik.com)*, 2025, [Online]. Available: <https://health.detik.com/berita-detikhealth/d-7819876/dpr-panggil-influencer-kosmetik-bahas-perlindungan-konsumen-ada->

- doktif?utm_source=chatgpt.com
- [7] A. M. Saprizal and N. Anisa, "Analisis Sentimen Tiktok: Wajib Militer dengan Metode Lexicon Based dan Naive Bayes Classifier," *TAMIKA J. Tugas Akhir Manaj. Inform. Komputerisasi Akunt.*, vol. 4, no. 2, pp. 242–246, 2024, doi: 10.46880/tamika.vol4no2.pp242-246.
- [8] S. R. Rasiban, "Analisis Sentimen Opini Masyarakat Terhadap Stadion Jakarta Internasional Stadium (Jis) Pada Twitter Dengan Perbandingan Metode Naive Bayes Dan Support Vector Machine," *J. Sains dan Teknol.*, vol. 5, no. 3, pp. 951–960, 2024.
- [9] F. N. Rahman and S. Lestari, "Analisis Sentimen Masyarakat Terhadap Pemerintah di Era Kabinet Joko Widodo Berdasarkan Sosial Media X Menggunakan Naive bayes dan K-Nearest Neighbor (KNN)," *INTECOMS J. Inf. Technol. Comput. Sci.*, vol. 7, no. 5, pp. 1537–1544, 2024, doi: 10.31539/intecom.s.v7i5.11823.
- [10] I. Sabar and S. Mulyati, "Implementasi Multinomial Naive Bayes Untuk Analisis Sentimen Ulasan Pengguna Aplikasi Maxim Implementation of Multinomial Naive Bayes for Sentiment Analyst Maxim Application User Reviews on Google Play Store," vol. 3, no. September, pp. 364–372, 2024.
- [11] R. M. Awangga and N. H. Khonsa', "Analisis Performa Algoritma Random Forest dan Naive Bayes Multinomial pada Dataset Ulasan Obat dan Ulasan Film," *InComTech J. Telekomun. dan Komput.*, vol. 12, no. 1, p. 60, 2022, doi: 10.22441/incomtech.v12i1.14770.
- [12] Y. A. Singgalen, "Performance Analysis of IndoBERT for Sentiment Classification in Indonesian Hotel Review Data," *J. Inf. Syst. Res.*, vol. 6, no. 2, pp. 976–986, 2025, doi: 10.47065/josh.v6i2.6505.
- [13] Y. Akbar and T. Sugiharto, "Analisis Sentimen Pengguna Twitter di Indonesia Terhadap ChatGPT Menggunakan Algoritma C4.5 dan Naive Bayes (Yuma Akbar 1*, Tri Sugiharto 2) Analisis Sentimen Pengguna Twitter di Indonesia Terhadap ChatGPT Menggunakan Algoritma C4.5 dan Naive Bayes," *J. Sains dan Teknol.*, vol. 5, no. 1, pp. 115–122, 2023, [Online]. Available: <https://doi.org/10.55338/saintek.v4i3.1368>
- [14] M. K. Insan, U. Hayati, and O. Nurdiawan, "Analisis Sentimen Aplikasi Brimo Pada Ulasan Pengguna Di," *J. Mhs. Tek. Inform.*, vol. 7, no. 1, pp. 478–483, 2023.
- [15] E. Laia and M. Yamin, "Penerapan Algoritma Naive Bayes dalam Menganalisis Sentimen pada Review Pengguna E-Commerce," *Media Online*, vol. 4, no. 1, pp. 305–316, 2023, doi: 10.30865/klik.v4i1.1186.
- [16] F. A. Ramadhan, S. H. Sitorus, and T. Rismawan, "Penerapan Metode Multinomial Naive Bayes untuk Klasifikasi Judul Berita Clickbait dengan Term Frequency - Inverse Document Frequency," *J. Sist. dan Teknol. Inf.*, vol. 11, no. 1, p. 70, 2023, doi: 10.26418/justin.v11i1.57452.
- [17] R. Sanjaya, E. Tohidi, E. Wahyudi, and K. Kaslani, "Analisis Sentimen Terhadap Berhentinya Tiktokshop Pada Media Sosial Twitter Menggunakan Algoritma Naive Bayes," *JATI (Jurnal Mhs. Tek. Inform.*, vol. 8, no. 1, pp. 507–514, 2024, doi: 10.36040/jati.v8i1.8443.
- [18] T. A. Dewi and E. Mailoa, "Perbandingan Implementasi Metode Smote Pada Algoritma Support Vector Machine (Svm) Dalam Analisis Sentimen Opini Masyarakat Tentang Mixue," *J. Indones. Manaj. Inform. dan Komun.*, vol. 4, no. 3, pp. 849–855, 2023, doi: 10.35870/jimik.v4i3.289.