ISSN: 1829-667X / E-ISSN: 2460-9021
DOI:10.31515/telematika.v19i1.6577

Sentiment Analysis of the Covid-19 Vaccine Using the Naive Bayes Algorithm and Levenshtein Distance Word Correction

Analisis Sentimen Vaksin Covid-19 Menggunakan Algoritma Naive Bayes dan Perbaikan Kata Levenshtein Distance

Fahmi Reza Prasastio[1], Heriyanto[2], Wilis Kaswidjanti[3]
[1,2,3] Informatika, Universitas Pembangunan Nasional Veteran Yogyakarta, Indonesia

[1*]123170068@student.upnyk.ac.id, [2]heriyanto@upnyk.ac.id, [3] wilisk@upnyk.ac.id

Article's Information / Informasi Artikel
Received: January 2022
Revised: January 2022
Accepted: February 2022
Published: February 2022

Abstract

Purpose: Knowing how accurate the use of the word improvement of the Levenshtein Distance method is for sentiment analysis of the Covid-19 vaccine using the Naïve Bayes method.
Design/methodology/approach: Implementing word improvement using the Levenshtein Distance algorithm for preprocessing and the Naïve Bayes algorithm in analyzing the sentiment of public comments about the Covid-19 vaccine.
Findings/result: By applying word refinement to the dataset used, it can increase the accuracy of the built Naïve Bayes model. The accuracy of testing using old test data that opened 479 data increased from 61% to 71% and tests with new test data that opened 100 data accuracy increased from 59% to 66%. By improving words, it is proven to make the system easier to classify so that it can increase accuracy. To classify new test data, it achieves low accuracy even though the data presented only reaches 100 data, this is due to the system being incapable of classifying new data that has never been trained before.
Originality/value/state of the art:
This study uses data with a total of 2394 data originating from comments on the Indonesian Ministry of Health's Instagram account. For preprocessing, word correction was performed using the Levenshtein Distance algorithm and for comment analysis using the Naïve Bayes algorithm with TF-IDF feature extraction

ISSN: 1829-667X / E-ISSN: 2460-9021
DOI:10.31515/telematika.v19i1.6577

Keywords: Sentiment Analysis; Topic Modeling; Machine Learning
Kata kunci: Analisis sentimen; Pemodelan topik; Machine Learning

Abstrak

Tujuan: Mengetahui seberapa akurat penggunaan perbaikan kata metode Levenshtein Distance terhadap analisis sentimen vaksin Covid-19 menggunakan metode Naïve Bayes.
Perancangan/metode/pendekatan: Menerapkan perbaikan kata Levenshtein Distance untuk preprocessing dan algoritma Naïve Bayes dalam melakukan analisis sentimen komentar masyarakat tentang vaksin Covid-19.
Hasil: Dengan diterapkannya perbaikan kata pada dataset yang digunakan dapat meningkatkan akurasi dari model Naïve Bayes yang dibangun. Akurasi pengujian menggunakan data uji lama yang berjumlah 479 datameningkat dari 61% menjadi 71% dan pengujian dengan data uji baru yang berjumlah 100 data akurasi meningkat dari 59% menjadi 66%. Namun untuk klasifikasi data testing baru memperoleh akurasi yang cukup rendah walaupun data yang dites hanya berjumlah 100 data, hal ini disebabkan oleh sistem yang kurang mampu dalam melakukan klasifikasi data baru yang belum pernah dilakukan training sebelumnya.
Keaslian/ state of the art: Penelitian ini menggunakan data dengan jumlah 2394 data yang berasal dari komentar akun Instagram Kemenkes RI. Untuk preprocessing dilakukan perbaikan kata dengan algoritma Levenshtein Distance dan untuk analisis komentar menggunakan algoritma Naïve Bayes dengan ekstraksi fitur TF-IDF

1. Pendahuluan

Virus Corona atau Severe Acute Respiratory Syndrome Coronavirus 2 (SARS-CoV-2) yang menyerang sistem pernapasan sehingga menyebabkan gangguan pada sistem pernapasan, infeksi paru-paru, bahkan kematian telah ditetapkan sebagai pandemi global pada tanggal 11 Maret 2020 lalu oleh organisasi kesehatan dunia yaitu World Health Organization (WHO). Virus ini bisa menyerang bayi, anak-anak, remaja, dewasa, hingga lansia. Kasus Covid-19 pertama kali ditemukan di Wuhan, China pada Desember 2019 dan meningkat pesat memasuki tahun 2020. Virus ini akhirnya menyebar ke berbagai penjuru dunia termasuk Indonesia. Dengan pesatnya penyebaran Covid-19 di dunia yang dapat menimbulkan bahaya yang semakin besar jika tidak segera ditangani, maka para ilmuan di dunia mengembangkan vaksin untuk mencegah penyebaran virus ini [1]. Pemerintah Indonesia juga turut aktif dalam pencanangan program vaksinasi melalui Perpres Nomor 90 Tahun 2020, hal ini menimbulkan pro dan kontra dari masyarakat mengenai keamanan dari vaksin itu sendiri. Dengan banyaknya komentar masyarakat Indonesia mengenai vaksin covid-19 tersebut pada media sosial maka dapat diperoleh sumber data yang dapat dimanfaatkan untuk melakukan penelitian yang bermanfaat, salah satunya untuk melakukan klasifikasi persepsi masyarakat terhadap vaksin Covid-19.

Klasifikasi adalah teknik mengolah data dengan melakukan pengelompokkan data berdasarkan karakteristik data atau ciri-ciri khususnya [2]. Analisis sentimen merupakan salah satu cabang dari text mining (teknik yang digunakan untuk melakukan klasifikasi dokumen teks) yang biasa dikenal dengan nama opinion mining dan bertujuan untuk menentukan persepsi publik terhadap permasalahan, kejadian atau topik pembahasan yang ada [3]. Analisis sentimen dilakukan dengan mengkategorikan polaritas teks sehingga dapat dikelompokkan berdasarkan sentimen positif, negatif, atau netral [4].Beberapa penelitian telah dilakukan untuk analisis sentimen tentang vaksin menggunakan berbagai macam metode. Penelitian yang dilakukan oleh penelitian yang dilakukan oleh Fajar Fathur Rachman dan Setia Pramana tentang analisis sentimen serta pengelompokkan opini masyarakat yang menggunakan metode Lexicon Based dan Latent Dirichlet Allocation (LDA) dengan data tweet yang berasal dari opini masyarakat tentang vaksin Covid-19 data ini menghasilkan sentimen positif sebesar 29,6%, sentimen netral sebesar 46,8%, dan sentimen negatif sebesar 23,6% selain itu model LDA yang dibangun dapat digunakan untuk menangkap dan mengelompokkan data opini masyarakat tersebut [3]. Kemudian penelitian yang dilakukan oleh Mulyawan & Slamet yang melakukan klasifikasi sentimen terhadap vaksin Covid-19 di Indonesia menggunakan metode Support Vector Machine (SVM) dan menggunakan data tweet. Data kemudian dilakukan preprocessing dan dilabeli secara manual sesuai dengan kelasnya yaitu sentimen positif, negatif, dan netral. Klasifikasi kemudian dilakukan dengan metode Support Vector Machine yang menghasilkan nilai f1-score=89,219%, akurasi=82,738%, presisi=83,333%, dan recall=96% [5]. Penelitian selanjutnya dilakukan oleh Laurensz & Sediyono menggunakan metode Naïve Bayes dan Support Vector Machine (SVM) untuk mengetahui perbandingan akurasi dari kedua metode tersebut dalam mengatasi analisis sentimen vaksin Covid-19 menggunakan data yang berasal dari media sosial Twitter. Data kemudian dilakukan pelabelan secara manual, preprocessing, pembobotan TF-DIF, dan proses klasifikasi dengan menggunakan metode Naïve Bayes dan SVM untuk mengetahui perbandingan akurasinya. Metode Naïve Bayes mempunyai rata-rata tingkat akurasi lebih besar dengan persentase sebesar 85,59%, sedangkan metode SVM sebesar 84,41% [6].

Berbagai macam algoritma telah digunakan untuk menyelesaikan permasalahan analisis sentimen vaksin yang tentunya memiliki kelebihan dan kekurangannya masing-masing. Walaupun beberapa algoritma yang digunakan sebelumnya memiliki kelebihan masing-masing namun hal yang masih sering terjadi yaitu dalam menangani permasalahan analisis sentimen sangat erat sekali terjadi kesalahan penulisan pada komentar yang ada, sehingga sistem salah mengidentifikasi komentar yang mengakibatkan ketidak akuratan dalam melakukan klasifikasi [7]. Berdasarkan uraian yang sudah dipaparkan diatas, pada penelitian ini akan dilakukan analisis sentimen vaksin Covid-19 dengan menambahkan proses perbaikan kata pada preprocessing untuk mengatasi kesalahan penulisan kata pada dataset. Data komentar masyarakat yang didapatkan pada akun media sosial Instagram milik Kemenkes RI mengenai vaksin Covid-19 dengan proses web scraping selanjutnya akan dilakukan proses preprocessing dan dilanjutkan dengan perbaikan kata dengan metode Levenshtein Distance, metode ini dipilih karena dapat mengatasi masalah kata yang tidak baku dengan cara memperbaikinya [8]. Setelah dilakukan preprocessing dan perbaikan kata tentunya data siap untuk dilakukan analisis sentimen dengan menggunakan algoritma Naïve Bayes, metode ini dipilih karena memiliki kesederhanaan model, kecepatan, serta keefektifan metode tersebut dalam mengklasifikasikan dokumen teks [8]. Tujuan dari penelitian ini adalah untuk mengetahui hasil akurasi dari metode

klasifikasi Naïve Bayes yang dikombinasikan dengan penambahan metode untuk memperbaiki kesalahan penulisan kata yaitu dengan metode Levenshtein Distance pada preprocessing dalam mengatasi analisis sentimen vaksin covid-19.

2. Metode

Metode penelitian yang digunakan dalam penelitian ini adalah metode kuantitatif. Metode kuantitatif digunakan pada model matematis dari algoritma Naïve Bayes. Penelitian ini merupakan penelitian implementatif yang bertujuan untuk membangun sebuah sistem analisis sentimen dengan mengimplementasikan algoritma Naïve Bayes, sebelum dilakukan analisis sentimen data terlebih dahulu dilakukan preprocessing yang meliputi *case folding, remove punctuation, remove number, tokenizing, stopword removal, stemming*, dan perbaikan kata.

2.1. Pengumpulan Data

Data yang digunakan pada penelitian ini merupakan data sekunder yang berasal dari komentar masyarakat pada akun media sosial Instagram milik Kementerian Kesehatan Republik Indonesia dengan username kemenkes_ri. Data dikumpulkan dengan menggunakan teknik web scraping pada postingan Instagram Kemenkes RI yang berkaitan dengan vaksin pada bulan yaitu Juni 2021 sampai dengan Juli 2021 karena lonjakan kasus Covid-19 di Indonesia sangat tinggi pada bulan tersebut. Web scraping dilakukan dengan bahasa pemrograman Python dengan library Selenium untuk kemudian menghasilkan dataset berjumlah 2394 data dan disimpan dengan format .csv. Adapun **Tabel 1** merupakan hasil scraping dari postingan Instagram Kemenkes RI.

Tabel 1. Data Hasil *Scraping*

No.	Username	Komentar
1	kemenkes_ri	Vaksinasi bukan hanya buat kamu, tapi untuk kesehatan dan keselamatan kita semua. Yuk segera vaksinasi, karena semakin cepat divaksinasi, maka semakin cepat pula kekebalan kelompok terbentuk. Tapi ingat, walaupun udah divaksin, protokol kesehatan harus tetap dijalankan secara ketat Salam sehat!
2	hafeezhito	Please kemenkes, segera lebih permudah lagi akses masyarakat untuk vaksin ya.
3	meidinti89	Iya bener bgt, vaksin sangat membantu untuk meringkan gejala covid. Ini nyata krn keluarga saya sedang berjuang melawan covid dan sudah divaksin semua, jd gejalanya tdk terllau berat.

2.2. Pelabelan Data

Kemudian data tersebut akan dilakukan labelling secara manual oleh dua orang psikolog berdasarkan kelas sentimen positif, negatif, dan netral. Proses pelabelan data dilakukan oleh dua psikolog untuk membandingkan hasil label dari dua psikolog dan jika terdapat perbedaan antara label yang diberikan psikolog maka data komentar dibaca ulang dan ditentukan label akhirnya yang lebih tepat diantara perbedaan tersebut. Dari data yang berjumlah 2394 data terdapat 1005 label positif, 547 label netral, dan 842 label negatif yang akan digunakan untuk menghitung akurasi pada penelitian ini. Pelabelan data komentar dapat dilihat pada **Tabel 2**.

Tabel 2. Pelabelan Data Ulasan

No.	Username	Komentar	Psikolog 1	Psikolog 2	Label Akhir
1	kemenkes_ri	Vaksinasi bukan hanya buat kamu, tapi untuk kesehatan dan keselamatan kita semua. Yuk segera vaksinasi, karena semakin cepat divaksinasi, maka semakin cepat pula kekebalan kelompok terbentuk. Tapi ingat, walaupun udah divaksin, protokol kesehatan harus tetap dijalankan secara ketat Salam sehat!	Netral	Netral	Netral
2	hafeezhito	Please kemenkes, segera lebih permudah lagi akses masyarakat untuk vaksin ya.	Negatif	Negatif	Negatif
3	meidinti89	Iya bener bgt, vaksin sangat membantu untuk meringkan gejala covid. Ini nyata krn keluarga saya sedang berjuang melawan covid dan sudah divaksin semua, jd gejalanya tdk terllau berat.	Positif	Positif	Positif

2.3. Preprocessing Data

Preprocessing sendiri merupakan tahap pembersihan data terhadap noise yang tidak diperlukan sehingga dapat diperoleh data yang berkualitas dan siap untuk dilakukan proses selanjutnya. Hasil yang diharapkan dari proses preprocessing data adalah kumpulan data akhir yang berkualitas sehingga dapat digunakan dengan baik untuk proses data mining berikutnya [9]. Pada tahap ini terdapat beberapa proses seperti case folding, dilanjutkan dengan remove punctuation untuk menghilangkan tanda baca, remove number untuk menghilangkan angka, dan stopword removal. Selanjutnya melakukan tokenizing dan stemming. Setelah itu dilakukan normalisasi/perbaikan kata menggunakan metode Levenshtein Distance. Berikut setiap tahapan preprocessing yang dilakukan pada penelitian ini.

2.3.1. *Case Folding*

Case folding merupakan proses pertama yang dilakukan dalam teks preprocessing dan bertujuan untuk mengubah seluruh huruf pada data menjadi huruf kecil [5]. Contoh dari *case folding* dapat dilihat pada **Tabel 3**.

Tabel 3. Contoh *Case Folding*

Sebelum	Sesudah
Halo, klo tetap 5m di lakukan . Buat apa di vaksin ?? Pembodohan	halo, klo tetap 5m di lakukan . buat apa di vaksin ?? pembodohan

2.3.2. Remove Punctuation

Remove punctuation merupakan proses menghapus tanda baca pada teks karena dianggap tidak penting dan termasuk delimiter, contoh tanda baca yang yang dihapus titik (.), koma(,), tanda tanya (?), tanda seru (!), slash (/), hastag (#), dan lain-lain. Hal ini bertujuan untuk mengurangi beban saat melakukan pemrosesan dan karena tanda baca tidak mempunyai arti dalam proses

analisis sehingga perlu dihilangkan. Contoh dari remove punctuation dapat dilihat pada **Tabel 4**.

Tabel 4. Contoh *Remove Punctuation*

Sebelum	Sesudah
halo, klo tetap 5m di lakukan . buat apa di vaksin ?? pembodohan	halo klo tetap 5m di lakukan buat apa di vaksin pembodohan

2.3.3. Remove Number

Remove number bertujuan untuk menghilangkan angka dari dokumen teks karena angka tidak memiliki makna. Contoh dari remove number dapat dilihat pada **Tabel 5**.

Tabel 5. Contoh *Remove Number*

Sebelum	Sesudah
halo klo tetap 5m di lakukan buat apa di vaksin pembodohan	halo klo tetap m di lakukan buat apa di vaksin pembodohan

2.3.4. Tokenizing

Tokenizing adalah tahapan dalam preprocessing teks untuk memisahkan/memenggal setiap kata yang tersusun pada teks [10]. Contoh tokenizing dapat dilihat pada **Tabel 6**.

Tabel 6. Contoh *Tokenizing*

Sebelum	Sesudah
halo klo tetap m di lakukan buat apa di vaksin pembodohan	'halo', 'klo', 'tetap', 'm', 'di', 'lakukan', 'buat', 'apa', 'di', 'vaksin', 'pembodohan'

2.3.5. Stopword Removal

Proses stopword removal bertujuan untuk membuang kata yang tidak memiliki arti atau kata tidak penting. Daftar kata yang merupakan stopword terdapat pada kamus stopword. Pada penelitian ini akan digunakan library nltk Python untuk melakukan stopword removal. Contoh stopword removal dapat dilihat pada **Tabel 7**.

Tabel 7. Contoh *Stopword Removal*

Sebelum	Sesudah
'halo', 'klo', 'tetap', 'm', 'di', 'lakukan', 'buat', 'apa', 'di', 'vaksin', 'pembodohan'	'halo', 'klo', 'm', 'lakukan', 'vaksin', 'pembodohan'

2.3.6. Stemming

Setelah data dilakukan stopword removal maka langkah selanjutnya yaitu proses stemming yang dilakukan untuk mengembalikan kata ke bentuk dasar yaitu dengan menghilangkan imbuhannya. Pada penelitian ini karena data yang digunakan menggunakan bahasa Indonesia maka akan dilakukan stemming dengan library Sastrawi. Library sastrawi menerapkan algoritma Nazief dan Adriani. Data yang digunakan merupakan data hasil proses stopword

removal. Membuang imbuhan dari kata dimulai dengan *inflection suffixes* (-lah, -kah, -ku, dll), kemudian menghapus *derivational suffix* (-i, -kan, -an) dan terakhir menghapus derivational prefix (-be, -di, -me, -pe, -se, dan –te). Contoh stemming dapat dilihat pada **Tabel 8**.

Tabel 8. Contoh *Stemming*

Sebelum	**Sesudah**
'halo', 'klo', 'm', 'lakukan', 'vaksin', 'pembodohan'	'halo', 'klo', 'm', 'laku', 'vaksin', 'bodoh’

2.3.7. Perbaikan Kata *Levenshtein Distance*

Data yang sudah melalui proses stemming kemudian akan dilanjutkan dengan proses selanjutnya yaitu perbaikan kata dengan metode Levenshtein Distance. Setiap kata pada data akan dibandingkan dengan kamus kata baku, kemudian jika kata tidak terdapat pada kamus maka akan dilakukan perbaikan menggunakan algoritma Levenshtein Distance proses insertion(penyisipan), deletion(penghapusan), dan substitution(penggantian) [11]. Perbaikan ini dilakukan untuk memperbaiki kesalahan penulisan komentar oleh pengguna seperti singkatan atau bahasa penulisan sendiri yang membuat kata menjadi tidak baku. Persamaan untuk algoritma Levenshtein Distance dapat dilihat pada persamaan 1, persamaan 2, persamaan 3.

$$Dist_{a,b}(i,j) = Min \begin{cases} Dist_{a,b}\big((i,j-1)+1\big) = Min & (1) \\ Dist_{a,b}\big((i-1,j)+1\big) = Min & (2) \\ Dist_{a,b}\left((i-1,j-1)+1_{(a_i \neq b_j)}\right) = Min & (3) \end{cases}$$

Keterangan:

$Dist$: Jarak

a: String pertama

b: String kedua

i: iterasi posisi string pertama

j: iterasi posisi string kedua

Pada penelitian ini perbaikan kata yang dilakukan menggunakan kata-kata yang berasal dari kamus Colloquial Indonesian Lexicon yang merupakan hasil dari penelitian yang dilakukan oleh Nikmatun Aliyah Salsabila et al [12], karena kamus tersebut menggunakan metode Levenshtein Distance pada pembuatannya. Kamus tersebut diambil daftar kata-kata perbaikan di dalamnya yang sesuai dengan dataset komentar vaksin Covid-19 yang digunakan pada penelitian ini. Selain itu juga dilakukan penambahan data kata-kata perbaikan lain yang belum terdapat pada Colloquial Indonesian Lexicon tadi karena belum terdapatnya perbaikan untuk kata-kata yang berkaitan dengan vaksin dan covid-19 dengan cara membaca isi dataset kemudian menambahkan daftar kata yang salah beserta kata perbaikannya ke dalam kamus dengan pedoman KBBI. Beberapa daftar kamus perbaikan kata pada penelitian ini dapat dilihat pada **Tabel 9.**

Tabel 9. Daftar Kamus Perbaikan Kata

Kata Tidak Baku	Kata Baku
tpi	tapi
klo	kalau
blm	belum
jd	jadi
mhon	mohon
sdh	sudah
dpet	dapat
kopit	covid
paksin	vaksin

Contoh proses penerapan perbaikan kata pada data ulasan terdapat pada **Tabel 10**.

Tabel 10. Contoh Perbaikan Kata

Sebelum	Sesudah
'halo', 'klo', 'm', 'laku', 'vaksin', 'bodoh'	halo kalau m laku vaksin bodoh

2.4. Pembobotan TF-IDF

Term Frequency-Inverse Document Frequency (TF-IDF) adalah cara memberikan bobot relasi pada kata (term) terhadap dokumen teks. Metode TF-IDF menggunakan dua konsep untuk menghitung bobot yaitu frekuensi kemunculan sebuah kata di dalam sebuah dokumen tertentu (Term Frequency) dan inverse frekuensi dokumen yang mengandung kata tersebut (Inverse Document Frequency) [13]. Term frequency (TF) digunakan untuk menghitung jumlah kemunculan term pada satu dokumen sedangkan inverse document frequency (IDF) digunakan untuk menghitung kemunculan term pada berbagai dokumen. Pada penelitian ini TF-IDF digunakan untuk memberikan bobot tiap kata pada dataset komentar masyarakat yang telah dilakukan preprocessing. Langkah-langkah pembobotan dengan TF-IDF dapat dijabarkan sebagai berikut [13].

1. Menghitung *term frequency* $tf_{t,d}$
2. Menghitung *weight term frequency* W_{tf} dengan persamaan 4

$$W_{tf_{t,d}} = \begin{cases} 0 \\ 1 + \log 10\, tf_{t,d} \\ if\ tf_{t,d} > 0 \end{cases} \quad (4)$$

3. Menghitung *document frequency* (df)
4. Menghitung *inverse document frequency* (idf) dengan persamaan 5

$$idf_t = log\left(\frac{D}{df_t}\right) \quad (5)$$

5. Menghitung nilai bobot TF-IDF dengan persamaan 6

$$W_{d.t} = tf_{d.t} \times idf_{d.t} \quad (6)$$

Keterangan:

ISSN: 1829-667X / E-ISSN: 2460-9021
DOI:10.31515/telematika.v19i1.6577

$tf_{t,d}$: frekuensi term

$W_{tf_{t,d}}$: bobot frekuensi term

df : jumlah frekuensi dokumen yang memiliki term

D : banyaknya dokumen

$W_{t,d}$: bobot TF-IDF

2.5. Algoritma Naïve Bayes

Untuk mendapatkan hasil dalam melakukan penelitian yaitu hasil klasifikasi sentimen dari data komentar instagram, dilakukan pengklasifikasian dengan menggunakan metode klasifikasi Naïve Bayes. Algoritma Naïve Bayes adalah metode klasifikasi yang dasarnya merupakan teorema Bayes. Teorema Bayes bertujuan untuk memperkirakan probabilitas kejadian berdasarkan kategori yang terdapat pada data latih. Algoritma ini memiliki kelebihan cocok digunakan untuk jumlah input yang besar, selain itu juga memiliki keunggulan dalam kecepatan dan kesederhanaannya. Walaupun algoritma ini sederhana namun memiliki performa yang baik dalam melakukan klasifikasi [8]. Persamaan menghitung teorema Bayes dapat dilihat pada persamaan 7 [14].

$$P(c_j|w_i) = \frac{P(c_j) \times P(w_i|c_j)}{P(w_i)} \quad (7)$$

Keterangan:

$P(c_j|w_i)$: Posterior merupakan peluang kategori j ketika terdapat kemunculan kata i
$P(w_i|c_j)$: Conditional probability merupakan peluang sebuah kata i masuk dalam kategori j
$P(c_j)$: Prior merupakan peluang kemuculan sebuah kategori j
$P(w_i)$: Peluang kemunculan sebuah kata

Pada persamaan di atas peluang kemunculan kata dapat dihilangkan karena peluang tersebut tidak berpengaruh pada perbandingan hasil klasifikasi setiap kategori. Sehingga persamaannya menjadi persamaan 8.

$$P(c_j|w_i) = P(c_j) \times P(w_i|c_j) \quad (8)$$

Nilai Prior pada persamaan di atas didapatkan dengan persamaan 9.

$$P(c_j) = \frac{N_c}{N} \quad (9)$$

Keterangan:

N_c : banyak dokumen berkategori c_j pada dokumen latih

N : jumlah keseluruhan dokumen latih yang digunakan

Sedangkan nilai Posterior didapatkan dengan mengalikan prior dengan total *conditional probability*. Persamaannya dapat dilihat pada persamaan 10.

$$P(c_j|w_i) = P(c_j) \times P(w_i|c_j) \times \ldots \times P(w_n|c_j) \quad (10)$$

2.6. Pengujian

Pengujian yang dilakukan pada penelitian ini dilakukan dengan cara membandingkan akurasi model Naïve Bayes yang dibangun sebelum dan sesudah penggunaan perbaikan kata Levenshtein Distance. Untuk menyatakan tingkat kebenaran dari proses klasifikasi yang dilakukan oleh model Naïve Bayes, maka hal yang perlu dilakukan yaitu membuat tabel *Confusion Matrix*. *Confusion matrix* adalah cara untuk mengukur performa dari model klasifikasi yang dibuat dengan menghitung *precision*, *recall*, dan *accuracy*. Ada 4 istilah yang digunakan untuk merepresentasikan hasil proses klasifikasi yaitu *True Positive* (TF), *True Negative* (TF), *False Positive* (FP), *False Negative* (FN) [15]. Pengujian terbagi menjadi dua skenario yaitu pengujian menggunakan data yang melalui proses perbaikan kata dan data yang tidak melalui proses perbaikan kata. Data yang digunakan dibagi menjadi 80% data training dan 20% data testing sehingga dari 2394 data yang digunakan pada penelitian ini maka 479 data menjadi data testing. Selain itu dilakukan pengujian dengan menggunakan data uji lain yang belum pernah dilakukan training yang berjumlah 100 data untuk mengetahui tingkat akurasi sistem dalam melakukan klasifikasi data yang belum pernah ditemui sebelumnya.

3. Hasil dan Pembahasan

3.1. Hasil Pengujian Model

Pengujian model dilakukan untuk mengetahui performa kinerja dari model yang sudah dibangun untuk penelitian analisis sentimen vaksin Covid-19. Pada pengujian ini dibagi menjadi 2 tahap yaitu pengujian yang *preprocessing* datanya melalui proses perbaikan kata Levenshtein Distance dan pengujian yang *preprocessing* datanya tidak melalui proses perbaikan kata Levenshtein Distance untuk membandingkan akurasi dari kedua tahap tersebut. Pengujian yang dilakukan menggunakan *confusion matrix*.

Tanpa Menggunakan Perbaikan Kata Levenshtein Distance

Pada tahap ini pengujian dilakukan menggunakan data yang tidak dilakukan proses perbaikan kata. Sehingga *preprocessing* yang dilakukan hanya meliputi *case folding, remove punctuation, remove number, tokenizing, stopword removal*, dan *stemming*. Pada pengujian ini data testing lama berjumlah 479 data, sementara data testing baru berjumlah 100 data. *Confusion matrix* untuk data testing lama dapat dilihat pada **Tabel 11** sedangkan *confusion matrix* untuk data testing baru dapat dilihat pada **Tabel 12**.

Tabel 11. *Confusion Matrix Data* Testing Lama Tanpa Perbaikan Kata

		Prediksi		
		Positif	**Netral**	**Negatif**
Aktual	**Positif**	150	30	28
	Netral	46	60	20
	Negatif	41	22	82

Tabel 12. *Confusion Matrix* Data Testing Baru Tanpa Perbaikan Kata

		Prediksi		
		Positif	**Netral**	**Negatif**
Aktual	**Positif**	18	2	32
	Netral	9	9	11
	Negatif	10	2	7

Dengan Menggunakan Perbaikan Kata Levenshtein Distance

Pada tahap ini pengujian dilakukan menggunakan data yang dilakukan proses perbaikan kata. Sehingga preprocessing yang dilakukan hanya meliputi case folding, remove punctuation, remove number, tokenizing, stopword removal, stemming, dan perbaikan kata. Pada pengujian ini data testing lama berjumlah 479 data, sementara data testing baru berjumlah 100 data. Confusion matrix untuk data testing lama dapat dilihat pada **Tabel 13** sedangkan confusion matrix untuk data testing baru dapat dilihat pada **Tabel 14**.

Tabel 13. *Confusion Matrix Data Testing Lama Dengan Perbaikan Kata*

		Prediksi		
		Positif	**Netral**	**Negatif**
Aktual	**Positif**	153	24	24
	Netral	35	60	14
	Negatif	30	10	129

Tabel 14. *Confusion Matrix Data Testing Baru Dengan Perbaikan Kata*

		Prediksi		
		Positif	**Netral**	**Negatif**
Aktual	**Positif**	20	0	7
	Netral	8	8	13
	Negatif	6	0	38

Berdasarkan tabel confusion matrix yang dihasilkan pada dua skenario pengujian dan tercantum pada **Tabel 11**, **Tabel 12**, **Tabel 13**, dan **Tabel 14** maka dapat dihitung nilai *accuracy*, *recall*, *precision*, dan *f1-score* dari model yang sudah dibangun. Berikut ini **Tabel 15** merupakan hasil dari pengujian yang dilakukan.

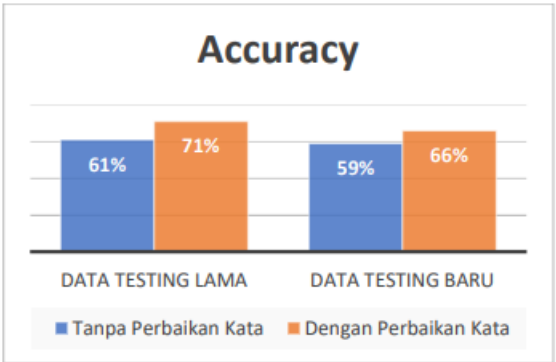
Tabel 15. Hasil Skenario Pengujian

Skenario	**Data**	**Accuracy**	**Recall**	**Precision**	**F1-Score**
Tanpa Menggunakan Perbaikan Kata	Data Testing Lama	61%	61%	61%	61%
	Data Testing Baru	59%	59%	61%	58%
Dengan Menggunakan Perbaikan Kata	Data Testing Lama	71%	71%	71%	71%
	Data Testing Baru	66%	66%	66%	63%

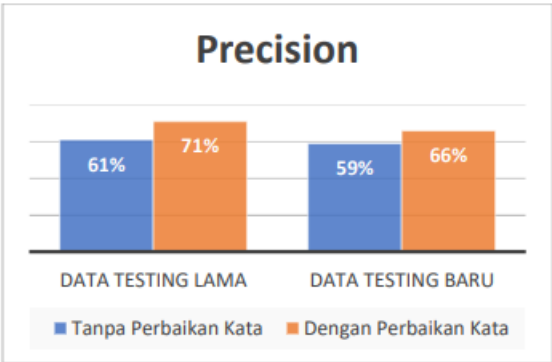
ISSN: 1829-667X / E-ISSN: 2460-9021
DOI:10.31515/telematika.v19i1.6577

3.2. Pembahasan

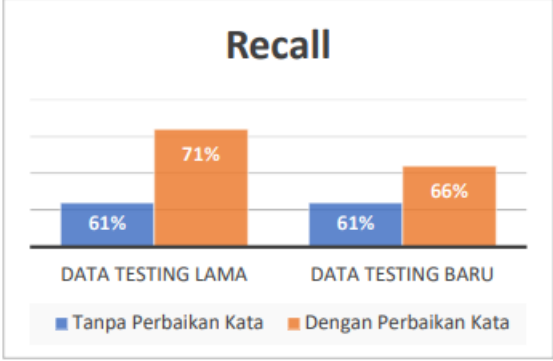
Setelah melakukan pengujian terhadap metode Naïve Bayes dengan menggunakan data testing lama dan data testing baru yang dilakukan preprocessing perbaikan kata dan tidak dilakukan preprocessing perbaikan kata, maka didapatkan hasil bahwa dengan dilakukannya preprocessing perbaikan kata pada data dapat berpengaruh dengan meningkatkan *accuracy*, *precision*, *recall*, dan *f1-score* pada metode Naïve Bayes dalam melakukan analisis sentimen vaksin Covid-19. Untuk grafiknya dapat dilihat pada **Gambar 1**, **Gambar 2**, **Gambar 3**, dan **Gambar 4**.



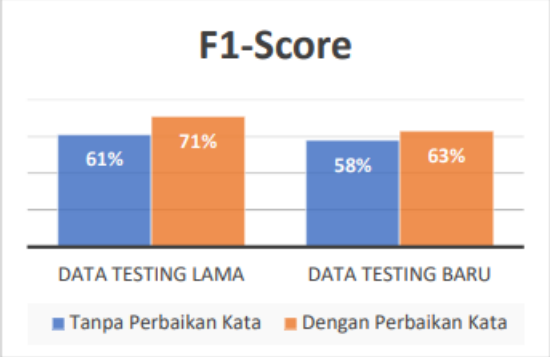
Gambar 1. Grafik Hasil Pengujian Accuracy



Gambar 2. Grafik Hasil Pengujian Precision



Gambar 3. Grafik Hasil Pengujian Recall



Gambar 4. Grafik Hasil Pengujian F1-Score

Pada data testing lama dengan data yang berjumlah 479 data dengan memperbaiki sebanyak 1158 kata pada data yang digunakan, menghasilkan nilai *accuracy*, *recall*, *precision*, dan *f1-score* yang meningkat sebesar 10% dari yang tadinya 61% menjadi 71%. Sementara untuk data testing baru dengan data yang berjumlah 100 data dengan memperbaiki sebanyak 135 kata pada data yang digunakan menghasilkan nilai *accuracy* dan *precision* yang meningkat 7% dari 59% menjadi 66%, *recall* meningkat 5% dari yang tadinya 61% menjadi 66%, dan *f1-score* meningkat sebesar 5% dari yang tadinya 58% menjadi 63%. Dengan dilakukannya perbaikan kata membuat sistem menjadi lebih mudah dalam melakukan klasifikasi komentar sehingga dapat meningkatkan akurasi. Namun untuk klasifikasi data testing baru memperoleh akurasi yang cukup rendah walaupun data yang dites hanya berjumlah 100 data, hal ini disebabkan oleh sistem yang kurang mampu dalam melakukan klasifikasi data yang belum pernah dilakukan training sebelumnya.

ISSN: 1829-667X / E-ISSN: 2460-9021
DOI:10.31515/telematika.v19i1.6577

4. Kesimpulan

Berdasarkan hasil penelitian yang telah dilakukan maka dapat ditarik kesimpulan bahwa dengan menggunakan metode perbaikan kata Levenshtein Distance pada preprocessing data dalam menyelesaikan permasalahan analisis sentimen vaksin Covid-19 dapat meningkatkan akurasi dari metode Naïve Bayes yang digunakan. Data testing lama yang tidak dilakukan preprocessing perbaikan kata memperoleh akurasi 61%, sementara untuk data yang melalui proses preprocessing perbaikan kata memperoleh akurasi sebesar 71% sehingga terjadi peningkatan sebesar 10%. Sedangkan untuk data testing baru yang berjumlah 100 data memperoleh akurasi 59% untuk data yang tidak dilakukan preprocessing perbaikan kata dan 66% untuk data yang dilakukan perbaikan kata sehingga terjadi peningkatan akurasi sebesar 7%. Namun untuk klasifikasi data testing baru memperoleh akurasi yang cukup rendah walaupun data yang dites hanya berjumlah 100 data, hal ini disebabkan oleh sistem yang kurang mampu dalam melakukan klasifikasi data baru yang belum pernah dilakukan training sebelumnya. Saran yang dapat dilakukan untuk pengembangan penelitian lebih lanjut adalah menggunakan metode machine learning lain untuk mengetahui metode mana yang lebih optimal dalam melakukan analisis sentimen vaksin Covid-19, menggunakan dataset dengan data yang seimbang sehingga dapat mengoptimalkan akurasi dari model yang dibangun karena pada penelitian ini masih menggunakan imbalance dataset, dan menambahkan daftar kata pada kamus perbaikan kata yang digunakan agar penggunaan metode Levenshtein Distance lebih optimal, selain itu sistem yang dibangun kurang mampu untuk melakukan klasifikasi terhadap data yang belum pernah dilakukan training sebelumnya sehingga membutuhkan data training yang lebih banyak.

Daftar Pustaka

[1] I. P. Sari and S. Sriwidodo, "Perkembangan Teknologi Terkini dalam Mempercepat Produksi Vaksin COVID-19," Maj. Farmasetika, vol. 5, no. 5, p. 204, 2020, doi: 10.24198/mfarmasetika.v5i5.28082.

[2] A. P. Wibawa, M. G. A. Purnama, M. F. Akbar, and F. A. Dwiyanto, "Metode-metode Klasifikasi," Pros. Semin. Ilmu Komput. dan Teknol. Inf., vol. 3, no. 1, pp. 134–138, 2018.

[3] F. F. Rachman and S. Pramana, "Analisis Sentimen Pro dan Kontra Masyarakat Indonesia tentang Vaksin COVID-19 pada Media Sosial Twitter," Heal. Inf. Manag. J. ISSN, vol. 8, no. 2, pp. 2655–9129, 2020.

[4] M. Ritonga, M. Ali, A. Ihsan, and A. Anjar, "Sentiment analysis of COVID-19 vaccine in Indonesia using Naïve Bayes Algorithm Sentiment analysis of COVID-19 vaccine in Indonesia using Naïve Bayes Algorithm," 2021, doi: 10.1088/1757-899X/1088/1/012045.

[5] M. D. Mulyawan and I. Slamet, "Analisis Sentimen Terkait Vaksin Covid-19 Pada Data Twitter Menggunakan Support Vector Machine," pp. 133–139, 2021.

[6] B. Laurensz and E. Sediyono, "Analisis Sentimen Masyarakat terhadap Tindakan Vaksinasi dalam Upaya Mengatasi Pandemi Covid-19 (Analysis of Public Sentiment on Vaccination in Efforts to Overcome the," vol. 10, no. 2, pp. 118–123, 2021.

[7] A. R. Satria and S. Adinugroho, "Analisis Sentimen Ulasan Aplikasi Mobile menggunakan Algoritma Gabungan Naïve Bayes dan C4 . 5 berbasis Normalisasi Kata Levenshtein," vol. 4, no. 11, pp. 4154–4163, 2020.

ISSN: 1829-667X / E-ISSN: 2460-9021
DOI:10.31515/telematika.v19i1.6577

[8] F. Gunawan, M. A. Fauzi, and P. P. Adikara, "Analisis Sentimen Pada Ulasan Aplikasi Mobile Menggunakan Naive Bayes dan Normalisasi Kata Berbasis Levenshtein Distance (Studi Kasus Aplikasi BCA Mobile)," Syst. Inf. Syst. Informatics J., vol. 3, no. 2, pp. 1–6, 2017, doi: 10.29080/systemic.v3i2.234.

[9] S. García, J. Luengo, and F. Herrera, "Tutorial on practical tips of the most influential data preprocessing algorithms in data mining," Knowledge-Based Syst., vol. 98, pp. 1–29, 2016, doi: 10.1016/j.knosys.2015.12.006.

[10] T. Mardiana, H. Syahreva, and T. Tuslaela, "Komparasi Metode Klasifikasi Pada Analisis Sentimen Usaha Waralaba Berdasarkan Data Twitter," J. Pilar Nusa Mandiri, vol. 15, no. 2, pp. 267–274, 2019, doi: 10.33480/pilar.v15i2.752.

[11] B. P. S. A. P. Pratama, "Analisis Kinerja Algoritma Levenshtein Distance," Logika, no. 2, pp. 131–143, 2016.

[12] N. Aliyah Salsabila, Y. Ardhito Winatmoko, A. Akbar Septiandri, and A. Jamal, "Colloquial Indonesian Lexicon," Proc. 2018 Int. Conf. Asian Lang. Process. IALP 2018, no. August 2020, pp. 226–229, 2019, doi: 10.1109/IALP.2018.8629151.

[13] R. T. Wahyuni, D. Prastiyanto, and E. Supraptono, "Penerapan Algoritma Cosine Similarity dan Pembobotan TF-IDF pada Sistem Klasifikasi Dokumen Skripsi," J. Tek. Elektro, vol. 9, no. 1, pp. 18–23, 2017, doi: 10.15294/jte.v9i1.10955.

[14] I. F. Rozi, R. Ardiansyah, and N. Rebeka, "Penerapan Normalisasi Kata Tidak Baku Menggunakan Levenshtein Distance pada Analisa Sentimen Layanan PT . KAI di Twitter," Semin. Inform. Apl., pp. 106–112, 2019.

[15] J. Han, Jiawei; Kamber, Micheline; Pei, Data mining: Data mining concepts and techniques. 200.