



SPECTA Journal of Technology

E-ISSN : 2622-9099

P-ISSN : 2549-2713

Homepage journal: <https://journal.itk.ac.id/index.php/sjt>



Analisis Sentimen Isu Vaksinasi COVID-19 Pada Twitter Menggunakan Metode Naïve Bayes dan Pembobotan TF-IDF Tokenisasi 1-2

Yashmine Hapsari¹, Syamsul Mujahidin², Nisa Rizqiya Fadhliana^{3*}

^{123*}Program Studi Informatika, Institut Teknologi Kalimantan, Balikpapan, Indonesia

*Corresponding author: nisafadhliana@lecturer.itk.ac.id

Received: 17/January/2023
Accepted: 31/August/2023

Revised: 30/August/2023
Published: 31/August/2023

To cite this article:

Hapsari, Y., Mujahidin, S & Fadhliana, N. R (2023). Analisis Sentimen Isu Vaksinasi COVID-19 Pada Twitter Menggunakan Metode Naïve Bayes dan Pembobotan TF-IDF Tokenisasi 1-2. *SPECTA Journal of Technology*, 7(2), 573 - 583. [10.35718/specta.v7i2.812](https://doi.org/10.35718/specta.v7i2.812)

Abstract

The COVID-19 vaccination has been implemented to cut down the spread of the virus in society, but the status of the vaccine, which has been in the development stage, is one of the factors causing people to hesitate to vaccinate. Therefore, a sentiment analysis was carried out on the issue of COVID-19 vaccination with processes and parameters that could increase the model's accuracy. In this study, sentiment classification was performed using the Naive Bayes method and a dataset of 5,000 tweets related to the vaccination of COVID-19. The weighting stage was applied using the TF-IDF method in which a comparison was made of the effect of using unigram, bigram and 1-2 gram tokenization on model accuracy. The results of one of the experiments with the Gaussian classifier and the ratio train: test is 7:3, the model accuracy is 67.4% for the unigram parameter, 65.5% for the bigram parameter, and 70% for the 1-2 gram parameter, where the model with the combined token is 1 -2 grams has a higher accuracy when compared to using only 1 type of token. Based on these results, it can be concluded that the combination of unigram and bigram tokenization types can provide added value to the model for classifying data, thereby increasing accuracy in analysis related to public sentiment.

Keywords: COVID-19, Naïve Bayes, Sentiment analysis, Vaccination

Abstrak

Vaksinasi COVID-19 telah diberlakukan untuk mengurangi penyebaran virus corona dalam masyarakat, namun status vaksin yang masih dalam tahap pengembangan menjadi salah satu faktor keraguan masyarakat untuk melakukan vaksinasi. Oleh karena itu, dilakukan analisis sentimen terhadap isu vaksinasi COVID-19 dengan proses dan parameter yang dapat meningkatkan akurasi model. Pada penelitian ini, pengklasifikasian sentimen dilakukan dengan menggunakan metode Naive Bayes dan dataset berupa 5000 tweet terkait vaksinasi COVID-19. Tahap pembobotan dilakukan dengan metode TF-IDF dimana dilakukan perbandingan terhadap pengaruh penggunaan tokenisasi unigram, bigram dan 1-2 gram terhadap akurasi model. Hasil dari salah satu percobaan dengan pengklasifikasi Gaussian dan perbandingan train:test yaitu 7:3, didapatkan akurasi model 67.4% untuk parameter unigram, 65.5% untuk parameter bigram, dan 70% untuk parameter 1-2 gram, dimana model dengan token gabungan yaitu 1-2 gram memiliki akurasi yang lebih tinggi jika dibandingkan dengan penggunaan 1 jenis token saja. Berdasarkan hasil tersebut, dapat disimpulkan bahwa penggabungan jenis tokenisasi unigram dan bigram dapat memberikan nilai tambah terhadap model untuk mengklasifikasikan data, sehingga dapat meningkatkan akurasi dalam analisis terkait sentimen publik.

Kata Kunci: COVID-19, Naïve Bayes, Analisis sentiment, Vaksinasi

1. Pendahuluan

Pandemi COVID-19 yang mulai mewabah sejak akhir tahun 2019 telah menyebabkan kematian jutaan orang dari berbagai negara. *Coronavirus Disease 2019* atau COVID-19 adalah sebutan dari penyakit yang menyerang organ pernapasan dan saraf dengan tingkat penyebaran yang sangat cepat terhadap manusia. Hingga tahun 2022, COVID-19 masih mewabah di banyak negara, termasuk Indonesia. Salah satu upaya pencegahan mewabahnya COVID-19 oleh Pemerintah yaitu pelaksanaan vaksinasi terhadap masyarakat secara menyeluruh. Namun status vaksin yang masih dalam tahap pengembangan dan belum terbukti secara kuat untuk menangkal COVID-19 menjadikan hal ini sebagai salah satu faktor yang menyebabkan keraguan dalam masyarakat perihal keperluan vaksinasi. Tidak hanya itu, keraguan dan kesalahan informasi vaksin juga menjadi hambatan besar dalam mendorong masyarakat untuk melakukan vaksinasi COVID-19.

Salah satu tolak ukur untuk menganalisis sentimen terkait isu pelayanan kesehatan dan kebijakan pemerintah saat pandemi COVID-19 dapat dilakukan dengan mengamati pembicaraan publik. Saat ini, media sosial Twitter telah menjadi salah satu tempat untuk menyuarakan pendapat dan keluhan penggunaannya terhadap vaksinasi COVID-19. Media ini juga dimanfaatkan oleh Pemerintah dan lembaga kesehatan untuk mengkomunikasikan, dan mendengar suara rakyat tentang COVID-19 yang ada di Twitter. Oleh karena itu, perlu dilakukan pengamatan terhadap sentimen di media sosial agar pemerintah dan/atau lembaga kesehatan dapat menentukan strategi yang tepat dalam menangani wabah COVID-19, khususnya dalam hal komunikasi publik.

Terdapat berbagai metode untuk melakukan klasifikasi dalam analisis sentimen, diantaranya yaitu *Support Vector Machine* (SVM), pendekatan *Lexicon*, dan *Naive Bayes Classification*. Salah satu penelitian terdahulu oleh Hayati dan Alifi (2021) mengenai analisis sentimen terkait isu vaksinasi dengan menggunakan metode SVM, menerapkan metode SVM dengan menggunakan tokenisasi *unigram* dan *bigram* untuk menambah akurasi analisis sentimen yang dilakukan. Dari proses evaluasi yang dilakukan, penambahan tokenisasi *bigram* pada tahap *preprocessing* terbukti dapat menambah nilai rata-rata setiap aspek pengukuran evaluasi sebesar 0,6% sampai 0,7%. Dari penelitian tersebut didapatkan nilai tertinggi seluruh aspek pengukuran dengan metode SVM mencapai 84%. Penelitian lain mengenai analisis sentimen dengan metode *Naive Bayes* salah satunya oleh Winda (2021) terkait analisis sentiment isu vaksin COVID-19, yang menyatakan bahwa hasil analisis yang dilakukan memiliki nilai akurasi sebesar 93%, namun tidak melakukan penggabungan *n-gram* pada proses tokenisasi. Oleh karena itu proses analisis pada penelitian ini akan menggunakan metode Naive Bayes, dengan memasukkan penggabungan token *unigram+bigram* pada proses tokenisasi untuk melihat pengaruh yang dihasilkan proses tersebut terhadap akurasi klasifikasi jika dibandingkan dengan tokenisasi *unigram* saja atau *bigram* saja. Penelitian ini diharapkan dapat memberikan manfaat dalam pengembangan penelitian analisis sentimen, serta diharapkan dapat menjadi dasar dalam pengambilan langkah untuk sosialisasi atau pendekatan pada masyarakat mengenai isu vaksinasi COVID-19.

2. Metodologi

Penelitian diawali dengan melakukan studi mengenai penelitian terkait dan mencakup penjelasan mengenai penerapan metode *Naive Bayes Classifier* dalam analisis sentimen, pengumpulan *dataset* dari Twitter API, sistem *preprocessing*, dan evaluasi dari model klasifikasi dan hasilnya. *Dataset* diambil dari *tweet-tweet* dengan akses token Twitter API. *Tweet* dengan konteks mengenai isu vaksinasi COVID-19 dicari dengan kata kunci terkait COVID-19 dan vaksinasi.

Pengolahan *dataset* dilakukan dengan membersihkan *tweet* yang digunakan dengan tahap-tahap diantaranya yaitu *case folding*, ekstraksi *RT mention*, *hashtag* dan *emoticon*, *cleaning*, normalisasi kata (dengan menggunakan *dictionary* dari *Colloquial Indonesian Lexicon*), *stemming* dan penghapusan

stopword. *Dataset* yang telah dibersihkan kemudian dilakukan proses *labeling* atau pemberian nilai terhadap setiap data *tweet*. Nilai *dataset* akan diklasifikasikan menjadi 2, yaitu *tweet* positif dan *tweet* negatif. Proses pelabelan *dataset* berdasarkan kamus *Indonesia Sentiment Lexicon* oleh Fajri Koto dan Gemala Y. Rahmaningtyas (2017) dengan metode *Lexicon*, yaitu memberi label berdasarkan total bobot dari setiap kata pada *tweet* berdasarkan kamus.

Data *tweet* yang telah memiliki label kemudian diekstraksi melalui tahap *feature extraction*. *Feature extraction* dilakukan dengan melakukan *weighting* terhadap token-token dengan TF-IDF. *Term frequency* dari suatu kata dihitung sebagai berapa kali kata tersebut muncul dalam sebuah dokumen (*tweet*), terhadap jumlah keseluruhan kata dalam *tweet* tersebut; sedangkan nilai IDF dihitung dengan fungsi berikut (Ahuja dkk, 2019):

$$IDF(t) = \log\left(\frac{N}{DF}\right) \quad (1)$$

dimana N adalah jumlah *tweet* dalam sebuah *dataset* dan DF adalah jumlah *tweet* yang mengandung kata t tersebut (Ahuja dkk, 2019). Nilai TF dan IDF tersebut kemudian dikalikan untuk menghasilkan nilai TF-IDF yang akan disimpan untuk proses klasifikasi. Hasil dari proses TF-IDF adalah nilai bobot dari setiap token dalam *tweet* yang kemudian menjadi nilai dari fitur untuk proses klasifikasi, dimana terdapat 3 percobaan jenis tokenisasi dalam penelitian ini, yaitu *unigram*, *bigram*, dan *1-2 gram*.

Dataset yang telah memiliki bobot dan label kemudian dilatih dan diuji dalam algoritma pengklasifikasi sentimen metode *Naive Bayes*. Konsep dasar yang digunakan dalam sistem klasifikasi ini adalah teorema peluang bersyarat oleh *Bayes* berikut (Oktasari dkk, 2016):

$$P(C|x) = P(C) \cdot P(x|C) \quad (2)$$

dimana peluang kejadian kelas C dengan kondisi x ditentukan dari peluang C dan peluang bersyarat dari C . Persamaan ini kemudian dikembangkan menjadi persamaan *Naive Bayes* berikut untuk menghitung probabilitas positif dan negatif setiap data:

$$P(C|x) = \frac{P(x|C) \cdot P(C)}{P(x)} \quad (3)$$

dengan $P(C)$ merupakan probabilitas dari suatu kelas, $P(x \vee C)$ sebagai frekuensi kondisi data tertentu yang memiliki kelas C , dan $P(x)$ sebagai jumlah seluruh data latih (Oktasari dkk, 2016). Hasil dari persamaan (3) tersebut kemudian diklasifikasikan ke dalam kelas sentimen dengan kondisi apabila nilai probabilitas positif lebih besar daripada probabilitas negatif, maka sentimen masuk kedalam kelas positif, dan apabila tidak maka akan masuk kedalam kelas negatif.

Proses pelatihan dan pengujian *dataset* dilakukan dengan pelatihan model menggunakan data *train*, dan kemudian diuji dengan memasukkan data *test*. Validasi dilakukan dengan metode *K-folds cross validation* untuk mengurangi tingkat terjadinya *overfitting*. Dari penggunaan metode tersebut diperoleh sepuluh nilai evaluasi dari setiap tahap klasifikasi.

Hasil dari proses klasifikasi kemudian dievaluasi akurasi dengan *confusion matrix*, dan *classification report*, yaitu *f-score*, *precision*, *recall*, dan *accuracy*, dengan nilai didapatkan dari persamaan berikut (Rahmad dkk, 2020):

$$Precision = \frac{TP}{TP+FP} \quad (4)$$

$$Recall = \frac{TP}{TP+FN} \quad (5)$$

dengan TP adalah banyak data bernilai *True Positive*, FP adalah banyak data bernilai *False Positive*, dan FN adalah banyak data bernilai *False Negative* (Rahmad dkk, 2020). Rata-rata dari *recall* dan *precision* disebut *f1-score*. Nilai ini merupakan parameter yang lebih penting daripada akurasi ketika memiliki distribusi kelas yang tidak merata dalam data. *F1-score* dihitung sebagai berikut (Ahuja dkk,

2019):

$$Fscore = \frac{2 \cdot Recall \cdot Precision}{Recall + Precision} \quad (6)$$

Evaluasi ini akan membandingkan hasil klasifikasi dari *dataset* yang menggunakan tokenisasi *unigram* saja dan *bigram* saja, dengan hasil klasifikasi *dataset* yang menggunakan tokenisasi *1-2 gram*. Selanjutnya, model klasifikasi *Naive Bayes* akan dihubungkan dengan *back-end* pada aplikasi *web* untuk dapat menganalisis sentimen dengan *input* secara langsung. *Framework* yang digunakan pada pembuatan aplikasi ini yaitu *library Flask*.

3. Hasil dan Pembahasan

3.1. Pengumpulan Dataset

Proses *scraping* dilakukan dengan metode *by keyword*, dimana pencarian menggunakan kata kunci terkait vaksinasi COVID-19 yaitu ‘vaksinasi’, ‘vaksin booster’, ‘vaksin covid’, ‘vaksin corona’, ‘sinovac’ dan ‘covac’. Pengumpulan data dengan maksimal *tweet* diambil dengan 6 kata kunci berbeda tersebut (setiap kata kunci didapatkan maksimal 100 *tweet* per harinya) berjumlah total 7636 *tweet*, dengan pengumpulan *tweet* dilakukan sejak bulan Mei 2022 sampai Oktober 2022. Jumlah data *tweet* yang diambil melebihi dari rencana awal *dataset* yaitu 5000 *tweet* untuk mengantisipasi banyaknya data *tweet* yang harus dihapus saat proses penghilangan duplikasi, seperti data *retweet* yang terambil saat *scraping* data. Hasil dari proses pengumpulan data ini berupa 7636 data berisi tanggal *tweet*, nama pengirim (*username*) *tweet* dan isi *tweet*.

3.2. Pengolahan Dataset

Hasil dari proses pengumpulan data kemudian diproses pada tahap pengolahan data sebelum dilakukan klasifikasi.

1. **Case Folding** *Case folding*: dilakukan dengan menerapkan metode *lower()* terhadap data *string tweet*. Proses *case folding* menghasilkan data *tweet* yang awalnya memiliki huruf besar menjadi huruf kecil (*lowercase*).
2. **Ekstraksi RT Mention, Hashtag dan Emotico:n** Proses ekstraksi dilakukan dengan menggunakan fungsi *findall()* dari modul *Regular Expressions* untuk mencari pola *RT mention* dan *hashtag*, kemudian menggunakan fungsi *sub()* untuk menghapus *RT mention* dan *hashtag* yang terdapat dalam data ‘text’. Ekstraksi *emoticon* dilakukan dengan menggunakan modul *emoji*. Tujuan dari ekstraksi ini adalah untuk menyisahkan *mention*, *hashtag* dan *emoticon* dari sentimen yang digunakan untuk klasifikasi. Hasil ekstraksi ini disimpan pada tahap *preprocessing* dan tidak digunakan untuk analisis sentimen. *Mention*, *hashtag* dan *emoticon* yang disimpan akan ditampilkan sebagai salah satu *output* pada aplikasi *web*.
3. **Cleaning**: *Cleaning* data dilakukan dengan menghapus beberapa pola *string* dalam *dataset* yang tidak diperlukan atau tidak memiliki makna yang signifikan dalam proses klasifikasi, seperti angka, notasi, simbol, dan sebagainya. Pola yang dihilangkan dalam proses ini diantaranya: 1.) URL; 2.) Karakter ‘\n’; 3.) Tanda baca; 4.) Spasi yang berlebihan; 5.) Angka dan 6.) kata yang hanya mengandung 1 atau 2 huruf.
4. **Normalisasi**: Kata Normalisasi data dilakukan dengan mengubah kata-kata *slang* dan singkatan kata menjadi bahasa formal Indonesia menggunakan *dictionary* dari *Colloquial Indonesian Lexicon*. Normalisasi bertujuan untuk menyamakan makna dan meminimalisir terjadinya kesalahan penafsiran saat klasifikasi, misalnya kata ‘nggak’ dan ‘gak’ akan diubah menjadi kata ‘tidak’.
5. **Stemming**: Proses *stemming* dilakukan dengan menggunakan kelas *StemmerFactory()* dari kamus *Sastrawi*. Proses ini mentransformasi kata-kata dengan infleksi dalam bahasa Indonesia yang terdapat dalam data menjadi kata dasar atau baku.
6. **Penghapusan Stopword** : Tahap penghapusan *stopword* dilakukan 2 kali, yaitu dengan salah satu kelas pada *Sastrawi*, *StopWordRemoverFactory()*, dan dengan *filtering* dari kamus *stopword* bahasa Indonesia dari kamus NLTk.

Setelah dilakukan tahap-tahap *preprocessing*, data disaring dengan menerapkan salah satu metode dari *Pandas*, yaitu *drop_duplicates()*, serta menghapus data yang kosong pada data *tweet* yang telah

dibersihkan. Proses ini menyebabkan terjadinya pengurangan jumlah data akibat adanya duplikasi atau kesamaan dalam data *tweet*. *Dataset* awal yang berjumlah 7636 data berkurang menjadi 6142 data setelah mengalami proses pembersihan ini.

3.3. Pelabelan Dataset

Kamus *Lexicon* yang digunakan untuk *labeling* data dilakukan proses *stemming* terlebih dahulu untuk menyesuaikan bentuk token-token yang terdapat dalam kamus dengan *dataset* yang telah diproses. Selain itu ditambahkan beberapa kata baru dari *dataset* yang memiliki frekuensi distribusi kata cukup tinggi dan memiliki nilai positif atau negatif.

Telah dilakukan metode pelabelan lain selain dengan kamus *Lexicon*, salah satunya yaitu dengan menggunakan model klasifikasi sentimen terlatih yang telah ada. Model klasifikasi yang digunakan salah satunya yaitu *Indonesian RoBERTa Base IndoLEM Sentiment Classifier* oleh Wilson Wongso dengan rata-rata *f1-score* model 84%. Hasil yang didapat dengan *dataset* hasil *preprocessing* sebanyak 6142 data, yang mana saat *labeling* dengan kamus *Lexicon* menghasilkan 2855 data positif dan 3287 data negatif, pada prediksi sentimen dengan model *RoBERTa* didapatkan hasil yaitu 5169 data negatif dan 927 data positif. Oleh karena itu model ini tidak dapat digunakan.

3.4. Pembobotan Dataset

Dataset yang telah diberi label kemudian dilakukan *splitting* menjadi data *train* dan data *test* dengan menggunakan fungsi *train_test_split()* dari *scikit-learn* untuk melalui proses pembobotan secara terpisah. Fungsi ini membagi data sesuai ukuran argumen yang diberikan yaitu secara random menjadi 2 kali percobaan, yaitu *split 1* dengan perbandingan *train* : *test* 7:3 dan *split 2* dengan perbandingan 8:2 seperti pada Tabel 1 berikut.

Tabel 1: Hasil *Splitting* Data

	<i>Split 1</i>		<i>Split 2</i>	
	<i>Train</i>	<i>Test</i>	<i>Train</i>	<i>Test</i>
Persentase	70%	30%	80%	20%
	3500	1500	4000	1000
	1732	768	2000	500
	1768	732	2000	500

Data *train* dan data *test* kemudian diekstraksi untuk melalui tahap *tokenizing* dan *weighting* dengan TF-IDF. Proses TF-IDF yang diterapkan dengan fungsi *TfidfVectorizer()* dari *scikit-learn* pada penelitian ini dilakukan dengan menggabungkan 2 jenis *n-gram*, yaitu *unigram* dan *bigram*, serta percobaan *unigram* saja dan *bigram* saja sebagai perbandingan. Tokenisasi dilakukan dengan menentukan parameter *ngram_range* yang terdapat dalam fungsi. Selain *ngram_range*, terdapat beberapa parameter lainnya dalam *TfidfVectorizer()* yang diuji untuk mendapatkan model klasifikasi dengan nilai akurasi yang tinggi, dilihat pada Tabel 2.

Tabel 2: Daftar Parameter Proses Pembobotan

<i>Class</i>	<i>Parameter</i>	<i>Values</i>
	<i>ngram_range</i>	(1,1)
		(1,2)
		(2,2)
<i>TfidfVectorizer()</i>	<i>max_df</i>	0.5
		0.75
	<i>max_features</i>	None
		$fd \geq 2$
		$fd \geq 5$

<i>Class</i>	<i>Parameter</i>	<i>Values</i>
		$fd \geq 10$
		$fd \geq 50$
	<i>norm</i>	'11' '12'
	<i>use_idf</i>	True False

Hasil perhitungan disimpan dalam *list* berisi nilai TF-IDF per token pada masing-masing dokumen. Nilai TF-IDF kemudian digunakan sebagai nilai fitur setiap token dalam proses klasifikasi.

3.5. Proses Klasifikasi

Data *train* dan data *test* yang telah memiliki bobot masing-masing kemudian akan dilatih dan diuji dalam kelas pengklasifikasi dari *scikit-learn*. Percobaan pada penelitian ini dilakukan dengan metode *Naive Bayes* yang diantaranya menggunakan kelas *GaussianNB()* dan *MultinomialNB()*, serta metode *SVM* menggunakan kelas *SVC()* sebagai pembandingan.

Tabel 3: Daftar Parameter Proses Klasifikasi

<i>Class</i>	<i>Parameter</i>	<i>Values</i>
<i>GaussianNB()</i>	<i>var_smoothing</i>	1e-9
		2e-9
<i>MultinomialNB()</i>	<i>fit_prior</i>	True
		False
<i>SVC()</i>	<i>gamma</i>	'scale'
		'auto'
	<i>kernel</i>	'linear'
		<i>decision_function_shape</i>

Model yang telah dilatih kemudian divalidasi menggunakan metode *10-folds cross-validation* dan diuji dengan memasukkan data *test*. Pemberlakuan *10-folds* ini menjadikan data *train* dan data *test* terbagi menjadi 10 bagian, dan dihitung nilai akurasi dari setiap bagian tersebut.

3.6. Evaluasi

Hasil dari proses klasifikasi yang menghasilkan 10 nilai *fold* pada setiap percobaannya kemudian didapatkan akurasi dengan menghitung rata-rata dari 10 nilai *fold* setiap percobaan. Hasil evaluasi juga dihasilkan dalam bentuk *classification report* dan *confusion matrix* yang didapatkan dari fungsi *classification_report* dan *confusion_matrix* dari *scikit-learn*.

Perbandingan awal dilakukan dengan membandingkan nilai akurasi setiap kelas klasifikasi berdasarkan parameter *ngram_range*, yaitu *unigram*, *bigram* dan *1-2 gram*. Percobaan dengan akurasi tertinggi kemudian diambil dari setiap pengklasifikasi, yaitu *Gaussian* dan *Multinomial*, serta *SVM* sebagai metode klasifikasi pembandingan lainnya, seperti pada Tabel 4.

Tabel 4: Akurasi Tertinggi Model Klasifikasi

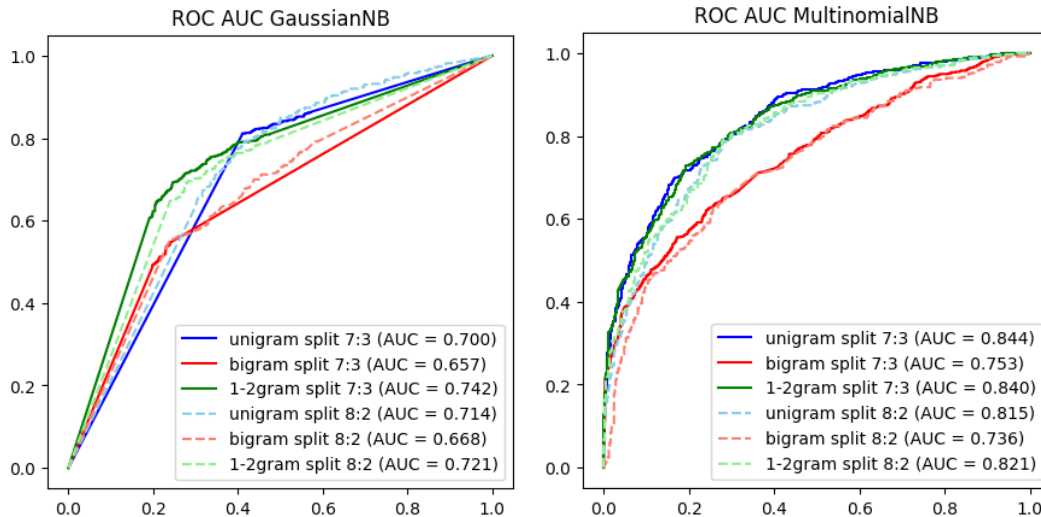
<i>Train : Test</i>	<i>Classifier</i>	<i>n-gram</i>	<i>Accuracy</i>	<i>Class</i>	<i>Precision</i>	<i>Recall</i>	<i>F1-Score</i>	<i>Confusion Matrix</i>
7:3	<i>Gaussian</i>	1	0.69467	-1	0.8	0.67	0.73	[514 254]
				1	0.7	0.83	0.76	[125 607]
		2	0.64933	-1	0.66	0.78	0.71	[598 170]
				1	0.71	0.58	0.64	[309 423]

<i>Train : Test</i>	<i>Classifier</i>	<i>n-gram</i>	<i>Accuracy</i>	<i>Class</i>	<i>Precision</i>	<i>Recall</i>	<i>F1-Score</i>	<i>Confusion Matrix</i>	
8:2	<i>Naive Bayes</i>	1-2	0.71467	-1	0.68	0.88	0.77	[675 93]	
				1	0.82	0.56	0.67	[320 412]	
		1	0.674	-1	0.75	0.66	0.7	[328 172]	
				1	0.69	0.78	0.73	[112 388]	
	2	0.655	-1	0.73	0.5	0.59	[250 250]		
			1	0.62	0.82	0.71	[91 409]		
	7:3	<i>Multinomial</i>	1-2	0.7	-1	0.65	0.88	0.75	[442 58]
					1	0.82	0.52	0.64	[238 262]
1			0.768	-1	0.83	0.69	0.75	[527 241]	
				1	0.72	0.85	0.78	[108 624]	
2		0.68467	-1	0.76	0.58	0.66	[442 326]		
			1	0.65	0.81	0.72	[139 593]		
8:2		<i>Multinomial</i>	1-2	0.76067	-1	0.82	0.7	0.75	[536 232]
					1	0.73	0.84	0.78	[117 615]
	1		0.751	-1	0.79	0.78	0.79	[389 111]	
				1	0.78	0.8	0.79	[102 398]	
	2	0.681	-1	0.74	0.64	0.68	[320 180]		
			1	0.68	0.77	0.72	[115 385]		
	7:3	<i>SVM</i>	1-2	0.751	-1	0.8	0.7	0.75	[352 148]
					1	0.74	0.83	0.78	[87 413]
1			0.76733	-1	0.82	0.8	0.81	[615 153]	
				1	0.79	0.81	0.8	[139 593]	
2		0.69	-1	0.69	0.8	0.74	[618 150]		
			1	0.75	0.62	0.68	[277 455]		
8:2		<i>SVM</i>	1-2	0.768	-1	0.82	0.8	0.81	[614 154]
					1	0.8	0.82	0.81	[132 600]
	1		0.756	-1	0.82	0.82	0.82	[412 88]	
				1	0.82	0.82	0.82	[89 411]	
	2	0.672	-1	0.68	0.83	0.75	[415 85]		
			1	0.78	0.62	0.69	[192 308]		
	1-2	0.755	-1	0.82	0.84	0.83	[418 82]		
			1	0.83	0.81	0.82	[94 406]		

Dilihat pada Tabel 4, dari setiap jenis pengklasifikasi, *splitting*, dan parameter tersebut, diambil percobaan dengan akurasi tertinggi sehingga dari 1440 percobaan didapatkan 18 model dengan akurasi tertinggi. Dilihat dari akurasinya, diantara 2 metode *Naive Bayes*, model *Multinomial* memiliki akurasi yang lebih tinggi dengan nilai 76.8%, dimana model *Gaussian* memiliki akurasi tertinggi 71.5%. Dilihat dari tokenisasinya, pada percobaan dengan model *Gaussian*, akurasi tertinggi didapatkan saat penggunaan tokenisasi jenis *1-2 gram*. Percobaan dengan model *Multinomial* dan *SVM* memiliki nilai akurasi tertinggi saat menggunakan tokenisasi *1-gram (unigram)*, namun memiliki selisih yang rendah

atau dapat sama dengan penggunaan tokenisasi *1-2 gram*.

Salah satu bentuk evaluasi yang digunakan pada penelitian ini selain *confusion matrix* yaitu kurva ROC dan nilai AUC. Semakin baik performansi suatu model, maka garis kurva ROC akan menjorok ke nilai tertinggi sumbu *y*, dan nilai luas AUC akan lebih besar. Sumbu *x* pada kurva merupakan nilai *False Positive Rate* dari model, sedangkan sumbu *y* merupakan nilai *True Positive Rate*. Kedua nilai tersebut merupakan nilai yang diambil dari *confusion matrix*. Berikut merupakan visualisasi kurva ROC dari percobaan menggunakan metode *Gaussian* dan *Multinomial*.



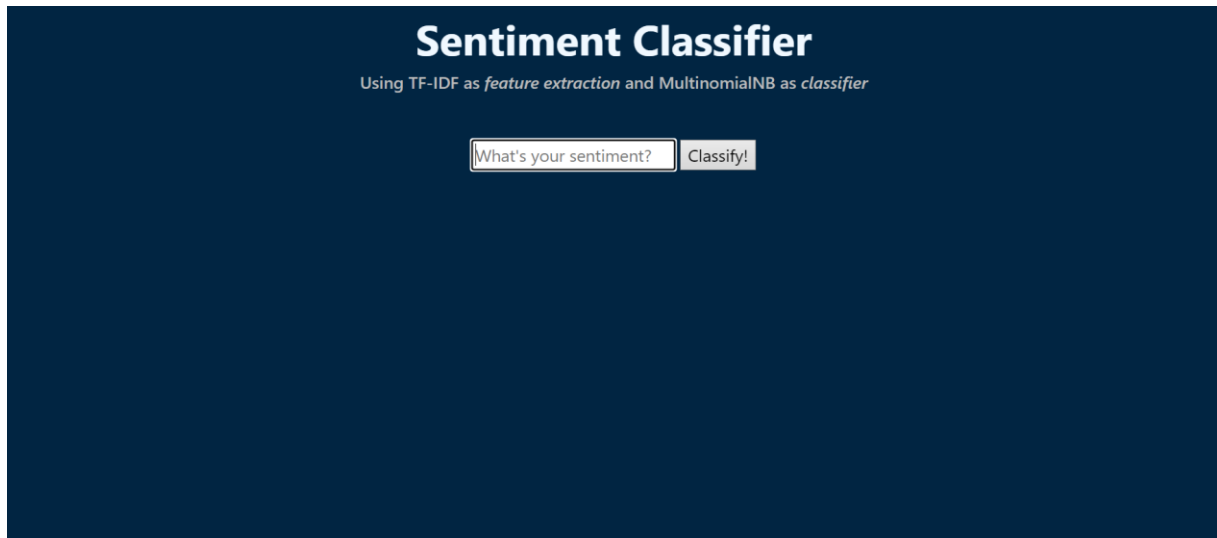
Gambar 1: Kurva ROC Percobaan *Gaussian* (kiri) dan *Multinomial* (kanan)

Dilihat dari perbandingan *n-gram* model *Gaussian* pada gambar 3.1, percobaan dengan parameter *bigram* memiliki garis kurva lebih rendah dibandingkan dengan 2 jenis tokenisasi lainnya. Hal ini menyatakan bahwa model *Gaussian* dengan penggunaan tokenisasi *unigram* dan *1-2 gram* memiliki performansi lebih baik daripada model dengan tokenisasi *bigram*. Dilihat dari luas AUC-nya, percobaan dengan tokenisasi *1-2 gram* memiliki nilai AUC yang lebih tinggi dibandingkan dengan percobaan tokenisasi *unigram*. Hal ini menyatakan bahwa model *Gaussian* dengan tokenisasi *1-2 gram* memiliki performansi yang lebih baik dalam menentukan probabilitas kelas data dibandingkan dengan model *Gaussian* tokenisasi *unigram*. Disisi lain, pada model *Multinomial*, percobaan dengan tokenisasi *bigram* memiliki garis kurva terendah, dan percobaan tokenisasi *1-2 gram* hampir sama dengan percobaan tokenisasi *unigram*, dimana selisih nilai AUC keduanya kurang dari 1%. Berbeda dengan model *Gaussian*, percobaan dengan tokenisasi *unigram* pada model *Multinomial* tidak selalu lebih kecil daripada percobaan dengan tokenisasi *1-2 gram*, dengan selisih yang kecil.

Evaluasi dari percobaan setiap model memiliki keragaman dari masing-masing parameter yang mengoptimalkan model klasifikasi. Berdasarkan keragaman parameter ini, dapat dinyatakan bahwa selain parameter tokenisasi pada TF-IDF, percobaan model dengan akurasi tertinggi memiliki parameter-parameter yang tidak tentu, disesuaikan dengan bentuk *dataset* yang disediakan.

3.7. Pembuatan Aplikasi Web

Pembuatan aplikasi *web* dilakukan dengan menerapkan *framework Flask* dengan satu *endpoint*. Model yang digunakan dalam program diambil dari percobaan *Multinomial* akurasi tertinggi dengan parameter *max_features* bernilai *None*, *norm* bernilai *'l1'*, *max_df* bernilai *0.5*, *use_idf* bernilai *True*, dan *fit_prior* bernilai *False*.



Gambar 2: Tampilan Laman Awal *Web*

Tampilan awal *web*, pada Gambar 2, menampilkan judul *web* dan form untuk menginput sentimen secara langsung. Selain itu ditampilkan juga metode yang digunakan dalam proses klasifikasi sentimen, yaitu TF-IDF dan *Naive Bayes Multinomial*.



Gambar 3: Tampilan Setelah *Submitting*

Setelah sentimen disubmit, laman akan menampilkan hasil proses, seperti pada Gambar 3. Hasil yang ditampilkan yaitu berupa sentimen yang diinput, prediksi klasifikasi, dan komponen lain selain kalimat yang terdapat pada sentimen, seperti *mention*, *hashtag*, dan *emoticon*. Hasil prediksi pada aplikasi *web* sederhana ini digolongkan menjadi 3, yaitu positif, negatif, dan tidak teridentifikasi atau netral. Dataset dan kode yang digunakan dimulai dari tahap pengumpulan data sampai pembuatan *web* dapat dilihat pada link *github* penulis (Hapsari, 2022).

4. Kesimpulan

Berdasarkan hasil yang didapatkan, dapat diambil kesimpulan dari penelitian ini adalah Proses analisis sentimen dilakukan dengan melakukan *scraping dataset* dari Twitter API dengan kata kunci seputar vaksinasi COVID-19, *preprocessing dataset*, pelabelan *dataset*, *splitting dataset* menjadi data *train* dan data *test*, pembobotan dengan metode TF-IDF, *training* model NBC dan validasi dengan sistem *cross validation*. Model yang telah divalidasi kemudian diuji dengan data *test* dan dievaluasi dengan *confusion matrix* dan *classification report*. Berdasarkan percobaan yang telah dilakukan terhadap tokenisasi *unigram*, *bigram*, dan *1-2 gram* pada proses TF-IDF, penggunaan *1-2 gram* dapat memberikan hasil

akurasi yang lebih tinggi pada model klasifikasi *Naive Bayes* dibandingkan dengan penggunaan *unigram* saja dan *bigram* saja. Diterapkan pada percobaan dengan pengklasifikasi *Naive Bayes Gaussian* dan perbandingan data *train* dan data *test* yaitu 7:3, didapatkan akurasi model 67.4% untuk parameter *unigram*, 65.5% untuk parameter *bigram*, dan 70% untuk parameter *1-2 gram*, dimana model dengan gabungan token *1-2 gram* memiliki akurasi yang lebih tinggi jika dibandingkan dengan penggunaan 1 jenis token saja.

Saran yang dapat diberikan dalam upaya memperbaiki penelitian ini pada masa yang akan datang diantaranya sebagai berikut.

1. Tahap *preprocessing* dapat disesuaikan atau ditambahkan untuk mengambil informasi dari *emoticon* dan *hashtag* yang terkandung dalam *dataset*.
2. Penelitian selanjutnya dapat menggunakan kamus yang sesuai dengan topik *dataset*, atau menggunakan teknik pelabelan lain untuk mendapatkan hasil *labeling* yang lebih baik, sehingga akan mendapatkan nilai akurasi yang lebih baik.
3. Percobaan dapat dilakukan dengan menambahkan variasi parameter yang terkait atau menambah jenis perbandingan *splitting* agar dapat lebih baik dalam menganalisis pengaruh masing-masing parameter terhadap *dataset*.

Referensi

- Ahuja, R., Chug, A., Kohli, S., Gupta, S. dan Ahuja, P. (2019) 'The Impact of Features Extraction on the Sentiment Analysis', *Procedia Computer Science*, 152, pp. 341–348. doi:10.1016/J.PROCS.2019.05.008.
- Berrar, D. (2018) 'Cross-Validation Call for Papers for Machine Learning journal: Machine Learning for Soccer View project Cross-validation'. doi:10.1016/B978-0-12-809633-8.20349-X.
- Darujati, C. dan Gumelar, A.B. (2012) 'Pemanfaatan teknik supervised untuk klasifikasi teks bahasa indonesia', *Jurnal Bandung Text Mining*, 16(1), pp.5-1.
- Gonçalves, L., Subtil, A., Oliveira, M.R. dan de Zea Bermudez, P. (2014) 'ROC curve estimation: An overview', *REVSTAT-Statistical journal*, 12(1), pp.1-20.
- Gurusamy, V., Kannan, S. dan Professor, A. (2014) 'Preprocessing Techniques for Text Mining Agent based distributed framework for Big data View project Multilingual Natural Language Processing View project Preprocessing Techniques for Text Mining'. Available at: <https://www.researchgate.net/publication/273127322> (Accessed: 1 May 2022).
- Hapsari, Y (2022). Online at <https://github.com/yashmine28/tugas-akhir>.
- Hayati, H. dan Alifi, M.R. (2021) 'Analisis Sentimen Pada Tweet Terkait Vaksin Covid-19 Menggunakan Metode Support Vector Machine', *Jurnal Teknologi Terapan* |, 7(2), pp. 110–119. doi:10.31884/JTT.V7I2.349.
- Jabbar, H. and Khan, R.Z. (2015) 'Methods to avoid overfitting and under-fitting in supervised machine learning (comparative study)', *Computer Science, Communication and Instrumentation Devices*, 70.
- Koto, F. dan Rahmaningtyas, G.Y., (2017) 'Inset lexicon: Evaluation of a word list for Indonesian sentiment analysis in microblogs', *International Conference on Asian Language Processing (IALP)*, (pp. 391-394), IEEE.
- Liu, Y., Ott, M., Goyal, N., Du, J., Joshi, M., Chen, D., Levy, O., Lewis, M., Zettlemoyer, L. dan Stoyanov, V. (2019) 'RoBERTa: A Robustly Optimized BERT Pretraining Approach'. doi:10.48550/arxiv.1907.11692.
- Nurhuda, F., Sihwi, S.W. dan Doewes, A. (2016) 'Analisis Sentimen Masyarakat terhadap Calon Presiden Indonesia 2014 berdasarkan Opini dari Twitter Menggunakan Metode Naive Bayes Classifier', *ITSMART: Jurnal Teknologi dan Informasi*, 2(2), pp. 35–42. doi:10.20961/ITSMART.V2I2.630.
- Oktasari, L., Chrisnanto, Y.H. dan Yuniarti, R. (2016) 'TEXT MINING DALAM ANALISIS SENTIMEN ASURANSI MENGGUNAKAN METODE NAÏVE BAYES CLASSIFIER | Oktasari | Prosiding SNST Fakultas Teknik', *Prosiding SNST Fakultas Teknik*, 1(1). Available at: https://www.publikasiilmiah.unwahas.ac.id/index.php/PROSIDING_SNST_FT/article/view/1506 (Accessed: 23 April 2022).
- Pratama, B.P. dan Pamungkas, S.A. (2016) 'Analisis Kinerja Algoritma Levenshtein Distance Dalam Mendeteksi Kemiripan Dokumen Teks', *Jurnal "LOGIK@"*, 6(2), pp. 131–143.
- Pustaka, K., Makmun, A. dan Hazhiyah, S.F. (2020) 'TINJAUAN TERKAIT PENGEMBANGAN VAKSIN COVID 19', *Molucca Medica*, pp. 52–59. doi:10.30598/MOLMED.2020.V13.I2.52.
- Rachman, F.F. dan Pramana, S. (2020) 'Analisis Sentimen Pro dan Kontra Masyarakat Indonesia tentang Vaksin COVID-19 pada Media Sosial Twitter', *Indonesian of Health Information Management Journal (INOHIM)*, 8(2), pp. 100–109. doi:10.47007/INOHIM.V8I2.223.
- Rahmad, F., Suryanto, Y. dan Ramli, K. (2020) 'Performance comparison of anti-spam technology using confusion matrix classification', In *IOP Conference Series: Materials Science and Engineering* (Vol. 879, No. 1, p.

012076). IOP Publishing.

- Rezaeian, N. and Novikova, G. (2020) 'Persian Text Classification using naive Bayes algorithms and Support Vector Machine algorithm', Indonesian Journal of Electrical Engineering and Informatics (IJEI), 8(1), pp. 178–188. doi:10.52549/IJEI.V8I1.1696.
- Salsabila, N.A., Winatmoko, Y.A., Septiandri, A.A. dan Jamal, A. (2019) 'Colloquial Indonesian Lexicon', Proceedings of the 2018 International Conference on Asian Language Processing, IALP 2018, pp. 226–229. doi:10.1109/IALP.2018.8629151.
- Saputra, N., Adji, B.T. dan Permanasari, A.E. (2015) 'Analisis Sentimen Data Presiden Jokowi Dengan Preprocessing Normalisasi Dan Stemming Menggunakan Metode Naive Bayes Dan SVM', Jurnal Dinamika Informatika, 5(1).Sembodo, J.E., Setiawan, E.B. and Baizal, A. (2016) 'Data Crawling Otomatis pada Twitter Social Network Analysis View project', Indonesia Symposium on Computing [Preprint]. doi:10.21108/INDOSC.2016.111.
- Wibowo, R., Wibowo, R.K. dan Hastuti, K. (2016) 'PENERAPAN ALGORITMA WINNOWING UNTUK MENDETEKSI KEMIRIPAN TEKS PADA TUGAS AKHIR MAHASISWA', Techno.Com, 15(4), pp. 303–311. doi:10.33633/tc.v15i4.1271.
- Wijaya, A.P. dan Santoso, H.A. (2016) 'Naive Bayes Classification pada Klasifikasi Dokumen Untuk Identifikasi Konten E-Government', Journal of Applied Intelligent System, 1(1), pp. 48–55. doi:10.33633/JAIS.V1I1.1032.