ISSN (Printed) : 2579-7271
ISSN (Online) : 2579-5406

Klasifikasi Sentimen Masyarakat terhadap Kebijakan Vaksin Covid-19 pada Twitter dengan *Imbalance Classes* Menggunakan *Naive Bayes*

Prima Yohana[1], Surya Agustian*[2], Siska Kurnia Gusti[3]
[1,2,3] Teknik Informatika, Fakultas Sains dan Teknologi, Universitas Islam Negeri Sultan Syarif Kasim Riau
Email: { [1]11751202168@students.; [2]surya.agustian@; [3]siskakurniagusti@ } uin-suska.ac.id

Abstrak

Penggunaan media sosial berkembang sangat pesat hingga sebuah informasi dalam bentuk apapun bisa viral (tersebar luas) dalam sekejap saja. Hal ini dikarenakan kebanyakan masyarakat telah memiliki telepon genggam baik dari usia anak-anak hingga dewasa. Masyarakat menggunakan media sosial twitter untuk berbagai kepentingan, antara lain memberi opini dan komentar. Terkait hal tersebut, dukungan dan penolakan juga banyak disampaikan dalam menanggapi program pemerintah untuk menangani pandemi COVID-19 (*corona virus disease* 2019) dengan mengadakan vaksinasi massal. Penelitian melakukan analisis dan klasifikasi adanya sentimen yang menggambarkan pandangan yang bersifat positif, negatif maupun netral masyarakat tentang covid-19 dengan menggunakan metode *Naïve Bayes Classfier*. Analisis dilakukan dengan mencari komposisi dataset yang relatif berimbang di antara kelas positif, negatif dan netral. Kombinasi tahapan teks *preprocessing* diselidiki untuk menghasilkan model NB yang memiliki performa terbaik dari data *training*, dan divalidasi menggunakan data *development*. Model final yang dipilih, menghasilkan akurasi 69,56% pada data *development*, kemudian diterapkan untuk menguji data *testing* yang belum pernah terlihat sebelumnya. Hasil akurasi yang diperoleh adalah 61% dengan *F1-score* sebesar 0,57. Pendekatan yang digunakan telah berhasil meningkatkan performa klasifikasi, karena berhasil mengidentifikasi kelas negatif dan positif dengan lebih baik, dibandingkan bila data digunakan apa adanya, tanpa melakukan *balancing*.

Kata kunci: *analisis sentimen, kelas tak seimbang, pengklasifikasi naive bayes, seleksi fitur.*

Abstract

The use of social media is rapidly growing so that any information in any form can be viral (widely spread) in an instant. This is because most people already have mobile phones from the age of children to adults. People use Twitter social media for various purposes, including giving opinions and comments. In this regard, many supports and refusals have also been conveyed in response to the government's program to deal with the COVID-19 pandemic (corona virus disease 2019) by holding mass vaccinations. The research analyzes and classifies sentiments that describe positive, negative and neutral views of the community about COVID-19 using the Naïve Bayes Classifier method. The analysis was carried out by looking for a relatively balanced composition of the dataset among the positive, negative and neutral classes. The combination of preprocessing text stages was investigated to produce an NB model that has the best performance from training data, and validated using development data. The final model chosen, resulting in an accuracy of 69.56% on the development data, was then applied to test data testing that had never been seen before. The accuracy obtained are 61% with an F1-score of 0.57. The approach used has succeeded in improving the classification performance, because it has succeeded in identifying negative and positive classes better, compared to when the data is used as is, without balancing.

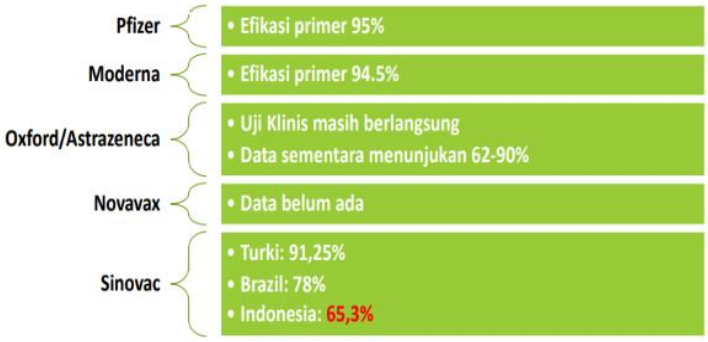
Keywords*: analisis sentimen, feature selection, imbalance classes, naive bayes classifier.*

ISSN (Printed) : 2579-7271
ISSN (Online) : 2579-5406

1. Pendahuluan

Corona Virus Disease 2019 yang sering disebut dengan COVID-19 merupakan virus yang bisa menyerang organ pernapasan tubuh manusia, yang dapat menyebabkan kesulitan pernapasan, demam, batuk, sampai menyebabkan kematian. Covid-19 merupakan penyakit menular yang berpotensi menimbulkan kedaruratan kesehatan masyarakat [1]. Covid-19 pertama kali dideteksi di Indonesia pada tanggal 2 Maret 2020. Berdasarkan data dari *Official Covid-19 Information* pertanggal 22 Maret 2021 pasien di indonesia yang telah terkonfirmasi sebanyak 1,62 juta kasus. Untuk mengurangi dampak negatif virus ini, pemerintah mengeluarkan Peratuan Menteri Kesehatan Republik Indonesia No. 84 Tahun 2020 yaitu pelaksanaan vaksinasi dalam rangka penang-gulangan pandemi corona virus disease 2019 (COVID-19).

Keputusan Menkes Nomor HK.01.07/Menkes/12758/2020 yang telah ditanda tangani pada 28 Desember 2020 menetapkan tujuh vaksin yang akan digunakan dalam vaksinasi diindonesia. Adapun beberapa vaksin-vaksin tersebut yaitu Sinovac, AstraZeneca, Sinopharm, Moderna, Novavax, Pfizer, dan Vaksin Merah Putih[1]. Setiap vaksin wajib melalui beberapa tahap penyeleksian di Indonesia, dengan kriteria harus efektif, aman dan halal. Vaksin covid-19 dari segi efikasi, dapat dilihat pada Gambar 1 berikut:



Gambar 1. Infografis tingkat evikasi vaksin yang beredar di dunia[2]

Menurut WHO, syarat efikasi sebuah vaksin yang layak diedarkan adalah diatas 50% [2]. Sebagai contoh vaksin sinovac memiliki efekasi 65,3% artinya orang yang mendapatkan suntikan vaksin sinovac memiliki resiko terinfeksi covid-19 berkurang 65,3% di bandingkan orang yang tidak di vaksin. Itu artinya masih ada 34,7% pada vaksin sinovac yang tidak mampu melindungi tubuh terhadap infeksi covid-19. Kedua, keamanan vaksin sinovac dilihat dari uji klinis tahap akhir dan sudah izinkan untuk diedarkan oleh BPOM RI[3]. Ketiga, adapun kehalalan vaksin sinovac dibuktikan dengan adanya sertifikat halal yang diterbitkan oleh Majelis Ulama Indonesia yang dapat dilihat pada fatwa Majelis Ulama Indonesia Nomor 02 Tahun 2021 tentang "Produk Vaksin Covid-19 dari Sinovac Life Sciences Co.Ltd China dan PT. Bio Farma (Persero)".

Namun demikian, masih banyak masyarakat yang tetap anti untuk divaksin, disebabkan oleh berita adanya efek samping dapat ditimbulkan dari vaksin Covid-19, seperti alergi, badan lemas, nyeri otot dan sebagainya. Sebenarnya, efek samping tersebut tidak dirasakan dalam jangka waktu yang panjang, rata-rata 3-5 hari sudah hilang. Bahkan Sebagian besar tidak meraskaan efek samping sama sekali untuk vaksin *Sinovac* yang dipakai pada saat pelaksanaan program vaksin pertama kali. Menurut SMF Pulmonologi dan Kedokteran Respirasi Laboratorium Ilmu Penyakit Dalam Fakultas Kedokteran Universitas Mulawarman dan Rumah Sakit Umum A.W Sjahranie Samarinda, alasan masyarakat di indonesia menolak vaksin ini dikarena oleh beberapa hal, sebagaimana diterangkan pada Gambar 2.

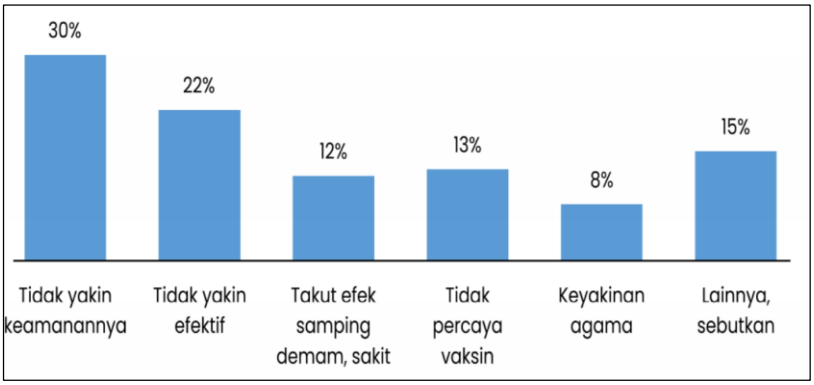
Pada saat program vaksinasi massal covid-19 disampaikan ke publik oleh pemerintah, topik ini menjadi viral dan mengisi tren topik teratas (*trending topic*) di twitter. Dukungan dan penolakan dari masyarakat disampaikan melalui twitter, dapat diklasifikasikan sebagai sentiment positif, maupun negatif. Bagi pemerintah, analisis terhadap sentimen masyarakat dapat menjadi factor pendukung yang dipertimbangkan untuk pengambilan kebijakan selanjutnya.

[1] https://www.halodoc.com/artikel/ketahui-berbagai-jenis-vaksin-yang-digunakan-di-indonesia

[2] https://lp2m.unmul.ac.id/webadmin/public/upload/files/9584b64517cfe308eb6b115847cbe8e7.pdf

[3] https://www.pom.go.id/new/view/more/berita/20883/Badan-POM-Terbitkan-EUA--Vaksin-CoronaVac-Sinovac-Siap-Disuntikkan.html

ISSN (Printed) : 2579-7271
ISSN (Online) : 2579-5406



Gambar 2. Persentase alasan menolak vaksin covid-19 di Indonesia[4]

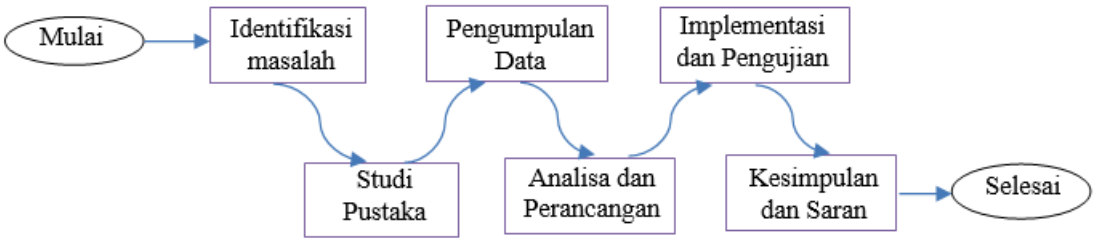
Analisis sentimen pada dasarnya adalah sebuah *task* klasifikasi di dalam penelitian pemrosesan bahasa manusia (*natural language processing*). Beberapa metode berdasarkan pembelajaran mesin (*machine learning*) yang biasa digunakan, seperti *Support Vector Machine* [3], *Logistic Regression* [4], *Random Forest* [5], *K-Nearest Neighbors* [6], *Decision Tree* [7], dan *Naive Bayes Classifier* [8] [9]. Metode berbasis *deep learing* juga dalam tahun-tahun terakhir dapat melaksanakan tugas klasifikasi dengan hasil yang mencapai *state-of-the-art*, seperti RNN (*recursive neural network*) berbasis LSTM [10]

Masalah akurasi menjadi penting dalam pengembangan sistem pembelajaran mesin, khususnya kemampuang mesin mendeteksi kelas yang tidak dominan di dalam dataset yang timpang. Di dalam pengembangan metode NB, beberapa cara untuk meningkatkan akurasi antara lain adalah melalui optimasi seleksi fitur (*feature selection*). *Feature selection* bertujuan menganalisa dan menyeleksi data untuk memilih fitur yang berpengaruh terhadap hasil klasifikasi (fitur optimal) dan mengesampingkan fitur yang tidak berpengaruh [8], [9] [11].

Penelitian ini melakukan klasifikasi sentimen pada teks twitter berdasarkan topik program vaksin Covid-19 dari pemerintah, dengan menggunakan metode *Naive Bayes Classifier*. Kontribusi dari penelitian ini adalah bagaimana menangani permasalahan *imbalance class* pada dataset, penggunaan *token word-2-gram* bersamaan dengan *word-1-gram* untuk menambah perbendaharaan fitur Naïve Bayes, dan pencarian kombinasi preprocessing untuk menghasilkan model NB dengan performa terbaik.

2. Metode Penelitian

Metode penelitian merupakan penjelasan langkah-langkah yang akan dilakukan agar mampu menjawab pernyataan-pertanyaan dalam penelitian ini dan menghasilkan informasi yang akurat berdasarkan pertanyaan atau permasalahan pada penelitian yang dilakukan. Tahap penelitian yang dilalui dalam paper ini diorganisasikan sebagai mana Gambar 1 di bawah ini.



Gambar 3. Metode Penelitian

2.1 Tahap *Data Collection*

Data sentimen yang dikumpulkan berjumlah total 13.115 *tweet* unik, didapatkan dengan proses *crawling* dalam rentang Maret 2020 sampai April 2021. Data diperoleh dari aplikasi

[4] https://lp2m.unmul.ac.id/webadmin/public/upload/files/9584b64517cfe308eb6b115847cbe8e7.pdf

tweepy[5], yang mengkases API Twitter untuk menjaring cuitan *tweet* dengan kata kunci tertentu. Beberapa kata kunci yang digunakan adalah "vaksin berhasil", "vaksin gagal", "sakit habis vaksin", "positif covid setelah vaksin", "optimis vaksin aman", dan lain-lain.

Dari proses awal pembersihan data, dipilih data yang dinilai bersih, berjumlah 12.000 *tweet*. Kemudian diberikan label untuk masing-masing *tweet* secara manual oleh manusia (*human annotator*) dengan teknik *crowdsourcing*. Sebanyak 12 orang penutur asli (*native speaker*) dilibatkan untuk memberikan label, setiap *tweet* diberikan label oleh 3 orang annotator. *Label gold-standard* untuk setiap *tweet* ditetapkan berdasarkan suara terbanyak (*majority vote*). Setelah proses anotasi dilakukan terhadap keseluruhan data *tweet*, terdapat sejumlah *tweet* yang tidak valid, karena di antara kelas positif, netral dan negatif, masing-masing mendapat label 1, sehingga tidak dapat dilakukan voting. Oleh sebab itu, *tweet* seperti ini dihapus dari dataset.

Data tersebut kemudian dibagi ke dalam data *training* (8000 *tweet Data-dev*) dan data validasi (778 *tweet Data-dev*), serta data *testing*. Sebanyak 400 *tweet Data-test* yang tidak pernah terlihat selama proses pengembangan metode klasifikasi, digunakan untuk pengujian akhir.

Pada pengujian awal, digunakan seluruh data training untuk diklasifikasi dengan metode dasar/*baseline* NB. Diperoleh fakta bahwa sistem gagal memprediksi kelas positif dan negatif, walaupun akurasinya tinggi, di atas 80%. Hal ini terjadi karena data tidak seimbang antara kelas netral (82,21%) dan kelas yang mengandung sentimen positif maupun negatif (sisanya). Akurasi yang tinggi adalah karena kontribusi dari label netral yang benar diprediksi, dengan label positif dan negatif kebanyakan gagal diprediksi oleh sistem. Oleh karena itu, diperlukan penyeimbangan data antara kelas netral dengan posif dan negatif.

2.2 Tahap *Text Preprocessing* dan *Feature Selection*

Tahap ini bermaksud untuk membersihkan, memproses serta mengkombinasikan data teks dari dataset, Adapun Langkah *preprocessing* dilakukan untuk pembentukan dan pemilihan fitur (*feature selection*), dengan rincian sebagai berikut:

1. *Remove User* (Penghapusan *token* user)
   Yaitu menghilangkan *token* user yang disebut di dalam suatu *tweet* (*mentioned user*)
2. *Remove URL* (Penghapusan URL)
   Yaitu menghapus *link-link* URL dan URI yang terdapat pada *tweet*. *Link* perlu dihapus karena tidak memiliki arti semantik yang jelas, dan tidak bermanfaat untuk klasifikasi
3. *Remove Digit* (Penghapusan Digit Angka)
   Yaitu menghapus *digit-digit* angka yang tidak bermanfaat untuk tugas klasifikasi teks, karena tidak mengandung emosi atau sentimen.
4. *Remove Emoji*
   Yaitu menghapus *tweet* yang mengandung karakter spesial yang menggambarkan emosi tertentu, seperti berikut ini:
   a. *Smile* (senyum) yaitu :), :), :-), (:, (:, (-:, :') , :O
   b. *Laugh* (tertawa) yaitu :D, : D, :-D, xD, x-D, XD, X-D
   c. *Love* (cinta/sayang) yaitu <3, :*
   d. *Wink* (mengedip) yaitu ;-), ;), ;-D, ;D, (;, (-; , @-)
   e. *Sad* (sedih) yaitu :-(, : (, :(,):,)-:, :-/ , :-|
   f. *Cry* (menangis) yaitu :,(, :'(, :"(
5. *Replace Double Spaces with Single Space*
   Yaitu menghapus spasi berlebihan (pemisah antara 2 kata) menjadi spasi tunggal saja.
6. *Remove Repetition*
   Yaitu mengubah huruf-huruf yang double atau huruf yang terdapat repetisi diubah menjadi satu huruf, seperti kata "tidaaakkkkkk" menjadi "tidak". Kelemahannya ada kata-kata yang berubah maknanya, seperti "balikkan" menjadi "balikan".
7. *Remove Single Character*
   Yaitu menghapus setiap jenis karakter yang single.
8. *Remove Punctuation*
   Yaitu menghapus jenis-jenis tanda baca. Hal ini dilakukan dengan mempersingkat waktu analisis dapat mepengaruhi tingkat akurasi.
9. *Case Folding*
   Yaitu mengubah huruf-huruf kapital menjadi huruf kecil.

[5] https://www.tweepy.org/

10. Remove stopword

Yaitu menghapus kata yang yang tidak penting karena frekuensinya yang sangat besar, biasanya digunakan sebagai kata penghubung dan kata ganti.

2.3 *Feature Extraction*

Feature Extraction adalah tahap ekstraksi fitur sebagai input dari sitem *machine learning.* Fitur yang dipilih adalah berupa *token* yang diekstrak dari kalimat-kalimat pada *tweet* setelah melalui *preprocessing* dan pemilihan fitur. Proses ekstraksi merupakan proses tokenisasi menggunakan fungsi *countvectorizer* dari library *Ptyhon scikit-learn*[6]. Proses ini mengubah data teks (kalimat, paragraph, kata-kata di dalam dokumen) menjadi vektor *Bag of Word*s (BoW), sekaligus menghitung kemunculan *token* (kata-kata) di dalam teks (*word count*). Informasi jumlah kemunculan sangat dibutuhkan karena NB bekerja berdasarkan probabilitas kemunculan kata di dalam suatu kelas yang diperiksa.

Sedangkan problem *imbalance class* yang ditangani dengan penyesuaian jumlah kelas netral, menyebabkan jumlah data menjadi jauh lebih kecil. Hal ini dikompromikan dengan melakukan proses tokenisasi ke dalam bentuk kata *unigram* (*word-1-gram*) dan *bigram* (*word-2-gram*). Hasil tokenisasi *unigram* dan *bigram* dapat dilihat pada Tabel 1 dan Tabel 2 berikut.

Tabel 1. *Tweet* ke-1 dan ke-2 pada Dataset yang Akan Ditokenisasi

ID	*Tweet*
1	Akhirnya sekolah dapet undangan vaksin! Yay!
2	Alhamdulillah lansia diedukasi pada paham & ngerti, mereka paham *kalo* kena yah insyallah jd bapil biasa bagi org ygâ€¦

Tabel 2. Sebagian *Token* dari *Tweet* ke-1 dan ke-2 dan Posisinya pada *List Bag of Words*

Kata / *Token*	*Token* ID	Kata / *Token*	*Token* ID
'akhirnya'	429	'diedukasi'	5313
'sekolah'	19327	'paham'	16291
'dapet'	4796	'amp'	803
'undangan'	22769	'ngerti'	15282
'vaksin'	22946	'kalo'	9965
'yay'	24420	'kena'	10870
'akhirnya sekolah'	439	'yah'	24368
'sekolah dapet'	19328	'insyallah'	8763
'dapet undangan'	4813	'jd'	9286
'undangan vaksin'	22770	'bapil'	2024
'vaksin yay':	23790	'biasa'	2994

2.4 Multinomial Naïve Bayes

Untuk menyelesaikan problem berbentuk data diskrit dari fitur-fitur inputan, metode NB yang paling optimal adalah *Multinomial Naïve Bayes* [12]. Klasifikasi teks pada dasarnya adalah klasifikasi dari fitur-fitur berupa *token* yang bersifat diskrit, artinya setiap *token* memiliki nilai probabilitas yang independen. *Multinomial NB* menghitung probabilitas kemunculan suatu *token* berada di masing-masing kelas berdasarkan informasi kemunculan *token* pada kelas tersebut. Formulasi untuk menghitung probabilitas setiap *token* ke-*i* di dalam kalimat yang terdiri atas *k token*, dapat dituliskan sebagaimana persamaan (1) berikut,

$$P(V_i|C = c) = \frac{CountTerms\big(v_i, docs(c)\big) + 1}{AllTerms\big(docs(c)\big) + |V|} \quad (1)$$

[6] https://scikit-learn.org/stable/modules/generated/sklearn.feature_extraction.text.CountVectorizer.html

dengan V_i adalah *token* atau *vocabulary* ke-*i* di dalam teks kalimat yang sedang dievaluasi/ dihitung probabilitasnya, dengan syarat berada di kelas $C = c$ (nama kelas ke-*c*), dan $|V|$ adalah jumlah fitur *token* yang unik di seluruh dokumen. Metode ini menghitung kemunculan *vocabulary token* ke-*i* di dalam dokumen dengan kelas *c*, dengan menjalankan fungsi *CountTerms(vi, docs(c))*, dan menghitung jumlah seluruh fitur *token* kata di dalam seluruh dokumen dengan kelas c, melalui fungsi *AllTerms(docs(c))*.

Setelah probabilitas seluruh fitur *token* dihitung untuk setiap kelas *C*, maka selanjutnya adalah menghitung probabilitas *prior* dari masing-masing kelas, dengan persamaan (2),

$$P(C = c) = \frac{N(C = c)}{|N|} \tag{2}$$

yaitu membagi jumlah dokumen di dalam kelas *c* terhadap jumlah keseluruhan dokumen.

Penetapan kelas hasil klasifikasi NB untuk kalimat tersebut selanjutnya dilakukan dengan menghitung $ProbNB(C = c)$ (nilai probabilitas NB masing-masing kelas), sesuai dengan persamaan (3), yaitu perkalian antara persamaan (1) dan persamaan (2). Nilai yang paling besar di antara kelas ke-*j* dari sejumlah *n* kelas yang ada, menjadi label kelas yang akan diberikan kepada kalimat tersebut (persamaan 4),

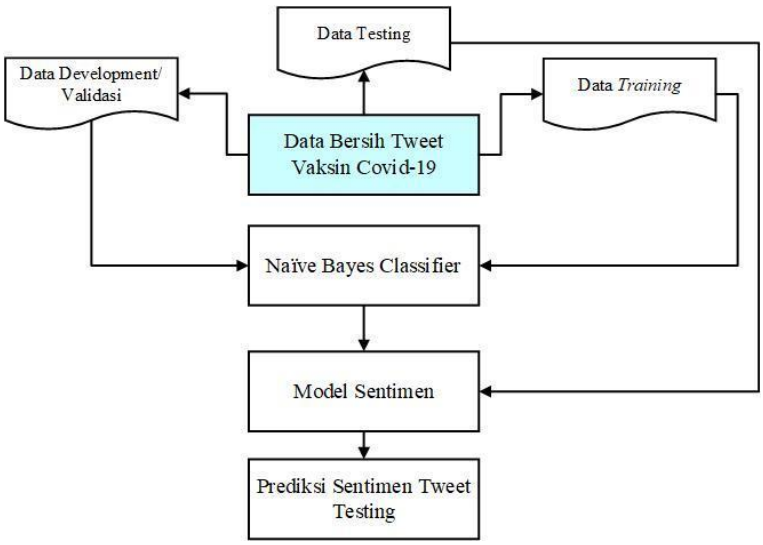
$$ProbNB(C = c) = P(C = c) \cdot \prod_{i=1}^{k} \quad P(C = c) \tag{3}$$

$$C = argmax(ProbNB(C_j^n)) \tag{4}$$

dengan *i* adalah urutan *token* fitur pada kalimat yang terdiri atas k *token*. Pada umumnya, NB akan bekerja dengan sangat baik apabila jumlah label kelas hanya 2 (klasifikasi biner). Dengan cara pemilihan frekuensi fitur alih-alih menggunakan informasi TF.IDF, *multinomialNB* dapat diterapkan untuk tugas klasifikasi *multilabel* dengan lebih baik dan efisien.

2.5 Eksperimen Setup

Untuk mendapatkan hasil yang paling baik dari metode ini, dilakukan Langkah eksperimen sebagaimana diagram pada Gambar 4 berikut ini.



Gambar 4. Proses pelaksanaan eksperimen

Pada saat pelaksanaan eksperimen ini, proses yang pembersihan data teks *twitter* dilakukan menggunakan fungsi teks preprocessing yang sama. Selanjutnya dari komposis Data-train yang tidak seimbang label kelasnya, dilakukan proses penyeimbangan kelas (*balancing*) secara empiris, dengan mengurangi jumlah *tweet* pada kelas netral. Penyeimbangan *Data-dev* juga dilakukan, agar proses validasi model yang dihasilkan dapat diukur dengan proporsional di antara kelas-kelas positif dan negatif terhadap kelas netral yang jumlahnya dominan, seperti terlihat pada Tabel 3.

Tabel 3. Komposisi dan Jumlah tweet pada Data-train dan Data-dev

Label kelas	Data-train				Data-dev			
	Komposisi Awal		setelah *class-balancing*		Komposisi Awal		setelah *class-balancing*	
Positif	463	(5 %)	463	(22%)	45	(5 %)	45	(20%)
Negatif	873	(10%)	873	(42%)	85	(10%)	85	(36%)
Netral	6664	(83%)	700	(34%)	648	(83%)	100	(43%)

Pembentukan matriks Bag of Word berdasarkan kemunculan *token* unigram dan bigram, dilakukan kepada data *Data-train* untuk membentuk matriks fitur. Setelah itu dari matriks fitur ini dihitung probabilitas setiap *token* di dalam setiap data *tweet* (persamaan (1), kemudian dihitung probabilitas NB-nya sebagaimana persamaan (3) untuk setiap kelas, Hasilnya dibandingkan, probabilitas NB terbaik menjadi label yang dipilih sebagai hasil prediksi. Dari keseluruhan label hasil prediksi, dibandingkan terhadap data label yang diberikan manual (*gold-standard*), dan dihitung akurasi prediksi secara keseluruhan.

Hasil model NB yang terbentuk kemudian divalidasi menggunakan *Data-dev* sebagai data ujinya. Model yang memberikan output akurasi tertinggi terhadap *Data-dev* dipilih sebagai kandidat metode yang akan diterapkan untuk mengklasifikasi *Data-test* yang tidak pernah terlihat pada saat training.

Skema pengujian validasi ini dilakukan untuk berbagai kombinasi percobaan sebagai mana tabel 4 berikut. Kombinasi yang dilakukan antara lain dengan memilih komponen pengolahan fitur diterapkan atau tidak diterapkan pada setiap pemodelan NB terhadap *Data-train*.

Tabel 4. Skenario penentuan model terbaik dari *multinomialNB* untuk klasifikasi

No	Kode Eksperimen	Deskripsi
1	STP	Stop Word removal
2	CSF	Case Folding
3	PCT	Punctuation removal
4	CHR	Single char removal
5	REP	Repetition char removal

2.7. Pengukuran Performa

Performa metode klasifikasi diukur berdasarkan *F1-score* sebagai *benchmark scoring* dalam task penelitian klasifikasi sentiment terhadap program vaksin Covid-19 ini, sebagaimana persamaan (5). Untuk melihat akurasi prediksi pada tiap kelas, digunakan persamaan (6) yang dapat dihitung dari *confusion matrix*. Gambar 5 berikut menunjukkan komponen hasil klasifikasi yang dihitung untuk 3 kelas, yaitu Positif (*Pos*), Negatif (*Neg*) dan Netral (*Net*). Aktual artinya adalah hasil anotasi yang dilakukan manusia sebagai *gold-standard* untuk pengujian, sedangkan prediksi adalah hasil yang diperoleh oleh system yang dikembangkan. True (T) berarti hasil prediksi benar sesuai dengan nilai aktualnya, sedangkan False (F) berarti hasil prediksi tidak benar dari kelas yang seharusnya pada aktual.

		PREDIKSI		
		POSITIF	NEGATIF	NETRAL
AKTUAL	POSITIF	TPos	FPosNeg	FPosNet
	NEGATIF	FNegPos	TNeg	FNegNet
	NETRAL	FNetpos	FNetNeg	TNet

Gambar 5. Tabel *multiclass confusion matrix*

Nilai *recall* dan *precision* dihitung untuk melihat seberapa baik metode yang dikembangkan dapat menemukan kasus sentiment di setiap kelasnya (persamaan 7-12). Nilai *True* dan *False* pada setiap kelas Positif, Negatif dan Netral ini dapat dihitung dengan melihat tabel *confusion matrix* di atas. Dan seberapa tepat dari yang berhasil ditemukan. Dan nilai akurasi dapat diperoleh dari perbandingan hasil klasifikasi yang benar untuk keseluruhan kelas, terhadap jumlah semua kasus yang ada.

$$F1 - Score = 2 \cdot \frac{Precision \cdot Recall}{Precision + Recall} \tag{5}$$

$$Akurasi = \frac{TPos + TNeg + TNet}{seluruh\ data} \cdot 100\% \tag{6}$$

$$Recall_{Pos} = \frac{TPos}{TPos + FPosNeg + FPosNet} \tag{7}$$

$$Recall_{Neg} = \frac{TNeg}{TNeg + FNegPos + FNegNet} \tag{8}$$

$$Recall_{Net} = \frac{TNet}{TNet + FNetPos + FNetNeg} \tag{9}$$

$$Precision_{Pos} = \frac{TPos}{TPos + FNegPos + FNetPos} \tag{10}$$

$$Precision_{Neg} = \frac{TNeg}{TNeg + FPosNeg + FNetNeg} \tag{11}$$

$$Precision_{Net} = \frac{TNet}{TNet + FPosNet + FNegNet} \tag{12}$$

Keterangan:
TPos : Jumlah data yang aktualnya positif dan hasil dari prediksi benar bernilai positif
FPosNeg : Jumlah data yang aktualnya positif dan hasil dari prediksi salah bernilai negatif
FPosNet : Jumlah data yang aktualnya positif dan hasil dari prediksi salah bernilai netral
TNeg : Jumlah data yang aktualnya negatif dan hasil dari prediksi benar bernilai negatif
FNegPos : Jumlah data yang aktualnya negatif dan hasil dari prediksi salah bernilai positif
FNegNet : Jumlah data yang aktualnya negatif dan hasil dari prediksi salah bernilai netral
TNet : Jumlah data yang aktualnya netral dan hasil dari prediksi benar bernilai netral
FNetPos : Jumlah data yang aktualnya netral dan hasil dari prediksi salah bernilai positif
FNetNeg : Jumlah data yang aktualnya netral dan hasil dari prediksi salah bernilai negatif

3. Hasil dan Analisa

Eksperimen penelusuran model terbaik dari berbagai kombinasi komposisi langkah pemrosesan teks pada *Data-train* yang sudah di-*balancing*, menghasilkan data akurasi pada *Data-dev* sebagaimana Tabel 4 berikut ini. Tanda centang menyatakan opsi langkah preprocessing teks tersebut diterapkan, dan tanda silang menyatakan tidak diterapkan.

Tabel 4. Data eksperimen hasil penelusuran model NB terbaik

No	FEATURE SELECTION					Akurasi (%)
	STP	CSF	PCT	CHR	REP	
1	X	X	X	X	√	66,52

ISSN (Printed) : 2579-7271
ISSN (Online) : 2579-5406

2	X	X	X	√	X	68,26
3	X	X	X	√	√	66,52
4	X	X	√	X	X	68,26
5	X	X	√	X	√	66.52
6	X	X	√	√	X	68,26
7	X	X	√	√	√	66.52
8	X	√	**X**	X	X	68,26
9	X	√	**X**	X	√	66,52
10	X	√	**X**	√	X	68,26
11	X	√	**X**	√	√	66,52
12	X	√	√	X	X	68,26
13	X	√	√	X	√	66,52
14	X	√	√	√	X	68,26
15	X	√	√	√	√	66,52
16	√	**X**	**X**	**X**	**X**	**69,56**
17	√	**X**	**X**	**X**	√	**69,13**
18	√	**X**	**X**	√	**X**	**69,56**
19	√	**X**	**X**	√	√	**69,13**
20	√	X	√	X	X	69,13
21	√	X	√	X	√	68,69
22	√	X	√	√	X	69,13
23	√	X	√	√	√	68,69
24	√	√	X	X	X	68,26
25	√	√	X	X	√	68,26
26	√	√	X	√	X	65,45
27	√	√	X	√	√	65,45
28	√	√	√	X	X	67,82
29	√	√	√	X	√	65,45
30	√	√	√	√	X	66,95
31	√	√	√	√	√	66,95

Dari Tabel 4, terdapat 2 model yang memiliki akurasi tertinggi, yaitu 69,56%, pada percobaan nomor 16 dan 18. Kedua model NB ini dipilih untuk diterapkan pada *Data-test*. Sementara itu untuk perbandingan, peringkat berikutnya disertakan untuk diujicobakan pada *Data-test* untuk melihat konsistensi hasil pengujian.

3.1 Pengujian dengan *Data-test*

Dari tabel 5 berikut ini, eksperimen terhadap data pengujian yang tidak terlihat pada saat training menghasilkan akurasi sebesar 61%**.** Hasil ini diperoleh dari komposisi penghapusan *stopword* saja dan/atau penghapusan karakter tunggal. Hal ini menunjukkan bahwa komponen-komponen lainnya berguna untuk tugas klasifikasi sentimen, seperti tetap menggunakan huruf kapital yang ada di dalam teks, tidak menghapus tanda baca, dan tidak menghapus karakter-karakter yang berulang di dalam suatu kata. Ini sangat sesuai dengan pengamatan langsung pada *tweet* bahwa kata-kata tersebut sangat berpengaruh terhadap hasil klasifikasi, karena mengandung emosi tertentu, seperti kata “tidaaaakkkk” (mengandung karakter repetisi), atau “DASAR” (penekanan pada kata seperti adanya intonasi keras/tegas di dalam bahasa verbal), atau tanda baca yang biasa digunakan sebagai emoji.

ISSN (Printed) : 2579-7271
ISSN (Online) : 2579-5406

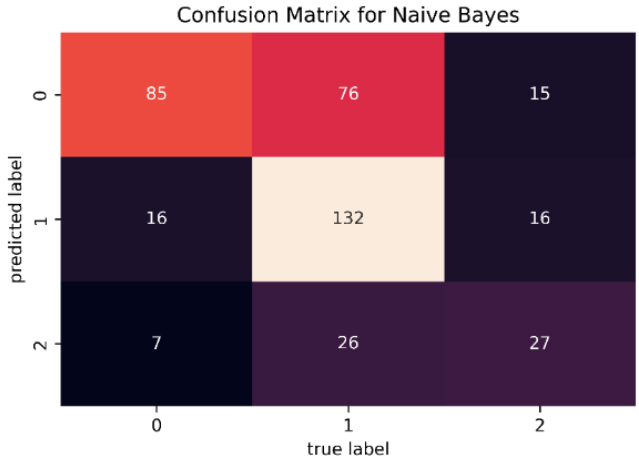
Tabel 5. Akurasi Model NB terpilih setelah diuji dengan Data-test

No.	STP	CSF	PCT	CHR	REP	Train Acc	Test Acc
1.	√	**X**	**X**	**X**	**X**	**69,56**	**61,0**
2.	√	X	X	X	√	69,13	59,5
3.	√	**X**	**X**	√	**X**	**69,56**	**61,0**
4.	√	X	X	√	√	69,13	59,5

3.2 *Confusion Matrix*

Bila ditelusuri hasil-hasil kesalahan klasifikasi secara detil sesuai dengan *confusion matrix* pada Gambar 6, maka dapat dijelaskan bahwa masih ada kesalahan klasifikasi untuk setiap kelas positif (label 2), negatif (label 0) dan netral (label 1). Sistem cukup berhasil memprediksi kelas positif dan negatif, karena nilai *Tpos* dan *Tneg* yang masih lebih tinggi dari nilai *False*-nya, namun secara agregat, nilai *True* masih lebih kecil dari agregat nilai *False*-nya (gabungan nilai *False* untuk kelas-kelas yang salah prediksi).

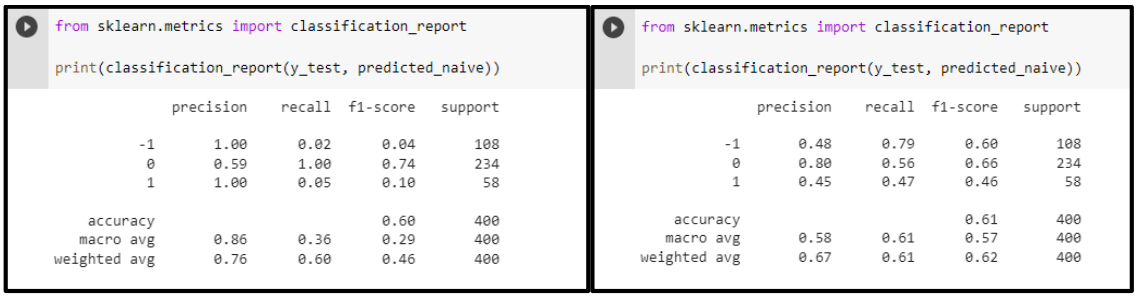
Ditinjau dari proses *balancing* data kelas, keberhasilan ini ditunjukkan oleh *F1-score* pada pengujian yang dilakukan terhadap data awal dan data setelah dilakukan *balancing*, sebagaimana terlihat pada gambar 7 (a) dan (b) berikut. Pada eksperimen menggunakan *Data-train* apa adanya (Gambar 7a), terlihat bahwa sistem tidak berhasil memprediksi kelas negatif dan positif, karena nilai *F1-score* pada kedua kelas tersebut sangat kecil, yaitu hanya 0,04 (kelas negatif) dan 0,10 (kelas positif). Oleh karena itu kita harus berhati-hati terhadap data dengan kelas yang tidak seimbang, tidak bisa berpatokan pada nilai akurasi, karena akurasi disumbangkan oleh data netral yang berhasil diprediksi sehingga terlihat cukup besar di atas 80%.



Gambar 6. Confusion Matrix

Setelah dilakukan proses *balancing* secara empiris dengan mengurangi porsi data netral di dalam *Data-train* (Gambar 7b), diperoleh nilai *F1-score* yang meningkat untuk kelas positif dan negatif, yaitu sebesar 0,6 (kelas positif) dan 0,46 (kelas negatif). Secara keseluruhan mungkin nilai akurasi terlihat turun, namun sistem sudah berhasil mendeteksi adanya sentiment di dalam *tweet*, yang meningkat dari nilai *F1-score* keseluruhan 0,29 menjadi 0,57. `

ISSN (Printed) : 2579-7271
ISSN (Online) : 2579-5406



```
from sklearn.metrics import classification_report

print(classification_report(y_test, predicted_naive))
```

	precision	recall	f1-score	support
-1	1.00	0.02	0.04	108
0	0.59	1.00	0.74	234
1	1.00	0.05	0.10	58
accuracy			0.60	400
macro avg	0.86	0.36	0.29	400
weighted avg	0.76	0.60	0.46	400

(a)

```
from sklearn.metrics import classification_report

print(classification_report(y_test, predicted_naive))
```

	precision	recall	f1-score	support
-1	0.48	0.79	0.60	108
0	0.80	0.56	0.66	234
1	0.45	0.47	0.46	58
accuracy			0.61	400
macro avg	0.58	0.61	0.57	400
weighted avg	0.67	0.61	0.62	400

(b)

Gambar 7. Hasil F1-score dari pengujian menggunakan *imbalance* (a) dan *balanced* (b) data

3.3 Perbandingan dengan metode *machine learning* lainnya

Task penelitian klasifikasi sentimen terhadap kebijakan vaksin covid-19 ini juga dilakukan oleh [3], [4], [10], [13] dengan menggunakan berbagai metode pembelajaran mesin lainnya. Di antara metode-metode tersebut seperti terlihat pada Tabel 6, metode multinomial NB ini dapat digunakan dalam mendeteksi data multilabel, namun masih belum dapat menyaingi performa dari SVM.

Tabel 6. Hasil yang diperoleh berbagai metode ML dengan data-test yang sama

Metode	**F1-score**	**Akurasi**	**Precision**	**Recall**
LSTM [10]	0.54	0.66	**0.75**	0.53
Logistic Regression [4]	0.60	0.67	0.62	0.59
SVM with TF.IDF [3]	0.56	0.65	0.61	0.54
SVM+word embeddings [13]	**0.65**	**0.69**	0.69	**0.63**
NB (penelitian ini)	0.57	0.61	0.58	0.60

4. Kesimpulan

Peneitian ini telah berhasil meningkatkan kemampuan prediksi metode Naïve Bayes untuk dataset yang memilki kelas tidak seimbang (*imbalance classes*), secara signifikan, dengan cara mengurangi porsi data yang berlebihan (timpang). Namun ada hal yang harus dikompromikan terhadap penggunaan cara ini, yaitu jumlah data yang dapat dipakai untuk training menjadi jauh lebih kecil, sehingga jumlah fitur (*token*) menjadi lebih sedikit. Untuk mengantisipasinya, dapat diterapkan penggunaan fitur *token word bigram* disamping *token* kata tunggal (*word unigram*).

Kombinasi dari pemilihan langkah-langkah pre-processing juga dapat dilakukan untuk meningkatkan hasil prediksi, karena tidak semua langkah *preprocessing* cocok diterapkan untuk kasus klasifikasi data *tweet*. Peningkatan hasil *F1-score* yang diperoleh dengan pendekatan yang telah diuraikan dalam paper ini mencapai hampir 200%, yaitu dari 0,29 bila data-train yang digunakan apa adanya, menjadi 0,57 setelah diseimbangkan

Saran untuk penelitian selanjutnya, bila tetap mengembangkan metode Naïve Bayes adalah melakukan optimasi pada proses *balancing dataset*, dan menguji coba fitur TF, TF.IDF terhadap kombinasi *word bigram* dan *trigram* sebagai tambahan fitur dengan batas minimal frekuensi kemunculan di dalam dataset. Saran lainnya adalah mengembangkan metode deteksi bertahap, yaitu proses deteksi berdasarkan pendekatan OVA (*one versus all)* karena metode NB secara native dikembangkan untuk klasifikasi biner.

Referensi

[1] D. Telaumbanua, “Urgensi Pembentukan Aturan Terkait Pencegahan Covid-19 di Indonesia,” *Qalamuna*, vol. 12, pp. 59–70, 2020.

[2] M. Peiris and G. M. Leung, "What can we expect from first-generation COVID-19 vaccines?," *The Lancet*, vol. 396, no. 10261, Nov. 2020, doi: 10.1016/S0140-6736(20)31976-0.

[3] Muhammad Rizki, "Analisis Sentimen Masyarakat Terhadap Vaksin Covid-19 Menggunakan Metode Support Vector Machine pada Media Sosial Twitter," UIN Sultan Syarif Kasim Riau, Pekanbaru, 2022.

[4] Ash Shiddicky and S. Agustian, "Analisis Sentimen Masyarakat Terhadap Kebijakan Vaksinasi Covid-19 pada Media Sosial Twitter menggunakan Metode Logistic Regression," *Jurnal CoSciTech (Computer Science and Information Technology)*, vol. 3, no. 2, pp. 99–106, Aug. 2022, doi: 10.37859/coscitech.v3i2.3836.

[5] A. Amri, "Implementasi Algoritma Random Forest untuk Mendeteksi Hate Speech dan Abusive Language pada Twitter Bahasa Indonesia," UIN Sultan Syarif Kasim Riau, Pekanbaru, 2019.

[6] A. Fadilah, "Penerapan Algoritma K-Nearest Neighbor untuk Mendeteksi Ujaran Kebencian dan Bahasa Kasar pada Twitter Bahasa Indonesia," UIN Sultan Syarif Kasim Riau, Pekanbaru, 2021.

[7] F. Ihsan, I. Iskandar, N. S. Harahap, and S. Agustian, "Decision tree algorithm for multi-label hate speech and abusive language detection in Indonesian Twitter," *Jurnal Teknologi dan Sistem Komputer*, vol. 9, no. 4, pp. 199–204, Oct. 2021, doi: 10.14710/jtsiskom.2021.13907.

[8] A. Arini, L. K. Wardhani, and D. Octaviano, "Perbandingan Seleksi Fitur Term Frequency & Tri-Gram Character Menggunakan Algoritma Naïve Bayes Classifier (Nbc) Pada *Tweet* Hashtag #2019gantipresiden," *KILAT*, vol. 9, no. 1, pp. 103–114, Apr. 2020, doi: 10.33322/kilat.v9i1.878.

[9] S. Suprianto, " Implementasi Algoritma Naive Bayes Untuk Menentukan Lokasi Strategis Dalam Membuka Usaha Menengah Ke Bawah di Kota Medan," *Jurnal Sistem Komputer dan Informatika (JSON)*, vol. I, no. 2, pp. 125–130, 2020.

[10] M. Ihsan, B. S. Negara, and S. Agustian, "LSTM (Long Short Term Memory) for Sentiment COVID-19 Vaccine Classification on Twitter," *Digital Zone: Jurnal Teknologi Informasi dan Komunikasi*, vol. 13, no. 1, pp. 79–89, May 2022, doi: 10.31849/digitalzone.v13i1.9950.

[11] M. Miftahuddin and M. Subianto, "Analisis Produktivitas Tumbuhan Buah Melalui Feature Selection," *Jurnal Matematika, Statistika dan Komputasi*, vol. 8, no. 2, Jan. 2012.

[12] F. Pedregosa *et al.*, "MultinomialNB Sklearn," *Journal of Machine Learning Research*, vol. 11, no. 85, pp. 2825–2830, 2011, Accessed: Sep. 01, 2022. [Online]. Available: https://jmlr.csail.mit.edu/papers/v12/pedregosa11a.html

[13] M. Sahbuddin and S. Agustian, "Support Vector Machine Method with Word2vec for Covid-19 Vaccine Sentiment Classification on Twitter," *JOURNAL OF INFORMATICS AND TELECOMMUNICATION ENGINEERING*, vol. 6, no. 1, pp. 288–297, Jul. 2022, doi: 10.31289/jite.v6i1.7534.