

## Pengaruh *Stemming* Terhadap Performa Klasifikasi Sentimen Masyarakat Tentang Kebijakan *New Normal*

Rama Ulgasesa<sup>a1</sup>, Arif Bijaksana Putra Negara<sup>a2</sup>, Tursina<sup>a3</sup>

<sup>a</sup>Program Studi Sarjana Informatika, Universitas Tanjungpura  
Jl. Prof. Dr. H. Hadari Nawawi, Kota Pontianak, 78115

<sup>1</sup>rama.ulgasesa@student.untan.ac.id

<sup>2</sup>arifbpn@informatika.untan.ac.id

<sup>3</sup>tursina@informatika.untan.ac.id

### Abstrak

Banyaknya pengguna twitter dapat dimanfaatkan untuk mengetahui sentimen masyarakat tentang kebijakan dan penanganan yang dilakukan oleh pemerintah terhadap Covid-19, salah satunya kebijakan mengenai adaptasi kebiasaan baru atau new normal. Untuk melakukan hal itu, bisa digunakan salah satu fungsi dari text mining, yaitu klasifikasi text. Sebelum model klasifikasi text dibuat, text akan melalui tahapan *preprocessing*. Setiap tahapan memiliki pengaruh terhadap hasil evaluasi klasifikasi text yang akan dilakukan. Tujuan dari penelitian ini adalah mengetahui perbandingan performa klasifikasi menggunakan proses *stemming* dan tanpa *stemming* pada *dataset* melalui tahapan *preprocessing* dan algoritma klasifikasi yang performanya paling baik jika pada *preprocessing* dilakukan *stemming*. Metode klasifikasi menggunakan metode Naive Bayes dan Logistic Regression. Hasil percobaan menunjukkan pengaruh model klasifikasi Naive Bayes dan Logistic Regression terhadap penggunaan tahapan *stemming* pada *preprocessing* dengan akurasi sebesar 74,11% dan 73,57%, sedangkan tanpa melakukan *stemming* mendapatkan akurasi masing -masing sebesar 78,47% dan 76,29% . Dari hasil pengujian model, dapat dilihat bahwa tanpa tahapan *stemming* pada *preprocessing* memiliki tingkat akurasi yang lebih unggul pada masing-masing model sebesar 4,36% dan 2,72% dibandingkan dengan penerapan tahapan *stemming* pada *preprocessing*. Hasil tersebut menunjukkan bahwa penggunaan tahapan *stemming* dapat menurunkan akurasi klasifikasi. Algoritma klasifikasi yang tidak banyak pengaruhnya jika pada *preprocessing* dilakukan *stemming* adalah Logistic Regression karena tingkat penurunan akurasi lebih tipis dari algoritma Naive Bayes.

**Kata kunci:** *stemming, preprocessing, klasifikasi teks, machine learning, Naive Bayes, Logistic regression*

## The Impact of Stemming on the Classifications Performance of Public Sentiments About New Normal Policy

### Abstract

A large number of Twitter users can be used to find out public sentiment about the policies and handling carried out by the government against Covid-19, one of which is the policy regarding the adaptation of new habits or the new normal. To do this, one of the functions of text mining can be used, namely text classification. Before the text classification model is created, the text will go through a preprocessing stage. Each stage influences the results of the evaluation of the text classification to be carried out. The purpose of this study is to determine the comparison of classification performance using stemming and non-stemming processes on datasets through preprocessing stages and classification algorithms whose performance is best if stemming is carried out in preprocessing. The classification method uses the Naive Bayes method and Logistic Regression. The experimental results show the effect of the Naive Bayes classification model and Logistic Regression on the use of stemming stages in preprocessing with the accuracy of 74,11% and 73,57%, while without stemming the accuracy of 78,47% and 76,29%, respectively. From the results of model testing, it can be seen that without the stemming stage in preprocessing, the accuracy rate is superior to each model by 4,36% and 2,72% compared to the application of the stemming stage in preprocessing. These results indicate that the use of stemming stages can reduce classification accuracy. The classification algorithm that does not have much influence if the preprocessing is done by stemming is Logistic Regression because the level of decrease in accuracy is thinner than the Naive Bayes algorithm.

**Keywords:** *stemming, preprocessing, Text Classification, machine learning, Naive Bayes, Logistic regression*

## I. PENDAHULUAN

Indonesia pertama kali mengkonfirmasi kasus Covid-19 pada Senin 2 Maret 2020 melalui pengumuman yang disampaikan langsung oleh Presiden Joko Widodo (Jokowi). Sejak terjadinya wabah tersebut di Indonesia, Presiden Jokowi menetapkan Peraturan Presiden (Perpres) Nomor 82 Tahun 2020 tentang Komite Penanganan Corona Virus Disease 2019 (Covid-19) dan Pemulihan Ekonomi Nasional yang ditandatangani pada 20 Juli 2020. Pemerintah turut aktif mengampanyekan kebijakan dan tindakan yang telah dilakukan melalui media sosial. Dalam hal ini pemerintah melalui kantor staf presiden juga mengeluarkan protokol komunikasi publik. Pemerintah Indonesia juga meluncurkan situs resmi penanganan Covid-19 untuk masyarakat dan memanfaatkan jejaring sosial (whatsapp, facebook, instagram, twitter, dan lain-lain) agar memberikan edukasi kepada masyarakat terkait Covid-19. Pemanfaatan media sosial dilingkup pemerintah juga dipertegas melalui peraturan menteri pendayagunaan aparatur negara dan reformasi birokrasi nomor 83 tahun 2012. Salah satu fungsi tugas dari humas pemerintah adalah menyebarkan informasi, mengkomunikasikan kebijakan, rencana kerja, dan capaian kinerja kepada masyarakat luas, melalui media tradisional, media konvensional, dan media baru.

Selain peraturan yang dikeluarkan, pemerintah memanfaatkan platform media sosial untuk menyebarkan informasi terkait langkah preventif pencegahan penularan Covid-19. Berdasarkan riset yang dilakukan oleh platform manajemen media sosial HootSuite dan agensi marketing sosial We Are Social, bahwa ada 175,4 juta pengguna internet di Indonesia dan dari jumlah tersebut 160 juta adalah pengguna aktif media sosial. Indonesia mempunyai situasi menguntungkan terkait penggunaan internet dan media sosial dalam hal memerangi Covid-19. Diberlakukannya *new normal* atau adaptasi kebiasaan baru merupakan salah satu kebijakan yang diambil pemerintah pada masa pandemi Covid-19 di Indonesia. Sejak diterapkan di beberapa daerah di Indonesia, kebijakan ini juga menimbulkan banyak opini di masyarakat. Ketidakpuasan terhadap kebijakan yang dibuat oleh pemerintah akan berdampak pada sikap publik yang kemudian menimbulkan permasalahan baru. Permasalahan baru terjadi salah satu sebabnya karena adanya kerugian akibat kebijakan yang dibuat atau sebuah perasaan bahwa hak-haknya belum terpenuhi dalam kebijakan yang dibuat.

Media sosial yang banyak digunakan untuk beropini salah satunya adalah twitter. Twitter merupakan media sosial yang banyak digunakan di Indonesia dan memiliki persebaran serta distribusi informasi yang sangat cepat. Berdasarkan riset per November 2019 yang dilakukan We Are Social, telah tercatat bahwa 78 Juta orang Indonesia yang menggunakan media sosial Twitter [1]. Pengguna Twitter yang berasal dari berbagai kalangan dan disiplin ilmu menyebabkan aliran komunikasi dan opini di media ini menjadi lebih beragam, salah satunya adalah kritikan dan komentar mengenai kebijakan pemerintah [2]. Banyaknya pengguna twitter dapat dimanfaatkan untuk

mengetahui sentimen masyarakat terhadap kebijakan dan penanganan yang dilakukan oleh pemerintah terhadap Covid-19, salah satunya kebijakan mengenai adaptasi kebiasaan baru atau *new normal*.

Polaritas positif atau negatif suatu opini dapat ditentukan secara manual, namun seiring dengan bertambahnya jumlah pengguna internet maka begitu pula dengan sumber opininya, sehingga membutuhkan banyak waktu dan tenaga untuk mengklasifikasikan polaritas opini tersebut. Oleh karena itu, penerapan teknik pembelajaran mesin telah diusulkan untuk mengklasifikasikan polaritas opini dari berbagai sumber data tersebut. Untuk melakukan hal itu, bisa digunakan salah satu fungsi dari *text mining*, yaitu klasifikasi *text* [3][4].

Ada beberapa teknik klasifikasi teks, seperti Naïve Bayes Classifier, Decision Trees, dan Support Vector Machines [5]. Salah satu metode klasifikasi teks yang paling umum digunakan saat ini adalah metode Naïve Bayes Classifier. Naïve Bayes dipilih karena kecepatan dan akurasi yang tinggi ketika diaplikasikan dalam basis data yang besar dan data yang heterogen [6]. Metode Naïve Bayes Classifier juga memiliki beberapa kelebihan antara lain, sederhana, cepat dan berakurasi tinggi [7]. Hal ini sesuai dengan penelitian yang dilakukan oleh Buntoro, G. A [8] dan Sa'rony, A., Adikara, P. P., dan Wihandika, R. C.[9] penggunaan algoritma Naïve Bayes memiliki nilai lebih dari 90% pada akurasi dalam hal klasifikasi *text*.

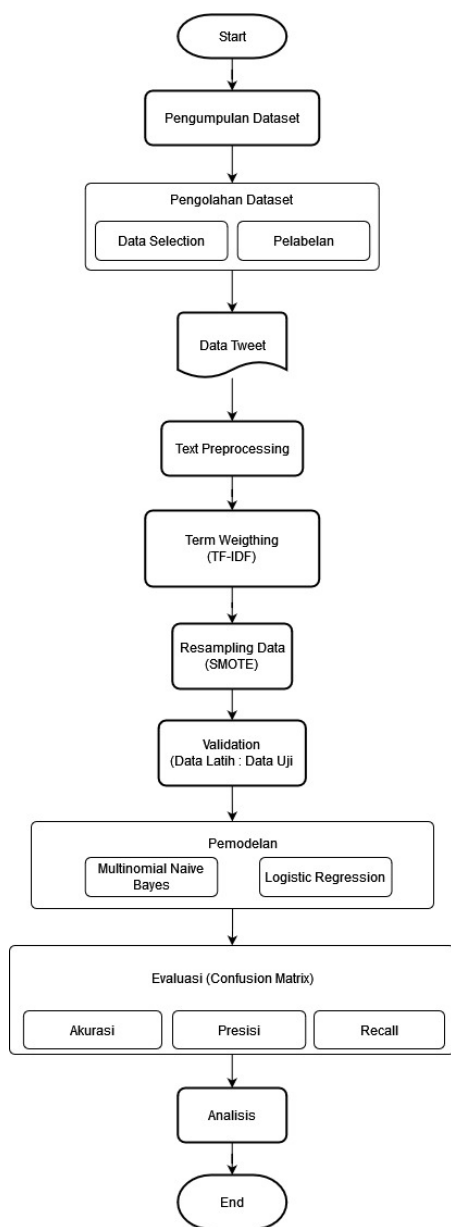
Sebelum dilakukan proses klasifikasi *text*, *text* akan melalui tahapan *preprocessing*. Tahap *preprocessing* merupakan proses untuk mempersiapkan data mentah sebelum dilakukan proses lain [10]. Sejalan dengan penelitian yang dilakukan Dr.S.Vijayarani et al [11] yang membagi tahapan *Preprocessing* menjadi tiga tahapan kunci yaitu stop words removal, *stemming* dan TF/IDF algorithms. Setiap tahapan memiliki pengaruh terhadap hasil evaluasi klasifikasi *text* yang akan dilakukan, sebagaimana dilakukan penelitian oleh Adriani et al [12], *stemming* dapat meningkatkan recall namun dapat menurunkan precision karena berkurangnya informasi dari kata yang telah di-stem. Penelitian lain yang juga melihat pengaruh dari penggunaan *stemming* adalah oleh Agastya, I. M. A. [13] dan Narulita, L. F. [14] yang masing-masing hasil pengujian disimpulkan bahwa penggunaan *stemming* dengan algoritma SVM dapat meningkatkan dan menurunkan akurasi tergantung jumlah data yang digunakan dan penggunaan *stemming* dengan algoritma KNN dapat berperan penting meningkatkan akurasi. Penelitian selanjutnya oleh Atwan, Jaffar et al [15] yang melakukan implementasi klasifikasi teks untuk teks arab dengan dan tanpa *stemmer* dengan hasil bahwa Naïve Bayes dengan light *stemmer* mencapai hasil yang lebih baik daripada Naïve Bayes tanpa *stemmer*.

Berdasarkan latar belakang yang telah dipaparkan, maka dibutuhkan analisis untuk melihat pengaruh dari *stemming* pada performa pengklasifikasian teks sentimen. Sumber data menggunakan kalimat *tweet* postingan media sosial twitter tentang kebijakan *new normal* di Indonesia. Metode klasifikasi menggunakan metode Naive Bayes dan

Logistic Regression. Metode Logistic regression dipilih sebagai pembanding karena dari penelitian yang dilakukan oleh Saputri, Mahendra, dan Adriani [16], Logistic regression mendapat nilai F1-score tertinggi dibandingkan dengan SVM dan Random Forest pada pengklasifikasian teks emosi. *Stemmer* yang digunakan adalah *stemmer* Sastrawi bahasa Indonesia. Pada penelitian ini kalimat *tweet* akan dibagi menjadi 3 kelas yaitu negatif, positif, dan netral. Analisis dilakukan terhadap pengaruh penggunaan *stemmer* atau tanpa *stemmer* terhadap akurasi klasifikasi sentiment yang menggunakan klasifikasi Naïve Bayes dan Logistic Regression.

## II. METODOLOGI

Adapun tahapan-tahapan penelitian yang akan dilakukan dapat dilihat pada Gambar 1. Tahapan yang dilakukan pada penelitian ini sebagai berikut.



Gambar 1. Tahapan-tahapan metodologi penelitian

### A. Pengumpulan Data

Pengumpulan data terbagi menjadi 2 bagian yaitu koleksi data kalimat *tweet* dan koleksi kamus kata. Koleksi data kalimat *tweet* yang digunakan berasal dari proses scraping melalui website twitter dengan kata kunci "kebijakan *new normal*", "kebijakan adaptasi kebiasaan baru", "adaptasi kebiasaan baru", "*new normal*" "#*newnormal*", "#*adaptasikebiasaanbaru*" "#*tatananhidupbaru*". Data diambil menggunakan library python Twitter Scrapper, dimulai dari awal penetapan kebijakan *new normal* 14 juni 2020 sampai puncak peningkatan kasus pertama sejak diberlakukan *new normal* 29 agustus 2020. Dataset ini kemudian diseleksi menjadi sebanyak 1500 *tweets* yang sesuai studi kasus *new normal*. Jumlah data setiap kelas dapat dilihat pada Tabel I. Kamus kata yang digunakan adalah kamus kata slangword dari penelitian Sri Juniarsih [17] dan Repositori Github tentang Formalization Dictionary Resources oleh Panggi Libersa [18].

Tabel I

JUMLAH DATASET PERKELAS

Label	Jumlah
Negatif	611
Positif	487
Netral	402

### B. Pengolahan Dataset

Dataset yang didapatkan akan mengalami proses pengolahan yang terdiri dari seleksi data dan pelabelan data. Seleksi data merupakan proses pemilihan data yang relevan terhadap penelitian. Selanjutnya pelabelan data merupakan proses memberikan label pada dataset yang sudah di seleksi, Adapun label yang digunakan terbatas pada orientasi opini positif, negatif, dan netral.

### C. Preprocessing

*Preprocessing* dilakukan untuk mempersiapkan data yang sebelumnya tidak terstruktur menjadi terstruktur sehingga dapat digunakan pada tahapan selanjutnya. *Preprocessing* data pada penelitian ini meliputi *case folding*, *cleaning*, tokenization, normalisasi kata, *Stopword Removal*, dan *stemming*.

- 1) *Case Folding*: *Case folding* adalah proses mengubah semua huruf kapital dalam teks menjadi huruf kecil
- 2) *Cleaning*: *Cleaning* adalah proses menghilangkan noisy pada teks seperti tag html, link, dan karakter yang tidak diperlukan atau tidak memiliki hubungan dengan informasi yang ada pada dokumen.
- 3) *Tokenization*: *Tokenization* (tokenisasi) Tokenisasi adalah proses pemotongan sebuah kalimat menjadi kata-kata (token).
- 4) *Normalisasi Kata*: *Normalisasi* adalah proses memperbaiki kata yang tidak baku, kesalahan dalam pengetikan, dan kata yang disingkat dalam

pengetikan. Cara yang diimplementasikan pada kode program adalah dengan mengecek setiap kata dan mencocokkan dengan kamus kata, apabila kata tersebut cocok maka akan diganti dengan kata yang memiliki makna sebenarnya.

- 5) *Stopword Removal*: *Stopword Removal* adalah menghilangkan kata-kata yang tidak merepresentasikan isi dari dokumen teks.
- 6) *Stemming*: *Stemming* adalah proses memetakan kata menjadi bentuk kata dasar. Proses *stemming* dilakukan dengan menghilangkan imbuhan-imbuhan dalam kata sehingga mengembalikan kata menjadi kata dasarnya.

**D. Term Weighting**

Term Weighting atau pembobotan kata merupakan kegiatan mengubah data teks menjadi numerik. Untuk mendapatkan nilai bobot setiap kata pada data yang digunakan dilakukan pembobotan kata term frequency-inverse document frequency.

**E. Resampling Data**

Resampling data merupakan proses menyeimbangkan distribusi data pada masing-masing kelas dataset. Pendekatan yang digunakan untuk proses resampling data pada penelitian ini adalah teknik SMOTE (*Synthetic Minority Oversampling Technique*). SMOTE bekerja dengan cara mereplikasi data dari kelas minor pada dataset. Data yang tidak seimbang pada kelas mayoritas dan minoritas dapat berpengaruh pada proses klasifikasi, sebagaimana penelitian yang dilakukan oleh Sutoyo, E dan Fadlurrahman, M.A [19] teknik SMOTE menghasilkan akurasi yang lebih baik dan juga dijelaskan dalam Mutmainah, Siti [20] data yang tidak seimbang hanya mengandalkan kelas mayoritas pada pengklasifikasian.

**F. Pemodelan**

Pendekatan yang digunakan dalam melakukan klasifikasi pada penelitian ini adalah metode *Naïve Bayes* dan *Logistic regression*. Pemodelan Naive Bayes dilakukan dengan cara memanggil fungsi Naive Bayes dari library `sklearn.naive_bayes.MultinomialNB`. Pemodelan *Logistic regression* dilakukan dengan cara memanggil fungsi *Logistic regression* dari library `sklearn.linear_model.LogisticRegression`.

**G. Evaluasi**

Dalam permasalahan klasifikasi pengukuran yang biasa digunakan adalah *precision*, *recall* dan *akurasi* [21]. *Precision*, *recall* dan *akurasi* dapat dihitung dengan menggunakan persamaan (1)(2)(3). *Confusion matrix* merupakan alat pengukuran yang digunakan untuk evaluasi model klasifikasi [17].

$$Precision = \frac{TP}{TP + FP} \tag{1}$$

$$Recall = \frac{TP}{TP + FN} \tag{2}$$

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \tag{3}$$

III. HASIL DAN PEMBAHASAN

Setelah melalui tahapan-tahapan dari metodologi penelitian, berikut hasil dari tahapan tersebut.

**A. Hasil Implementasi Text Preprocessing**

Hasil *sample* dari keseluruhan dataset yang sudah melalui tahapan *text preprocessing* ditampilkan pada Tabel II.

TABEL II  
HASIL TEXT PREPROCESSING

Text	Text Preprocessing
kebijakan <i>new normal</i> , kebijakan yg paling cacat secara ilmiah untuk Indonesia sendiri. Data semakin naik yg dites dikit, belum ada evaluasi lebih lanjut dari pemerintah pusat nih	Tanpa Preprocessing
kebijakan <i>new normal</i> , kebijakan yg paling cacat secara ilmiah untuk indonesia sendiri. data semakin naik yg dites dikit, belum ada evaluasi lebih lanjut dari pemerintah pusat nih	Case folding
kebijakan <i>new normal</i> kebijakan yg paling cacat secara ilmiah untuk indonesia sendiri data semakin naik yg dites dikit belum ada evaluasi lebih lanjut dari pemerintah pusat nih	Cleaning
kebijakan <i>new normal</i> kebijakan yang paling cacat secara ilmiah untuk indonesia sendiri data semakin naik yang dites dikit belum ada evaluasi lebih lanjut dari pemerintah pusat ini	Normalisasi
['kebijakan', 'new', 'normal', ', ', 'kebijakan', 'yang', 'paling', 'cacat', 'secara', 'ilmiah', 'untuk', 'indonesia', 'sendiri', ', ', 'data', 'semakin', 'naik', 'yang', 'dites', 'dikit', ', ', 'belum', 'ada', 'evaluasi', 'lebih', 'lanjut', 'dari', 'pemerintah', 'pusat', 'ini']	Tokenize

Text	Text Preprocessing
['kebijakan', 'new', 'normal', ', ', 'kebijakan', 'cacat', 'ilmiah', 'indonesia', ', ', 'data', 'dites', 'dikit', ', ', 'evaluasi', 'pemerintah', 'pusat']	Stopword Removal
bijak new normal bijak cacat ilmiah indonesia data tes dikit evaluasi perintah pusat	Stemming

B. Hasil Term Weighting

Hasil dari keseluruhan dataset yang telah melalui tahapan term weighting ditampilkan pada Gambar 2.

(0, 3465)	0.2788500851843863
(0, 3241)	0.1623951552357371
(0, 1141)	0.34214576559082777
(0, 933)	0.3758860166647649
(0, 4292)	0.3466138696739375
(0, 834)	0.31489476251774245
(0, 1626)	0.15617399044284733
(0, 1595)	0.39323089070737094
(0, 653)	0.43502199168678596
(0, 2935)	0.06097856249270528
(0, 2824)	0.06097856249270528
(0, 500)	0.20619113192737037
(1, 774)	0.13927653203605936
(1, 3114)	0.3933167487542213
(1, 637)	0.3398500967813282
(1, 4486)	0.3654707849833938
(1, 2203)	0.3654707849833938
(1, 3827)	0.30558358166946586
(1, 1293)	0.41541933363920885
(1, 3685)	0.3776347152102685
(1, 1626)	0.14120170321223643
(1, 2935)	0.0551325919187213
(1, 2824)	0.0551325919187213
(1, 500)	0.09321186880365413
(2, 2259)	0.5531076677958388

Gambar 2. Hasil proses TF-IDF

Output pada gambar 2 merupakan bobot angka yang dihasilkan dari proses TF-IDF yang kemudian dilanjutkan ke tahapan pemodelan dan evaluasi.

C. Hasil Resampling Data

Dataset dari pengumpulan dan pengolahan data pada masing-masing kelasnya memiliki jumlah data yang tidak seimbang seperti yang dapat dilihat pada Tabel I. Kelas negatif memiliki jumlah data yang lebih dominan daripada kelas netral dan positif. Resampling data dengan teknik SMOTE adalah ditujukan agar jumlah data antar masing-masing kelasnya seimbang. Teknik SMOTE akan mereplikasi data pada kelas minoritas sehingga mengimbangi data pada kelas mayoritas. Hasil dari resampling data ditampilkan pada Tabel III.

TABEL III  
JUMLAH DATASET PERKELAS SETELAH RESAMPLING

Label	Jumlah
Negatif	611
Positif	487
Netral	402

D. Hasil Evaluasi Model

Pengujian dilakukan dengan menggunakan dua skenario pengujian. Skenario pertama adalah dataset tanpa dilakukan proses stemming pada preprocessing. Skenario kedua adalah dataset dengan dilakukan proses stemming pada preprocessing.

Pengujian pada penelitian ini dilakukan dengan membandingkan tiga data yang dibagi secara split dengan tiga rasio berbeda. Tiga rasio tersebut dimulai dari 20% data uji dan 80% data latih, 30% data uji dan 70% data latih serta 40% data uji dan 60% data latih. Untuk menguji performa klasifikasi yang dihasilkan, maka dilakukan pengujian testing menggunakan confusion matrix. Confusion matrix digunakan untuk mendapatkan perhitungan nilai akurasi, precision, recall dan F1-score.

Pada pengujian data training dilakukan secara Cross validation dengan K-Fold = 10. 10-Fold Cross validation digunakan dalam membagi dataset menjadi 10-buah partisi yang berukuran sama, kemudian setiap partisi digunakan sebagai data uji dan sisanya untuk data latih. Pada cross validation akan dilakukan 10 kali eksperimen dengan masing-masing skenario, hasil pengukuran adalah rata-rata dari 10 kali eksperimen tersebut.

Hasil pengujian training dan pengujian testing dari masing-masing model dengan dua skenario tersebut dapat dilihat pada Tabel IV dan Tabel V.

TABEL IV  
HASIL PENGUJIAN TRAINING DAN PENGUJIAN TESTING NAÏVE BAYES

Naïve Bayes			
Rasio Data	Skenario	Pengujian Training (%)	Pengujian Testing (%)
8:2	1	68,55	78,47
	2	67,80	74,11
7:3	1	66,56	72,91
	2	65,71	70,36
6:4	1	62,97	71,39
	2	62,51	69,07

TABEL V  
HASIL PENGUJIAN *TRAINING* DAN PENGUJIAN *TESTING* LOGISTIC REGRESSION

Logistic Regression			
Rasio Data	Skenario	Pengujian <i>Training</i> (%)	Pengujian <i>Testing</i> (%)
8:2	1	69,58	76,29
	2	69,17	73,57
7:3	1	68,82	72,55
	2	68,59	69,64
6:4	1	65,61	72,34
	2	65,33	70,84

E. Analisis Hasil Evaluasi Model

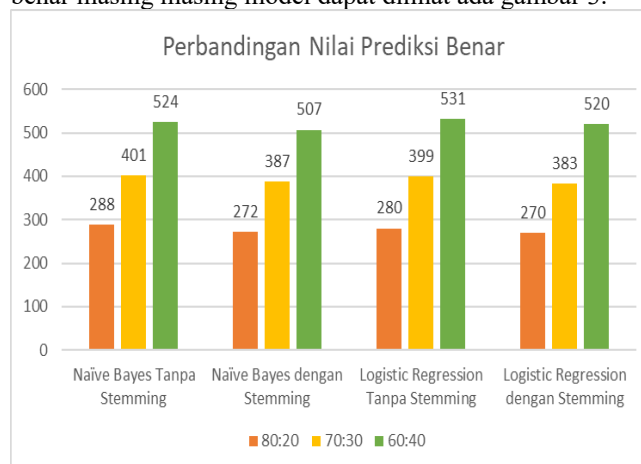
Hasil pengujian dari tiap skenario pada masing-masing model dan rasio data mengalami peningkatan hasil pengujian *testing* terhadap hasil pengujian *training*. Hal ini menunjukkan model yang dibuat tidak ada yang mengalami *overfitting*.

Hasil evaluasi berdasarkan rasio pembagian *dataset* menunjukkan bahwa model dengan rasio data latih dan data uji 80% dan 20% memperoleh hasil performa tertinggi. Dari evaluasi model yang telah dilakukan, model Naive Bayes dengan *dataset* tanpa *stemming* menghasilkan akurasi sebesar 78,47% dan model Logistic Regression dengan *dataset* tanpa *stemming* menghasilkan akurasi sebesar 76,29%. Model Naive Bayes dengan *dataset* dilakukan *stemming* menghasilkan akurasi sebesar 74,11% dan model Logistic Regression dengan *dataset* dilakukan *stemming* menghasilkan akurasi sebesar 73,57%. Dari hasil pengujian *testing* model, dapat dilihat bahwa *dataset* tanpa tahapan *stemming* pada *preprocessing* memiliki tingkat akurasi yang lebih unggul pada masing-masing model sebesar 4,36% dan 2,72% dibandingkan dengan *dataset* dilakukan tahapan *stemming* pada *preprocessing*.

Hasil pengujian *testing* dari kedua skenario menunjukkan adanya pengaruh performa model klasifikasi Naive Bayes dan Logistic Regression terhadap penggunaan tahapan *stemming* pada *preprocessing*. Pada rasio data latih dan data uji 80% dan 20% menunjukkan penurunan akurasi sebesar 4,36% dan 2,72% dengan *dataset* dilakukan *stemming*. Pada rasio data latih dan data uji 70% dan 30% menunjukkan penurunan akurasi sebesar 2,55% dan 2,91% dengan *dataset* dilakukan *stemming*. Pada rasio data latih dan data uji 60% dan 40% menunjukkan penurunan akurasi sebesar 2,32% dan 1,50% dengan *dataset* dilakukan *stemming*. Dari perbandingan ketiga rasio data latih dan data uji, proses *stemming* konsisten memengaruhi penurunan akurasi.

Pada rasio data latih dan data uji 80% dan 20% pada model Naive Bayes skenario pertama, model memprediksi kelas opini dengan benar sebanyak 288 data. Sedangkan, model Naive Bayes skenario kedua model memprediksi kelas opini dengan benar sebanyak 272 data. Kemudian rasio data latih dan data uji 70% dan 30% pada model Naive Bayes skenario pertama, model memprediksi kelas

opini dengan benar sebanyak 401 data. Sedangkan, model Naive Bayes skenario kedua model memprediksi kelas opini dengan benar sebanyak 387 data. Pada rasio data latih dan data uji 60% dan 40% pada model Naive Bayes skenario pertama, model memprediksi kelas opini dengan benar sebanyak 524 data. Sedangkan, model Naive Bayes skenario kedua model memprediksi kelas opini dengan benar sebanyak 507 data. Pada rasio data latih dan data uji 80% dan 20% pada model Logistic Regression skenario pertama, model memprediksi kelas opini dengan benar sebanyak 280 data. Sedangkan, model Logistic Regression skenario kedua model memprediksi kelas opini dengan benar sebanyak 270 data. Kemudian rasio data latih dan data uji 70% dan 30% pada model Logistic Regression skenario pertama, model memprediksi kelas opini dengan benar sebanyak 399 data. Sedangkan, model Naive Bayes skenario kedua model memprediksi kelas opini dengan benar sebanyak 383 data. Pada rasio data latih dan data uji 60% dan 40% pada model Logistic Regression skenario pertama, model memprediksi kelas opini dengan benar sebanyak 531 data. Sedangkan, model Logistic Regression skenario kedua model memprediksi kelas opini dengan benar sebanyak 520 data. Perbandingan nilai prediksi benar masing masing model dapat dilihat ada gambar 3.



Gambar 3. Grafik Perbandingan Nilai Prediksi Benar

Berkurangnya jumlah prediksi opini dengan benar mengakibatkan terjadinya penurunan pada akurasi yang dihasilkan. Pada perhitungan *confusion matrix*, nilai akurasi dipengaruhi oleh nilai banyaknya prediksi benar (TP dan TN) sebagaimana pada rumus akurasi (3) yang telah dipaparkan pada Bab Metodologi Subbab evaluasi model. Proses *stemming* dapat menurunkan akurasi karena data telah tereduksi yang mengakibatkan variasi kata pada *dataset* berkurang.

Pada Tabel IV dan Tabel V ditunjukkan bahwa rata-rata akurasi dari pengujian *training* pada dataset dengan *stemming* lebih rendah dari dataset tanpa *stemming*. Hal ini sejalan dengan rata-rata jumlah *unique word* pada dataset tanpa *stemming* dan dataset dengan *stemming* dimana rata-rata *unique word* pada dataset yang sudah dilakukan *stemming* mengalami reduksi (pengurangan) kata. Semakin tinggi reduksi (pengurangan) kata, maka tingkat variasi kata yang akan dipelajari oleh mesin pada

masing-masing kelas berkurang. Hal ini terjadi karena pada proses *stemming* suatu kata dengan kata dasar yang serupa akan dianggap sama. Dalam proses klasifikasi teks, semakin banyak kata yang sama dalam masing-masing kelas, semakin sulit juga data tes diprediksi. Pada Tabel VI, Tabel VII, dan Tabel VIII dapat di lihat jumlah *unique word* dari masing-masing *dataset* dengan 2 percobaan, menggunakan *stemming* dan tanpa menggunakan *stemming*. Dari percobaan yang telah dilakukan pada masing-masing rasio, *dataset* mengalami reduksi rata-rata per partisi/fold-nya sebesar 21,1% pada rasio 80:20, sebesar 20,7% pada rasio 70:30, dan sebesar 20,5% pada rasio 60:40. Semakin besar rasio data latih maka reduksi juga akan semakin besar. Hal ini karena *unique word* pada *dataset* penelitian ini berbanding lurus dengan pembagian rasio *dataset*.

TABEL VI  
JUMLAH *UNIQUE WORD* PADA *DATASET* RASIO 80:20

10-Fold Cross Validation		Jumlah			
		Unique Word Tanpa Stemming	Unique Word Stemming	Reduksi	Reduksi (%)
Iterasi	1	4980	3927	1053	21,1
	2	4993	3937	1056	21,1
	3	4981	3945	1036	20,8
	4	4968	3922	1046	21,1
	5	4967	3916	1051	21,2
	6	4942	3907	1035	20,9
	7	4938	3895	1043	21,1
	8	5024	3961	1063	21,2
	9	4996	3936	1060	21,2
	10	4855	3835	1020	21,0
Rata-Rata					21,1

TABEL VII  
JUMLAH *UNIQUE WORD* PADA *DATASET* RASIO 70:30

10-Fold Cross Validation		Jumlah			
		Unique Word Tanpa Stemming	Unique Word Stemming	Reduksi	Reduksi (%)
Iterasi	1	4468	3538	930	20,8
	2	4477	3558	919	20,5
	3	4498	3570	928	20,6
	4	4395	3499	896	20,4
	5	4512	3580	932	20,7
	6	4431	3500	931	21,0
	7	4427	3531	896	20,2
	8	4435	3520	915	20,6

10-Fold Cross Validation		Jumlah			
		Unique Word Tanpa Stemming	Unique Word Stemming	Reduksi	Reduksi (%)
9		4499	3568	931	20,7
10		4483	3543	940	21,0
Rata-Rata					20,7

TABEL VIII  
JUMLAH *UNIQUE WORD* PADA *DATASET* RASIO 60:40

10-Fold Cross Validation		Jumlah			
		Unique Word Tanpa Stemming	Unique Word Stemming	Reduksi	Reduksi (%)
Iterasi	1	4064	3222	842	20,7
	2	4106	3258	848	20,7
	3	4078	3242	836	20,5
	4	4049	3229	820	20,3
	5	4040	3211	829	20,5
	6	4108	3263	845	20,6
	7	4028	3186	842	20,9
	8	4030	3221	809	20,1
	9	4047	3215	832	20,6
	10	4078	3235	843	20,7
Rata-Rata					20,5

IV. KESIMPULAN

Berdasarkan hasil analisis dan pengujian, penggunaan tahapan pada *preprocessing* terhadap metode Naive Bayes dan Logistic regression yang paling baik adalah dengan tanpa tahapan *stemming*. Klasifikasi sentimen kebijakan *new normal* dari media sosial twitter menggunakan metode Naive Bayes dengan melalui tahap *stemming* pada *preprocessing* menghasilkan akurasi sebesar 74,11%. Sedangkan tanpa melalui tahap *stemming* pada *preprocessing* menghasilkan akurasi sebesar 78,47%. Hal ini menunjukkan adanya penurunan akurasi sebesar 4,36%. Klasifikasi sentimen kebijakan *new normal* dari media sosial twitter menggunakan metode Logistic Regression dengan melalui tahap *stemming* pada *preprocessing* menghasilkan akurasi sebesar 73,57%. Sedangkan tanpa melalui tahap *stemming* pada *preprocessing* menghasilkan akurasi sebesar 76,29%. Hal ini menunjukkan adanya penurunan akurasi sebesar 2,72%. Penggunaan tahapan *stemming* pada penelitian ini dapat menurunkan nilai akurasi, nilai presisi, dan nilai *recall*. Penurunan akurasi disebabkan karena pada proses *stemming* suatu kata dengan kata dasar yang serupa akan dianggap sama, sehingga tingkat variasi kata yang akan dipelajari oleh mesin pada masing-masing kelas berkurang. Dalam proses klasifikasi teks, semakin banyak kata yang sama dalam

masing-masing kelas, semakin sulit juga data tes diprediksi. Hal ini juga akan berdampak pada jumlah prediksi opini dengan benar. Jika jumlah prediksi yang benar berkurang maka akurasi akan turun karena nilai akurasi dipengaruhi nilai TP dan TN (prediksi benar) sebagaimana pada persamaan akurasi dalam *confusion matrix*. Algoritma klasifikasi yang tidak banyak pengaruhnya jika pada *preprocessing* dilakukan *stemming* adalah Logistic regression karena tingkat penurunan akurasi lebih tipis dari algoritma Naïve Bayes.

#### DAFTAR PUSTAKA

- [1] M. S. Abdurrahman, "Tantangan penetrasi internet Indonesia pada 2020," Mar. 09, 2020.
- [2] L. N. Pradany and C. Fatichah, "ANALISA SENTIMEN KEBIJAKAN PEMERINTAH PADA KONTEN TWITTER BERBAHASA INDONESIA MENGGUNAKAN SVM DAN K-MEDOID CLUSTERING," *SCAN - J. Teknol. Inf. dan Komun.*, vol. XI, pp. 59–66, 2016.
- [3] F. Nurhuda, S. Widya Sihwi, and A. Doewes, "Analisis Sentimen Masyarakat terhadap Calon Presiden Indonesia 2014 berdasarkan Opini dari Twitter Menggunakan Metode Naive Bayes Classifier," *J. Teknol. Inf. ITS smart*, vol. 2, no. 2, p. 35, 2016, doi: 10.20961/its.v2i2.630.
- [4] S. N. J. Fitriyyah, N. Safriadi, and E. E. Pratama, "Analisis Sentimen Calon Presiden Indonesia 2019 dari Media Sosial Twitter Menggunakan Metode Naive Bayes," *J. Edukasi dan Penelit. Inform.*, vol. 5, no. 3, p. 279, 2019, doi: 10.26418/jp.v5i3.34368.
- [5] L. D. Mahbubah and E. Zuliarso, "Analisa Sentimen Twitter Pada Pilpres 2019 Menggunakan," *Sintak*, pp. 194–195, 2019, [Online]. Available: <https://www.unisbank.ac.id/ojs/index.php/sintak/article/view/7585>.
- [6] D. T. Larose, *Data Mining Methods and Models*. Hoboken, New Jersey: John Wiley & Sons, Inc., 2006.
- [7] R. Mccue, "A Comparison of the Accuracy of Support Vector Machine and Nave Bayes Algorithms In Spam Classification," p. 17, 2009, [Online]. Available: <https://classes.soe.ucsc.edu/cmcs242/Fall09/proj/RitaMcCueReport.pdf>.
- [8] G. A. Buntoro, "Analisis Sentimen Calon Gubernur DKI Jakarta 2017 Di Twitter," *INTEGER J. Inf. Technol.*, vol. 2, no. 1, pp. 32–41, 2017, [Online]. Available: <https://t.co/jrvaMsgBdH>.
- [9] S. Akhmad, P. P. Adikara, and R. C. Wihandika, "Analisis Sentimen Kebijakan Pindahan Ibukota Republik Indonesia dengan Menggunakan Algoritme Term-Based Random Sampling dan Metode Klasifikasi Naïve Bayes," *J. Pengemb. Teknol. Inf. dan Ilmu Komput.*, vol. 3, no. 10, pp. 10086–10094, 2019.
- [10] S. Mujilawati, "Pre-Processing Text Mining Pada Data Twitter," *Semin. Nas. Teknol. Inf. dan Komun.*, vol. 2016, no. Sentika, pp. 2089–9815, 2016.
- [11] M. Kannan, S., Gurusamy, V., Vijayarani, S., Ilamathi, J. & Nithya, "Preprocessing Techniques for Text Mining Preprocessing Techniques for Text Mining," *Int. J. Comput. Sci. Commun. Networks*, vol. 5, no. October 2014, pp. 7–16, 2015.
- [12] J. Asian, H. E. Williams, and S. M. M. Tahaghoghi, "Stemming Indonesian," *Conf. Res. Pract. Inf. Technol. Ser.*, vol. 38, no. September 2018, pp. 307–314, 2005, doi: 10.1145/1316457.1316459.
- [13] I. M. A. Agastya, "Pengaruh Stemmer Bahasa Indonesia Terhadap Peforma Analisis Sentimen Terjemahan Ulasan Film," *J. Tekno Kompak*, vol. 12, no. 1, p. 18, 2018, doi: 10.33365/jtk.v12i1.70.
- [14] L. F. Narulita, "Pengaruh Proses Stemming Pada Kinerja Analisa Sentimen Pada Review Buku," *J. Has. Penelit. LPPM Untag Surabaya*, vol. 3, no. Januari, pp. 55–59, 2018.
- [15] J. Atwan, M. Wedyan, Q. Bsoul, A. Hamadeen, R. Alturki, and M. Ikram, "The Effect of using Light Stemming for Arabic Text Classification," *Int. J. Adv. Comput. Sci. Appl.*, vol. 12, no. 5, pp. 768–773, 2021, doi: 10.14569/IJACSA.2021.0120589.
- [16] M. S. Saputri, R. Mahendra, and M. Adriani, "Emotion Classification on Indonesian Twitter Dataset," *Proc. 2018 Int. Conf. Asian Lang. Process. IALP 2018*, no. November, pp. 90–95, 2019, doi: 10.1109/IALP.2018.8629262.
- [17] S. Juniarsih, E. F. Ripanti, and E. E. Pratama, "Implementasi Naive Bayes Classifier pada Opinion Mining Berdasarkan Tweets Masyarakat Terkait Kinerja Presiden dalam Aspek Ekonomi," *J. Sist. dan Teknol. Inf.*, vol. 8, no. 3, p. 239, 2020, doi: 10.26418/justin.v8i3.39118.
- [18] P. L. J. Akadol, "Formalization Dictionary," 2017. <https://github.com/panggi/pujangga/blob/master/resource/formalization/formalizationDict.txt>.
- [19] E. Sutoyo and M. A. Fadlurrahman, "Penerapan SMOTE untuk Mengatasi Imbalance Class dalam Klasifikasi Television Advertisement Performance Rating Menggunakan Artificial Neural Network," *J. Edukasi dan Penelit. Inform.*, vol. 6, no. 3, p. 379, 2020, doi: 10.26418/jp.v6i3.42896.
- [20] S. Mutmainah, "Penanganan Imbalance Data Pada Klasifikasi Kemungkinan Penyakit Stroke," *J. SNATI*, vol. 1, pp. 10–16, 2021, [Online]. Available: <https://library.uui.ac.id/osr>.
- [21] S. Defiyanti and D. L. Crispina Pardede, "Perbandingan kinerja algoritma id3 dan c4.5 dalam klasifikasi spam-mail," *ReCALL*, 2008.