Coding : Jurnal Komputer dan Aplikasi
Volume 11, No. 03 (2023), hal 338-345

p-ISSN : 2338-493X
e-ISSN : 2809-574X

Analisis Sentimen Vaksin Covid-19 Pada Media Sosial Twitter Menggunakan Metode Naive Bayes

[1]Tiya Indriyani, [2]Renny Puspita Sari, [3]Ferdy Febriyanto
[1][2][3]Jurusan Sistem Informasi, Fakultas MIPA Universitas Tanjungpura
Jl. Prof. Hadari Nawawi, Pontianak
Telp./Fax.: (0561) 577963
e-mail: [1]tiyaindriyani@student.untan.ac.id, [2]rennysari@sisfo.untan.ac.id,
[3]ferdyf@sisfo.untan.ac.id

Abstrak

Coronavirus disease 2019 (Covid-19) merupakan kasus pandemik pnumonia yang ditularkan dari manusia ke manusia melalui droplet. Sejak korban pertama ditemukan, Covid-19 telah menginfeksi jutaan manusia dan ratusan ribu kasus meninggal. Berbagai upaya pencegahan penyebaran virus dilakukan, yang salah satunya adalah pemberian vaksin kepada masyarakat. Namun banyak terjadi perdebatan yang dinilai pemerintah terlalu gegabah dengan mewajibkan vaksin. Banyak opini yang bermunculan dari masyarakat yang dituangkan melalui sosial media, diantaranya twitter. Oleh karena itu, dibuat sistem yang dapat mengolah data yang sangat banyak untuk dilihat kecendrungan sentimen publik terhadap kata 'vaksin'. Metode klasifikasi data yang digunakan dalam penelitian ini adalah Naive Bayes. Sebelum pengklasifikasian data terlebih dahulu dilakukan crawling data dari twitter yang selanjutnya setiap dokumen akan diberikan label untuk bisa dilakukan preprocessing dan pembobotan TF-IDF. Berdasarkan penelitian yang dilakukan dengan menggunakan metode Naive Bayes, disimpulkan bahwa sistem dapat mengklasifikasikan polaritas sentimen kedalam 3 kategori, yaitu positif, negatif dan netral. Hasil sentimen analisis yang dilakukan pada kata vaksin menunjukan tingkat sentimen positif lebih besar dibandingkan dengan tingkat sentimen negatif. Hal ini berdasarkan hasil performa Naive Bayes yang menunjukan hasil positif lebih besar.

Kata kunci— Analisis Sentimen, Naive Bayes, Klasifikasi, Twitter

1. PENDAHULUAN

Wabah Covid-19 (coronavirus disease 2019) yang disebabkan oleh virus SARS-CoV-2 menjadi pandemi baru yang menginfeksi jutaan manusia di berbagai belahan dunia. Virus yang menyebar secara massive melalui droplet ini tentu saja membawa perubahan yang sangat besar ditengah masyarakat. Berdasarkan Kementrian Kesehatan Republik Indonesia, dari data kasus pertama Covid-19 di Indonesia pada Maret 2020 hingga September 2021 ini terdapat lebih dari 4 juta kasus terkonfirmasi positif dan lebih dari 140 ribu kasus meninggal. Berbagai upaya dilakukan pemerintah untuk mencegah penyebaran virus yang berdampak pada berbagai aspek kehidupan sosial masyarakat. Salah satunya adalah pemberian vaksin kepada masyarakat. Meskipun vaksinasi merupakan salah satu usaha pemerintah dalam menanggulangi pandemi namun tidak lantas mendapat respon yang baik dari masyarakat. Banyak perdebatan yang terjadi diluar sana mengenai topik vaksinasi.

Pemerintah yang dinilai terlalu terburu-buru hingga menyebabkan vaksin yang digunakan diragukan kelayakannya dan resiko setelah vaksin. Berita inilah yang banyak menimbulkan opini dari masyarakat. Opini ini biasanya disampaikan secara langsung maupun melalui media sosial, salah satunya Twitter. Dengan pengguna yang sangat banyak tentunya banyak sekali pertukaran informasi di twitter dan banyak pula informasi yang dapat digali terutama dari data teks yang di posting oleh masyarakat. Opini melalui tweet ini dapat dimanfaatkan untuk melihat bagaimana

p-ISSN : 2338-493X
e-ISSN : 2809-574X

sentimen publik terhadap vaksinasi untuk menanggulangi pandemi Covid-19 di Indonesia. Sentimen publik sendiri dapat dibagi menjadi 3, yaitu sentimen positif, negatif dan netral.

Banyaknya opini yang disampaikan tentu saja dibutuhkan waktu dan usaha untuk mengelompokkan sentimen tersebut. Oleh karena itu penelitian ini menganalisis sentimen publik dari cuitan pengguna Twitter yang divisualisasikan melalui website dengan menerapkan metode klasifikasi Naive Bayes Classifier. Naïve Bayes adalah sebuah metode klasifikasi yang menggunakan perhitungan probabilitas. Naïve Bayes memiliki kesederhanaan dalam implementasi dan efisiensi kebutuhan sumber daya komputasi di banding dengan metode lain [1].

2. LANDASAN TEORI

2.1 Data Mining

Data mining adalah proses yang memperkerjakan satu atau lebih teknik pembelajaran komputer (machine learning) untuk menganalisis dan mengekstraksi pengetahuan (knowledge) secara otomatis. Defiisi lain diantaranya adalah pembelajaran berbasis induksi (induction-based learning) adalah proses pembentukan definisi-definisi konsep umum yang dilakukan dengan cara mengobservasi contoh-contoh spesifik dari konsep-konsep yang akan dipelajari.

Data mining berisi pencarian trend atau pola yang diinginkan dalam database besar untuk membantu pengambilan keputusan di waktu yang akan datang. Pola-pola ini dikenali oleh perangkat tertentu yang dapat memberikan suatu analisa data yang berguna dan berwawasan yang kemudian dapat dipelajari dengan lebih teliti, yang mungkin saja menggunakan perangkat pendukung keputusan yang lainnya .

2.2 Text Mining

Text mining merupakan salah satu cabang ilmu data mining yang menganalisis data berupa dokumen teks[2]. Text mining adalah satu langkah dari analisis teks yang dilakukan secara otomatis oleh komputer untuk menggali informasi yang berkualitas dari suatu rangkaian teks yang terangkum dalam sebuah dokumen. Ide awal pembuatan text mining adalah untuk menemukan pola-pola informasi yang dapat digali dari suatu teks yang tidak terstruktur. Langkah awal sebelum suatu data teks dianalisis menggunakan metode-metode dalam text mining adalah melakukan preprocessing teks. Selanjutnya, setelah didapatkan data yang siap diolah, analisis text mining dapat dilakukan.

Tahap-tahap preprocessing secara umum adalah sebagai berikut [3]:

2.2.1 Case Folding

toLowerCase (*Case Folding*), yaitu mengubah semua karakter huruf menjadi huruf kecil.

2.2.2 Tokenizing

Tokenizing yaitu proses penguraian deskripsi yang semula berupa kalimat-kalimat menjadi kata-kata dan menghilangkan delimiter-delimiter seperti tanda titik (.), koma (,), spasi dan karakter angka yang ada pada kata tersebut.

2.2.3 Stopword Removal

Stopword adalah kosakata yang bukan merupakan ciri (kata unik) dari suatu dokumen. Misalnya “di”, “oleh”, “pada”, “sebuah”, “karena” dan lain sebagainya. Sebelum proses *stopword removal* dilakukan, harus dibuat daftar *stopword* (*stoplist*). Jika termasuk di dalam stoplist maka kata-kata tersebut akan dihapus dari deskripsi sehingga kata-kata yang tersisa di dalam deskripsi dianggap sebagai kata-kata yang mencirikan isi dari suatu dokumen atau keywords.

2.2.4 Stemming

Stemming adalah proses pemetaan dan penguraian berbagai bentuk (*variants*) dari suatu kata menjadi bentuk kata dasarnya (*stem*). Tujuan dari proses *stemming* adalah menghilangkan imbuhan-imbuhan baik itu berupa prefiks, sufiks, maupun konfiks yang ada pada setiap kata. Jika imbuhan tersebut tidak dihilangkan maka setiap satu kata dasar akan disimpan dengan berbagai macam bentuk yang berbeda sesuai dengan imbuhan yang melekatinya sehingga hal tersebut akan menambah beban *database.*

p-ISSN : 2338-493X
e-ISSN : 2809-574X

2.3 Analisis Sentimen

Analisis sentimen adalah suatu bidang yang berlangsung dalam penelitian berbasiskan teks. Analisis sentimen atau opinion mining adalah kajian tetang cara untuk memecahkan masalah dari opini masyarakat, sikap dan emosi suatu entitas, dimana entitas tersebut dapat mewakili individu.

Analisis sentimen atau opinion mining merupakan proses memahami, mengekstrak dan mengolah data tekstual secara otomatis untuk mendapatkan informasi sentimen yang terkandung dalam suatu kalimat opini. Analisis sentimen dilakukan untuk melihat pendapat atau kecendrungan opini terhadap sebuah masalah atau objek oleh seseorang, apakah cenderung beropini negatif atau positif [4].

2.4 Pembobotan Kata

Term weighting atau pembobotan term sangat dipengaruhi oleh hal-hal berikut ini [5]:

1) Term Frequency (TF) factor, yaitu faktor yang menentukan bobot term pada suatu dokumen berdasarkan jumlah kemunculannya dalam dokumen tersebut. Nilai jumlah kemunculan suatu kata (term frequency) diperhitungkan dalam pemberian bobot terhadap suatu kata. Semakin besar jumlah kemunculan suatu term (tf tinggi) dalam dokumen, semakin besar pula bobotnya dalam dokumen atau akan memberikan nilai kesesuian yang semakin besar.
2) Inverse Document Frequency (IDF) faktor, yaitu pengurangan dominansi term yang sering muncul di berbagai dokumen. Hal ini diperlukan karena term yang banyak muncul di berbagai dokumen, dapat dianggap sebagai term umum (common term) sehingga tidak penting nilainya. Sebaliknya faktor kejarangmunculan kata (term scarcity) dalam koleksi dokumen harus diperhatikan dalam pemberian bobot. Kata yang muncul pada sedikit dokumen harus dipandang sebagai kata yang lebih penting (uncommon terms) daripada kata yang muncul pada banyak dokumen. Pembobotan akan memperhitungkan faktor kebalikan frekuensi dokumen yang mengandung suatu kata Inverse Document Frequency.

Metode TF-IDF merupakan metode pembobotan term yang banyak digunakan sebagai metode pembanding terhadap metode pembobotan baru. Pada metode ini, perhitungan bobot term t dalam sebuah dokumen dilakukan dengan mengalikan nilai Term Frequency dengan Inverse Document Frequency (Kurniasari, 2018).

Inverse Document Frequency (IDF) dihitung dengan menggunakan persamaan 1 berikut.

$$idf_j = \log\left(\frac{D}{df_j}\right) + 1 \quad (1)$$

Dimana:
D : adalah jumlah semua dokumen dalam koleksi
Df_i : adalah jumlah dokumen yang mengandung term t_j

Dengan demikian rumus umum untuk TF-IDF adalah penggabungan dari formula perhitungan raw TF dan formula IDF dengan cara mengalikan nilai *Term Frequency* (TF) dengan nilai *Inverse Document Frequency*(idf).

$$w_{ij} = tf_{ij} \times log\left(\frac{D}{df_j}\right) \quad (2)$$

Keterangan:
w_{ij}: adalah bobot *term* terhadap dokumen d_i
tf_{ij}: adalah jumlah kemunculan *term* t_j dalam dokumen d_i
D: adalah jumlah semua dokumen yang ada dalam *database*
df_i: adalah jumlah dokumen yang mengandung *term* t_j (minimal ada satu kata yaitu *term* t_j)

2.5 Klasifikasi

Klasifikasi adalah menentukan sebuah *record* data baru ke salah satu dari beberapa kategori atau kelas[6]. Tujuan klasifikasi adalah untuk memperkirakan kelas dari suatu objek yang labelnya tidak diketahui. Proses klasifikasi biasanya dibagi menjadi dua fase yaitu fase *learning* dan fase *testing*. Pada fase *learning*, sebagian data yang telah diketahui kelas datanya diumpankan untuk membentuk model perkiraan. Kemudian pada fase test model yang sudah terbentuk diuji dengan sebagian data lainnya untuk mengetahui akurasi dari model tersebut. Bila akurasinya mencukupi model ini

p-ISSN : 2338-493X
e-ISSN : 2809-574X

dapat dipakai untuk prediksi kelas data yang belum diketahui .

2.6 Naive Bayes Classifier

Naïve Bayes Classifier merupakan salah satu metode *machine learning* yang memanfaatkan perhitungan probabilitas dan statistik yang dikemukakan oleh ilmuwan Inggris Thomas Bayes, yaitu memprediksi probabilitas di masa depan berdasarkan pengalaman di masa sebelumnya (Aswendy, 2017).

Naïve Bayes Classifier atau bisa disebut sebagai *Multinomial Naïve Bayes* merupakan model penyederhanaan dari metode *Bayes* yang cocok dalam pengklasifikasian teks atau dokumen. Persamaan 3, 4, dan 5 menjelaskan hal tersebut.

$$V_{MAP} = arg\,max\, P\left(v_j|a_1, a_2, \dots, a_n\right) \quad (3)$$

$$V_{MAP} = \underset{v_j \in V}{arg\,max} \frac{P(a_1,a_2,\dots,a_n|v_j)P(v_j)}{P(a_1,a_2,\dots,a_n)} \quad (4)$$

$P(a_1, a_2, \dots, a_n)$ konstan sehingga dapat dihilangkan menjadi:

$$V_{MAP} = \underset{v_j \in V}{arg\,max}\, P(a_1, a_2, \dots, a_n|V_j)P(v_j) \quad (5)$$

Karena $P(a_1, a_2, \dots, a_n|v_j)$ sulit untuk dihitung, maka akan diasumsikan bahwa setiap kata pada dokumen tidak mempunyai keterkaitan.

$$V_{MAP} = \underset{v_j \in V}{arg\,max}\, P(v_j) \prod_i P\left(a_i|v_j\right) \quad (6)$$

Keterangan:

$$P(v_j) = \frac{|docs_j|}{|contoh|} \quad (7)$$

$$P(w_k|v_j) = \frac{n_k+1}{n+|kosakata|} \quad (8)$$

Di mana untuk:

- $P(v_j)$: Probabilitas setiap dokumen terhadap sekumpulan dokumen.
- $P(w_k|v_j)$: Probabilitas kemunculan kata w_k pada suatu dokumen dengan kategori *class* v_j.
- $|docs|$: Frekuensi dokumen pada setiap kategori.
- $|contoh|$: Jumlah dokumen yang ada.
- n_k: Frekuensi kata ke-k pada setiap kategori.
- $|kosakata|$: Jumlah kata pada dokumen *test*.

Pada persamaan (2.8) terdapat suatu penambahan 1 pada pembilang, hal ini dilakukan untuk mengantisipasi jika terdapat suatu kata pada dokumen uji yang tidak ada pada setiap dokumen data *training*.

2.7 Framework Django

Python adalah bahasa pemrograman yang mudah dipelajari dan *powerfull*. *Python* memiliki struktur data tingkat tinggi yang efisien dan pendekatan yang sederhana namun efektif untuk pemrograman berorientasi objek. *Sintaks Python* yang elegan dan dinamis, dengan sifatnya yang ter-*interpreted*, menjadikannya bahasa yang ideal untuk pembuatan *scripts* dan pengembangan aplikasi yang cepat di banyak area pada sebagian besar *platform* (python.org, 2021).

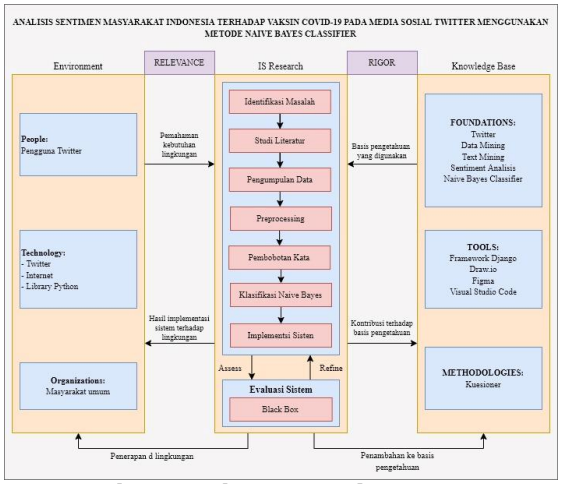
Django adalah kerangka kerja web atau *web framework python open source* tingkat tinggi yang mendorong pengembangan cepat dan desain yang bersih. Dibangun oleh pengembang berpengalaman, *django* menangani banyak kerumitan dalam pengembangan web, sehingga pengguna dapat lebih fokus penulisan aplikasi (djangoproject.com, 2021).

3. METODE PENELITIAN

Kerangka penelitian yang digunakan pada penelitian ini adalah kerangka kerja *IS Research* [7]. Kerangka kerja ini memiliki tahap-tahap analisis yang dilakukan berdasarkan pada konsep-konsep yang meliputi tahap penetapan perspektif pada aspek lingkungan yang terdiri dari tempat studi kasus, tahap penyusunan kumpulan dasar pengetahuan, dan tahap pengembangan sistem. Kerangka kerja *IS Research* dapat dilihat pada Gambar 1 berikut. Implementasi analisis sentimen ini dilakukan dengan beberapa tahapan yang dimulai dari *crawling* data dari API *Twitter*, membersihkan data dengan beberapa tahap *preprocessing*,

p-ISSN : 2338-493X
e-ISSN : 2809-574X

menghitung bobot kata, sehingga mendapatkan hasil analisis sentimen.



Gambar 1. Tahapan Penelitian

4. HASIL DAN PEMBAHASAN

Adapun tahapan dari proses klasifikasi adalah sebagai berikut.

4.1 *Crawling* Data

Proses paling awal yang dilakukan dalam penelitian ini adalah pengumpulan data atau *crawling* data. Proses pengumpulan data ini dilakukan dengan cara web *crawling*. Data *tweet* pengguna diambil dengan menggunakan API *Twitter* dengan menggunakan *library tweepy*. Setelah data *tweet* yang dibutuhkan terpenuhi data diekspor dalam bentuk file csv. atau *excel.*

4.2 *Labelling* Data Latih

Proses selanjutnya yang dilakukan setelah data telah terkumpul adalah dilakukan pemilihan data latih yang kemudian diberikan label berdasarkan polaritas positif, negatif dan netral. Proses pelabelan data latih ini dilakukan secara manual. Setelah selesai pemberian label pada data latih data disimpan dengan format .xlsx. Berikut merupakan contoh pelabelan data latih yang dapat dilihat pada Tabel 1.

Tabel 1 *Labelling* Data Latih

No.	Tweet	Label
1	#HentikanPaksaVaksin Vaksin adalah hak, bukan kewajiban!!!	Negatif
2	Itu vaksin apa fotokopi sampe bolak balik	Netral
3	Tidak Perlu Takut Vaksin, Vaksin Aman dan Halal #Vaksinyuk	Positif

4.3 *Preprocessing*

Data yang telah disimpan selanjutnya diimport ke sistem Analisis Sentimen Vaksin Covid-19 untuk melakukan proses preprocessing sebagai langkah mempersiapkan data untuk dibersihkan agar tidak mengganggu pada saat proses klasifikasi. Berikut merupakan penjelasan proses preprocessing:

4.3.1 Case folding

Case folding merupakan langkah pertama dalam proses *preprocessing*. *Case folding* merupakan proses mengkonversi teks yang awalnya huruf kapital menjadi huruf kecil. Contoh hasil dari proses *case folding* dari data Tabel 1 dapat dilihat pada Tabel 2 berikut ini.

***Tabel 1** Hasil Proses* Case Folding

No.	Tweet	Label
1	#hentikanpaksavaksin vaksin adalah hak, bukan kewajiban!!!	Negatif
2	itu vaksin apa fotokopi sampe bolak balik	Netral
3	tidak perlu takut vaksin, vaksin aman dan halal #vaksinyuk	Positif

4.3.2 Tokenizing

Setelah melakukan proses case folding, berikutnya yang dilakukan adalah tokenizing yang merupakan memisahkan kalimat menjadi kata per kata yang terpisah dengan spasi yang dihimpun pada array. Contoh hasil tokenizing dari berdasarkan hasil case folding dari Tabel 2 dapat dilihat pada Tabel 3 berikut ini.

Tabel 2 Hasil Proses Tokenizing

No.	Tweet	Label
1.	vaksin adalah hak bukan kewajiban	Negatif
2.	itu vaksin apa fotokopi sampe bolak balik	Netral
3.	tidak perlu takut vaksin vaksin aman dan halal	Positif

p-ISSN : 2338-493X
e-ISSN : 2809-574X

4.3.3 Stopword Removal

Setelah melakukan proses *tokenizing*, tahap selanjutnya yang dilakukan adalah *Stopword Removal* yang merupakan proses menghilangkan kata yang tidak diperlukan dalam proses klasifikasi. Kata-kata yang terdapat pada tweet akan dibandingkan dengan kata yang terdapat pada library NLTK *corpus*. Secara otomatis sistem akan menghapus kata yang terdeteksi sama. Contoh hasil *stopword removal* dari berdasarkan hasil *tokenizing* dari Tabel 3 dapat dilihat pada Tabel 4 berikut.

Tabel 5.3 Hasil Proses *Stopword Removal*

No.	Tweet	Label
1.	vaksin adalah hak bukan kewajiban	Negatif
2.	itu vaksin apa fotokopi sampe bolak balik	Netral
3.	tidak perlu takut vaksin vaksin aman halal	Positif

4.3.4 Stemming

Stemming merupakan peoses yang dilakukan setelah proses stopword yang digunakan untuk mengeliminasi imbuhan awal maupun akhir untuk menemukan kata dasar dengan menggunakan *library* Sastrawi. Contoh hasil *stemming* dari berdasarkan hasil *stopword removal* dari Tabel 4 dapat dilihat pada Tabel 5 berikut ini.

Tabel 4 Hasil Proses *Stemming*

No.	Tweet	Label
1.	vaksin hak bukan wajib	Negatif
2.	itu vaksin apa fotokopi sampe bolak balik	Netral
3.	tidak perlu takut vaksin vaksin aman halal	Positif

4.4 Pembobotan Kata

Setelah data melewati proses *preprocessing* makan data akan disimpan ke daalam *database* yang selanjutnya akan dilakukan pembobotan. Proses pembobotan dilakukan menggunakan algoritma TF-IDF. Algoritma ini akan menghitung nilai banyaknya *term* (kata) pada suatu dokumen yang bersangkutan. Pembobotan dilakukan menggunakan library *python Sklearn*. Tahap pertama dalam pembobotan kata untuk mendapatkan nilai TF-IDF adalah dengan menghitung frekuensi kata dalam suatu dokumen.

Kata	IDF
vaksin	0,301
hak	0,845
wajib	0,845
milik	0,845
riwayat	0,845
sakit	0,845
kencing	0,845
manis	0,845
dosis	0,845
parah	0,845
tinggal	0,845
dunia	0,845
anti	0,845
paksa	0,845
jadi	0,845

Kata	IDF
syarat	0,845
giat	0,845
fotokopi	0,845
bolak	0,845
takut	0,845
aman	0,845
halal	0,845
vaksinasi	0,845
selamat	0,845
keluarga	0,845
lingkung	0,845
bangsa	0,845
pulih	0,845
indonesia	0,845

Gambar 2 Nilai IDF

4.4 Klasifikasi Metode Naïve Bayes

Klasifikasi dengan metode *Naive Bayes* diperlukan bobot pada setiap katanya yang terdapat dalam data latih. Pembobotan ini dilakukan dengan menggunakan *library textlob*. Implementasi klasifikasi metode *Naive Bayes* dapat dilihat pada kode program 1.

```
cl = NaiveBayesClassifier(train_set)
akurasi = cl.accuracy(dataset)
from textblob import TextBlob
data_tweet = list(data['tweet_clean'])
polaritas = 0
status = []
total_positif = total_negatif = total_netral =
total = 0
for i, tweet in enumerate(data_tweet):
analysis = TextBlob(tweet, classifier=cl)
if analysis.classify() == 'Positif':
total_positif += 1
elif analysis.classify() == 'Netral':
total_netral += 1
else:
total_negatif += 1
status.append(analysis.classify())
total += 1
status = pd.DataFrame({'klasifikasi_bayes':
status})
data['klasifikasi_bayes'] = status
```

Kode Program 1. Proses Klasifikasi Naïve Bayes.

Menentukan klasifikasi sentimen dilakukan dengan menghitung probabilitas data uji dengan merujuk pada probabalitas kata pada data latih. Jika bobot probabilitas positif lebih

p-ISSN : 2338-493X
e-ISSN : 2809-574X

besar dari bobot negatif dan netral maka hasil sentimen adalah positif, jika bobot probabilitas netral lebih besar dari bobot positif dan negatif maka hasil sentimen adalah netral dan jika bobot probabilitas negatif lebih besar dari bobot positif dan netral makan hasil sentimen adalah negatif. Setelah mendapatkan hasil klasifikasi maka data akan disimpan ke dalam *database* untuk selanjutnya dapat divisualisasikan kedalam bentuk grafik.

Berikut merupakan contoh data uji "vaksin aman halal kinerja nyata" untuk ditentukan polaritasnya yang diasumsikan telah melakukan *preprocessing* sebagai berikut:

1. Menghitung *prior* setiap kelas pada data latih.

   $P(vpositif)$=26=0,333
   $P(vnegatif)$=26=0,333
   $P(vnetral)$=26=0,333

2. Menghitung probabilitas *likelihood* setiap term dari semua dokumen. Jumlah seluruh kata 29, 14 *term* dari kelas positif, 14 *term* dari kelas negatif dan 10 term dari kelas netral. Banyaknya term tergantung pada hasil preproses data. Menghitung probabilitas likelihood. Contoh probabilitas kata "aman" pada kelas positif, negatif dan netral sebagai berikut:

   $P(aman|vpositif)$=1+114+29=0,047
   $P(aman|vnegatif)$=0+114+29=0,023
   $P(aman|vnetral)$=0+110+29=0,026

Berikut merupakan daftar probabilitas setiap data latih yang dapat dilihat pada Gambar 3 terhadap kelas positif, negatif dan netral.

No	Kata	TF			Probabilitas Likelihood		
		Positif	Negatif	Netral	Positif	Negatif	Netral
1	vaksin	3	2	3	0,093	0,07	0,103
2	hak	0	1	0	0,023	0,047	0,026
3	wajib	0	1	0	0,023	0,047	0,026
4	milik	0	1	0	0,023	0,047	0,026
5	riwayat	0	1	0	0,023	0,047	0,026
6	sakit	0	2	0	0,023	0,07	0,026
7	kencing	0	1	0	0,023	0,047	0,026
8	manis	0	1	0	0,023	0,047	0,026
9	dosis	0	1	0	0,023	0,047	0,026
10	parah	0	1	0	0,023	0,047	0,026
11	tinggal	0	1	0	0,023	0,047	0,026
12	dunia	0	1	0	0,023	0,047	0,026
13	anti	0	0	1	0,023	0,023	0,051

Gambar 3. Probabilitas Likelihood.

5. KESIMPULAN

Sistem Analisis Seentimen Vaksin dapat mengklasifikasikan kelas sentimen opini publik terhadap vaksin Covid-19 menggunakan metode *Naive Bayes* pada media sosial twitter. Hal tersebut didapat setelah melalui tahap crawling data, pelabelan data, tahap *pre-processing: Casefolding*, *Tokenizing*, *Stopword* dan *Stemming*, serta melakukan pembobotan TF-IDF. Hasil tweet yang telah diklasifikasikan oleh sistem diimplementasikan dalam bentuk web sehingga bisa memudahkan pengguna untuk mengakses informasi. Secara keseluruhan, masih terdapat kekurangan pada sistem yaitu sistem masih belum mampu memberikan hasil klasifikasi sentimen publik dengan tingkat akurasi yang tinggi. Hal ini tejadi karena kualitas data hasil preprocessing yang tidak terlalu baik. Yang mana hal ini dapat terjadi karena dataset yang didapat tidak bersih. Ini disebabkan karena bahasa yang publik gunakan bukan bahasa yang formal. Tingkat akurasi data yang didapat: 0.856, jumlah positif: 241, jumlah negatif: 182, jumlah netral: 77 dari semua jumlah data 500 data.

DAFTAR PUSTAKA

[1] A. A. Amer and H. I. Abdalla, "A set theory based similarity measure for text clustering and classification," *J. Big Data*, vol. 7, no. 1, p. 74, Dec. 2020.

[2] I. C. Chang, T. K. Yu, Y. J. Chang, and T. Y. Yu, "Applying text mining, clustering analysis, and latent dirichlet allocation techniques for topic classification of environmental education journals," *Sustain.*, vol. 13, no. 19, 2021.

[3] N. Mutiah, D. Prawira, and I. Rusi, "Topic Modeling on Covid-19 Vaccination in Indonesia Using LDA Model," in *2022 Seventh International Conference on Informatics and Computing (ICIC)*, Dec. 2022, pp. 1–6.

[4] M. Rani, D. Prawira, and N. Mutiah, "Analisis Sentimen Terhadap Vaksin COVID-19 Menggunakan Naive Bayes Classifier , Support Vector Machine dan K-Nearest Neighbors Sentiment Analysis on COVID-19 Vaccine using

p-ISSN : 2338-493X
e-ISSN : 2809-574X

Naive Bayes Classifier , Support Vector Machine and K-Nearest Neighbors," *J. Comput. Eng. Syst. Sci.*, vol. 8, no. January, pp. 1–11, 2023.

[5] R. Melita, V. Amrizal, H. B. Suseno, and T. Dirjam, "Penerapan Metode Term Frequency Inverse Document Frequency (Tf-Idf) Dan Cosine Similarity Pada Sistem Temu Kembali Informasi Untuk Mengetahui Syarah Hadits Berbasis Web (Studi Kasus: Hadits Shahih Bukhari-Muslim)," *J. Tek. Inform.*, vol. 11, no. 2, pp. 149–164, Nov. 2018.

[6] W. K. Sari, D. P. Rini, and R. F. Malik, "Text Classification Using Long Short-Term Memory With GloVe Features," *J. Ilm. Tek. Elektro Komput. dan Inform.*, vol. 5, no. 2, p. 85, Feb. 2020.

[7] A. Hevner, S. March, P. Jinsoo, and R. Sudha, "Design Science in Information Systems Research," *MIS Q.*, vol. 28, pp. 75–105, 2004.