

# Implementasi Algoritma Naïve Bayes untuk Klasifikasi Konten Twitter dengan Indikasi Depresi

Andre Budiman<sup>1\*</sup>, Alethea Suryadibrata<sup>2</sup>, Julio Christian Young<sup>3</sup>

Informatika, Fakultas Teknik dan Informatika, Universitas Multimedia Nusantara, Tangerang  
Jl. Scientia Boulevard, Curug Sangereng, Kec. Klp. Dua, Tangerang, Banten 15810  
email: <sup>1</sup>andre.budiman@student.umn.ac.id, <sup>2</sup>alethea@umn.ac.id, <sup>3</sup>julio.christian@umn.ac.id

**Abstract** – Depression is one of the most common health problems which has a significant impact for its victims. Depression is characterized or influenced by many aspects of life, including life experiences, work, and social life. In 2018, approximately 6.1% from 267.7 million people experienced mental disorders in Indonesia. The stigma of psychiatric illness and low awareness of undergoing treatment to experts is becoming the underlying factor for such percentages. Nowadays, self-expression is often done via social media posts or comments. Twitter is one of the tools for self-expression in textual form. In this research, we collected the text dataset which contain keywords related to depression from Twitter and do a laborious process involving a psychiatrist for labelling each data to two classes, namely “indicated depression” or “not indicated”. Based on the labelled dataset, we build a predictive model by using Multinomial Naïve Bayes (MNB) and Complement Naïve Bayes (CNB) as a classifier and Term Frequency - Inverse Document Frequency (TF-IDF) as feature extractor. From our experiments, the combination of TF-IDF and MNB are able to achieve the best F-score of 91.30% while the TF-IDF and CNB are able to achieve the best F-score of 91.98%.

**Abstrak** – Depresi merupakan salah satu permasalahan kesehatan yang sangat berdampak bagi para penderitanya. Terdapat begitu banyak faktor depresi, di antaranya pengalaman hidup, pekerjaan, ataupun kehidupan sosial. Pada tahun 2018, diperkirakan 6.1% dari 267.7 juta penduduk di Indonesia mengalami depresi. Hal ini tentunya sangat dipengaruhi oleh stigma masyarakat terkait dengan penyakit kejiwaan dan rendahnya tingkat kesadaran masyarakat untuk melakukan konsultasi kejiwaan. Melalui perkembangan teknologi, saat ini, masyarakat seringkali mengekspresikan dirinya melalui konten-konten di media sosial. Pada penelitian ini dilakukan proses pengumpulan data-data dengan kata kunci yang mengindikasikan gangguan depresi di *platform* Twitter. Kemudian, dengan melibatkan seorang psikiatri, dilakukan proses pelabelan terhadap dataset untuk menentukan apakah konten memiliki label “terindikasi depresi” ataupun “tidak terindikasi”. Berdasarkan dataset tersebut, dikembangkan model prediktif dengan menggunakan metode *Multinomial Naïve Bayes* (MNB) dan *Complement Naïve Bayes* (CNB) sebagai metode klasifikasi dan metode *Term Frequency–Inverse Document Frequency* (TF-IDF) sebagai metode ekstraksi fitur. Berdasarkan eksperimentasi yang telah dilakukan gabungan metode TF-IDF dan MNB berhasil mencapai tingkat F-score sebesar 91.30% sementara gabungan metode TF-IDF dengan CNB berhasil mencapai tingkat performa sebesar 91.98%.

\*) penulis korespondensi: Andre Budiman  
Email: andre.budiman@student.umn.ac.id

**Kata Kunci** – Depresi, Klasifikasi Teks Otomatis, Multinomial Naïve Bayes, Complement Naïve Bayes

## I. PENDAHULUAN

Perkembangan teknologi saat ini berkembang semakin pesat sehingga memudahkan manusia dalam melakukan berbagai macam kegiatan sehari-hari mereka. Teknologi yang perkembangannya begitu pesat saat ini yaitu teknologi informasi dan komunikasi. Salah satunya, pertumbuhan media sosial *online* seperti Twitter yang digunakan untuk menyatakan fakta atau opini dari penggunanya. Menurut Kamus Besar Bahasa Indonesia (KBBI) [1,2], Fakta merupakan hal yang merupakan kenyataan, sedangkan opini merupakan ekspresi yang menggambarkan emosi atau sentimen dari penulis atau pembicaranya. Pengguna aktif harian Twitter di Indonesia saat ini cukup banyak, Menurut Adriansah selaku *Country Industry Head Twitter Indonesia* pada kuartal ke-3 tahun 2019, Indonesia mengalami peningkatan pengguna sebanyak 17% ke angka 145 juta pengguna [3].

Emosi yang diungkapkan dalam bentuk tulisan ini kemudian dapat dihubungkan dengan penyakit mental, khususnya depresi. Depresi merupakan suatu kondisi emosi negatif manusia dan sudah melalui waktu yang cukup panjang. Menurut Riskesdas pada tahun 2018 [4], tercatat 6.1% dari 267.7 juta jiwa di Indonesia mengidap penyakit mental berupa depresi. Selain emosi negatif, terdapat juga beberapa faktor yang dapat menyebabkan seseorang mengidap kondisi depresi, contohnya adalah faktor lingkungan, pengalaman hidup, genetika, dan lainnya.

Dalam penelitian ini akan diujicobakan dan dievaluasi dua macam pengembangan dari metode Naive Bayes Classifier, yaitu Multinomial Naïve Bayes (MNB) dan Complement Naïve Bayes (CNB) untuk mengklasifikasikan *tweet* dengan indikasi depresi. Kemudian, tahapan ekstraksi fitur yang akan digunakan dalam penelitian adalah *Term Frequency – Inverse Document Frequency* (TF-IDF). Dalam penelitian, metode berbasis *Naïve Bayes Classifier* dan TF-IDF dipilih dikarenakan performa yang cendrung baik dan sangat baik pada penelitian-penelitian serupa yang telah dilakukan dalam penelitian lainnya [6, 7, 8, 9, 10, 11, 12].

## II. PENELITIAN YANG TERKAIT

Bagian ini akan metode-metode yang akan digunakan dalam penelitian berdasarkan penelitian-penelitian terdahulu. Adapun pada bagian ini akan dibahas metode *Text Preprocessing*, *Term Frequency – Inverse Document*

Frequency, Naïve Bayes Classifier, Multinomial Naïve Bayes, dan Complement Naïve Bayes.

#### A. Text Preprocessing

Text preprocessing merupakan serangkaian tahapan yang dilakukan pada data teks sebelum tahap ekstraksi fitur [16]. Melalui tahapan *text preprocessing*, sebuah teks dapat memiliki fitur yang lebih representatif sehingga proses utama yang nantinya akan dilakukan terhadap teks dapat memiliki performa yang lebih baik [17]. Terdapat banyak rangkaian yang dapat dilakukan sebagai tahapan *text preprocessing*. Pada subbab ini hanya akan dibahas metode-metode *text preprocessing* yang akan digunakan dalam penelitian. Metode-metode yang akan digunakan di antaranya terdiri dari [16, 17],

1. *Case Folding* : tahap ini merupakan salah satu tahapan normalisasi pada data teks dengan mengubah seluruh huruf kapital menjadi huruf kecil. Selain itu, pada tahap ini, simbol-simbol (tanda baca dan/atau whitespace) yang dianggap tidak relevan terhadap tugas utama yang akan dilakukan akan dihilangkan. Tahap *case folding* akan mengurangi jumlah variasi kemunculan kata dalam data dan membuat sekumpulan dokumen teks menjadi lebih mudah untuk dipelajari oleh mesin.
  2. *Tokenization* : pada tahap ini sebuah dokumen akan dipecah ke dalam sekumpulan token (unit terkecil) yang menyusunnya. Pada bahasa yang menggunakan huruf alfabet tahap ini dapat dilakukan dengan melakukan pemisahan berdasarkan whitespace.
  3. *Stopwords Removal* : tahap ini dilakukan untuk menghilangkan kata-kata yang sangat sering muncul dan bukanlah merupakan penciri identik dari sebuah dokumen. Tahap ini dilakukan guna mengurangi jumlah token yang harus diproses oleh mesin dengan harapan proses pemelajaran yang dilakukan oleh mesin menjadi lebih mudah dan membutuhkan sumber daya komputasi yang lebih sedikit.
  4. *Stemming* : tahap ini bertujuan untuk mengubah setiap kata-kata yang muncul kembali ke kata dasarnya. *Stemming* dilakukan berdasarkan asumsi bahwa bentuk kata ubahan dan kata dasarnya memiliki makna yang serupa. Tahap ini dilakukan guna mengurangi variasi kata yang muncul dalam seluruh dokumen.
- Berdasarkan tahap-tahap yang telah dijabarkan, Tabel 1, menunjukkan contoh hasil output dari setiap tahap yang dilakukan.

TABEL I

CONTOH KELUARAN DARI TAHAP-TAHAP TEXT PREPROCESSING

Tahap	Teks
Teks Awal	Aku kehilangan seorang teman tanpa aku sadari. Aku manusia bodoh.
Case Folding	aku kehilangan seorang teman tanpa aku sadari aku manusia bodoh
Tokenization	[aku, kehilangan, seorang, teman, tanpa, aku, sadari, aku, manusia, bodoh]
Stopwords Removal	[kehilangan, teman, tanpa, sadari, manusia, bodoh]
Stemming	hilang teman tanpa sadar manusia bodoh

#### B. Term Frequency – Inverse Document Frequency

*Term Frequency – Inverse Document Frequency* (TF-IDF) merupakan salah satu metode ekstraksi fitur yang sering digunakan dalam permasalahan pengolahan data teks [11, 12, 13, 14, 15]. Metode TF-IDF mengubah representasi sekumpulan kata-kata ( $b_1, \dots, b_n$ ) dalam sebuah dokumen ( $d_i$ ) menjadi sekumpulan bobot ( $w_{i,1}, \dots, w_{i,n}$ ). Berdasarkan proses dalam metode TF – IDF, bobot-bobot ini merefleksikan derajat kepentingan dari suatu kata ( $b_j$ ) terhadap sebuah dokumen ( $d_i$ ) dalam sekumpulan dokumen ( $D$ ) di mana  $d_i \in D$ . Berdasarkan  $D$ , proses pembobotan untuk kata  $b_j$  untuk dokumen  $d_i$  yang dinotasikan sebagai  $w_{i,j}$  dihitung dengan menggunakan Persamaan 9.

$$w_{i,j} = TF(b_j, d_i) \times IDF(b_j, D)$$

dengan:

1.  $TF(b_j, d_i)$  menunjukkan kemunculan  $b_j$  dalam dokumen  $d_i$ ; dan
2.  $IDF(b_j, D)$  dihitung berdasarkan Persamaan 10.

$$IDF(b_j, D) = \log \frac{1 + n}{1 + df}$$

Pada Persamaan 10, nilai  $n$  menunjukkan jumlah dari dokumen dan  $df(b_j, D)$  menunjukkan jumlah kemunculan kata  $b_j$  pada sekumpulan dokumen  $D$ . Dalam penggunaan TF-IDF sebagai teknik ekstraksi fitur, sekumpulan kata-kata dapat dimaknai sebagai kata dalam kalimat menurut kaidah bahasa Indonesia dan gabungan dari kata; ataupun potongan dari kata.

#### C. Naïve Bayes Classifier

*Naive Bayes Classifier* merupakan teknik klasifikasi yang berdasarkan Teorema Bayes yang berdasarkan asumsi bahwa jumlah kemunculan suatu fitur terhadap sebuah kelas tidak dipengaruhi oleh fitur-fitur lainnya [5]. Teorema Bayes bekerja dengan mencari nilai *posterior probability* ( $P(A|B)$ ) berdasarkan nilai *prior probability* ( $P(B|A)$ ), *likelihood* ( $P(A)$ ), dan *evidence* ( $P(B)$ ) di mana  $A$  dan  $B$  masing-masing merupakan sebuah kejadian. Proses perhitungan nilai *posterior probability* berdasarkan nilai-nilai yang telah dijabarkan diberikan oleh Persamaan 1.

$$P(A|B) = \frac{P(B|A) \times P(A)}{P(B)}$$

Kemudian, berdasarkan Persamaan 1, saat nilai  $B$  berupa vektor dengan panjang  $n$ , ( $B = (b_1, b_2, \dots, b_n)$ ), yang merepresentasikan sekumpulan fitur, nilai *posterior probability* dapat dihitung berdasarkan Persamaan 2.

$$P(A|b_1, b_2, \dots, b_n) = P(A) \times$$

Dengan memanfaatkan Persamaan 2, proses penentuan sebuah kelas ( $a$ ) di mana  $a \in A$  dari sekumpulan fitur ( $B$ ) dapat dilakukan dengan menggunakan Persamaan 3.

$$C(B, A) = \arg \max_{a \in A} \left[ P(a) \log P(b_i | a) \right]$$

#### D. Multinomial Naïve Bayes

*Multinomial Naïve Bayes* (MNB) merupakan model pengembangan dari metode *Naïve Bayes Classifier* yang teruji baik untuk permasalahan klasifikasi teks [6, 7, 8, 9, 10]. MNB bekerja dengan mengasumsikan bahwa peluang kemunculan kata dalam suatu dokumen bersifat independent satu dengan lainnya; dan panjang dari dokumen yang nantinya akan mempengaruhi jumlah kata yang muncul data sebuah dokumen yang tidak bergantung pada kelasnya [5]. Pada proses klasifikasi teks menggunakan MNB, dilakukan perhitungan terhadap jumlah kemunculan setiap kata dalam setiap dokumen. Proses perhitungan ini dilakukan dengan menggunakan Persamaan 4,

$$P(b_i | a) = \frac{\text{count}(b_i | a)}{\text{count}(a)} \quad (4)$$

di mana:

1.  $\text{count}(b_i | a)$  merupakan jumlah kemunculan kata  $b_i$  dalam kelas  $a$ ;
2.  $\text{count}(a)$  merupakan total jumlah kemunculan setiap kata dalam kelas  $a$ ; dan
3.  $|V|$  menunjukkan jumlah variasi kata dalam seluruh dokumen.

Pada Persamaan 4, penambahan angka satu pada pembilang ditujukan untuk melakukan proses *laplacian smoothing* pada proses perhitungan. Selain menghitung nilai  $P(b_i | a)$  menggunakan Persamaan 4, dibutuhkan juga proses perhitungan terkait nilai probabilitas sebuah dokumen memiliki kelas  $a$ ,  $P(a)$ . Proses perhitungan nilai  $P(a)$  dilakukan dengan menggunakan Persamaan 5.

$$P(a) = \frac{\text{count}(a)}{|V|}$$

di mana  $N_a$  merujuk pada jumlah dokumen dengan kelas  $a$ ,  $N$  merupakan jumlah seluruh dokumen, dan  $N_a \subset N$ .

Dalam implementasinya, untuk menghindari fenomena *floating point underflow* (hasil perhitungan probabilitas setiap kelas sangat kecil sehingga model mengembalikan nilai probabilitas nol untuk setiap kelas), dalam implementasinya seringkali dilakukan modifikasi terhadap Persamaan 3. Modifikasi dilakukan dengan menggantikan operasi perkalian tiap nilai probabilitas dengan operasi penambahan nilai logaritmik dari setiap probabilitas seperti pada Persamaan 6.

$$C(B, A) = \arg \max_{a \in A} \left[ \log P(a) + \sum_{1 < i < n} P(b_i | a) \right]$$

#### E. Complement Naïve Bayes

*Complement Naïve Bayes* (CNB) merupakan pengembangan lainnya dari metode *Naïve Bayes Classifier* dengan memodifikasi persamaan akhir (Persamaan 6) yang digunakan pada metode MNB [18]. Pada model CNB, alih-

alah menghitung probabilitas dari suatu kelas menggunakan probabilitas kemunculan fitur berdasarkan kelas tersebut, CNB menghitung probabilitas dari suatu kelas dengan menggunakan probabilitas kemunculan fitur pada kelas lainnya. Berdasarkan argumentasi tersebut, proses perhitungan pada CNB dirumuskan dalam Persamaan 7.

$$C(B, A) = \arg \max_{a \in A} \left[ \log P(a) + \sum_{1 < i < n} P(b_i | a) \right]$$

di mana nilai  $P(b_i | a')$  dihitung dengan menggunakan Persamaan 8 berikut,

$$P(b_i | a') = \frac{\text{count}(b_i | a')}{\text{count}(a')} \quad (5)$$

dengan:

$$P(b_i | a') = \frac{\text{count}(b_i | a) + 1}{\text{count}(a)} \quad (4)$$

2.  $\text{count}(a')$  merupakan kemunculan seluruh kata pada seluruh kelas kecuali kelas  $a$ ; dan
3.  $|V|$  menunjukkan jumlah variasi kata dalam seluruh dokumen.

Berdasarkan modifikasi yang dilakukan algoritma CNB dipercaya lebih baik dalam melakukan proses klasifikasi berdasarkan dataset dalam keadaan tidak seimbang (*imbalance dataset*) [19].

#### F. F-Score

*F-Score* merupakan salah satu metrik evaluasi yang dapat digunakan untuk mengukur performa dari proses pembelajaran yang telah dilakukan oleh mesin [20]. *F-Score* dihitung berdasarkan nilai *precision* dan *recall* yang dimiliki oleh mesin. *Precision* merupakan nilai prediktif positif yang dihitung berdasarkan jumlah data yang diidentifikasi sebagai kelas positif dibagi dengan total jumlah prediksi positif. Di sisi lain, *recall* menunjukkan nilai berdasarkan jumlah data yang diidentifikasi sebagai kelas positif dibagi dengan total seluruh sampel yang memiliki label positif. Definisi formal dari proses perhitungan nilai *precision* dan *recall* dapat dihitung berdasarkan Persamaan 9 dan 10.

$$\text{Precision} = \frac{\text{True Positive}}{\text{True Positive} + \text{False Positive}}$$

$$\text{Recall} = \frac{\text{True Positive}}{\text{True Positive} + \text{False Negative}}$$

Kemudian, berdasarkan nilai *precision* dan *recall* yang telah dihitung, proses perhitungan nilai *F-Score* dapat dihitung dengan menggunakan Persamaan 11.

$$F_{\text{score}} = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}}$$

### III. METODE PENELITIAN

Dalam penelitian terdapat empat tahap utama yang dilakukan, di antaranya mencakup proses pengumpulan data, *text preprocessing*, proses pelatihan dan pengujian model.

#### A. Pengumpulan Data

Tahap pengumpulan data dilakukan melalui *platform* Twitter dengan melakukan pencarian terhadap konten-konten yang mengandung kata kunci: sedih, marah, depresi dan kata-kata turunan lainnya dari ketiga kata tersebut. Proses pengumpulan data dilakukan dengan memanfaatkan Twitter Developer API. Setelah dataset telah berhasil dikumpulkan, dilakukan tahap normalisasi secara manual terhadap data untuk mengubah slang dalam *dataset* menjadi bahasa formal. Kemudian, *dataset* yang telah dinormalisasikan akan dilabeli oleh seorang pakar psikologi ke dalam dua kelas, yaitu “terindikasi depresi” dan “tidak terindikasi depresi”. Tahap ini menghasilkan sejumlah 10,802 *tweets* dengan 5.617 data berlabel “terindikasi depresi” dan 4.465 “tidak terindikasi depresi”. Dataset yang digunakan dalam penelitian dapat diakses pada *online repository* berikut.

#### B. Text Preprocessing

Berdasarkan data yang telah terkumpul, setiap data akan melalui tahapan-tahapan *text preprocessing* sesuai dengan yang telah dijabarkan pada bab sebelumnya. Kemudian, dataset akan dikelompokkan secara acak ke dalam lima bagian untuk keperluan pelatihan dan pengujian model.

#### C. Proses Pelatihan dan Pengujian Model

Proses pelatihan dan pengujian model akan dilakukan dengan menggunakan strategi K-Fold Cross Validation dengan nilai parameter *k* sama dengan sepuluh. Hal ini mengartikan bahwa akan dataset yang dimiliki akan dibagi ke dalam sepuluh bagian dan proses pelatihan akan dilakukan sebanyak sepuluh kali. Setiap kali proses pelatihan dilakukan, secara bergantian, satu dari sepuluh bagian yang telah dibagi akan diperlakukan sebagai data uji, sementara bagian data lain yang tersisa akan dijadikan data latih. Hal ini dilakukan guna mengurangi bias dari performa aktual yang dimiliki oleh model.

Kemudian, proses parameterisasi terkait dengan *hyperparameter* yang dimiliki oleh TF-IDF juga akan dilakukan pada setiap tahapan validasi. Hal ini dilakukan guna memastikan untuk menghasilkan model dengan parameter terbaik yang tidak bias terhadap proses pembagian data uji dan data latih. Tabel 2 menunjukkan nilai parameter jenis ekstraksi fitur dan konfigurasi potongan/ gabungan (*terms*) dari metode TF-IDF yang diuji coba.

TABEL II

JENIS EKSTRAKSI FITUR DAN KONFIGURASI NILAI PARAMETER TF-IDF DALAM PENELITIAN

Jenis Ekstraksi Fitur	Konfigurasi <i>Terms</i>
Gabungan Kata	Satu kata
	Satu dan dua kata
	Dua kata
	Satu sampai dengan tiga kata
	Dua dan tiga kata
	3 kata
Potongan Kata	Dua dan tiga karakter
	Tiga karakter
	Tiga dan empat karakter
	Tiga sampai lima karakter
	Empat dan lima karakter
	Lima karakter

Selain percobaan terkait dengan jenis ekstraksi fitur dan konfigurasi nilai *terms* yang telah dijelaskan di Tabel 2, akan juga dilakukan percobaan terhadap nilai parameter frekuensi maksimum dan minimum dari *terms* yang digunakan dari metode TF-IDF. Konfigurasi nilai parameter maksimum dan minimum yang diuji coba dalam eksperimen ditunjukkan oleh Tabel 3.

TABEL II  
JENIS EKSTRAKSI FITUR DAN KONFIGURASI NILAI PARAMETER TF-IDF DALAM PENELITIAN

Nama Fitur	Nilai yang dicoba
Frekuensi kemunculan maksimum <i>terms</i>	<i>Terms</i> yang digunakan hanya <i>terms</i> yang tidak muncul lebih dari 90% dokumen yang ada
	<i>Terms</i> yang digunakan hanya <i>terms</i> yang tidak muncul lebih dari 95% dokumen yang ada
	Tidak terdapat batas kemunculan maksimum <i>terms</i>
Frekuensi kemunculan minimum <i>terms</i>	<i>Terms</i> yang digunakan hanya <i>terms</i> dengan frekuensi kemunculan di atas 10
	<i>Terms</i> yang digunakan hanya <i>terms</i> dengan frekuensi kemunculan di atas 5
	Tidak terdapat batas frekuensi minimum

## IV. HASIL DAN PEMBAHASAN

Berdasarkan metode penelitian yang telah dirancang, bab ini akan membahas hasil berdasarkan eksperimen yang telah dilakukan. Hasil dari proses pelatihan dan pengujian gabungan dari metode TF-IDF dan CNB atau MNB dapat dilihat pada Tabel 4.

TABEL IV  
HASIL PERFORMA DAN PARAMETER TERBAIK BERDASARKAN METODE DALAM PENELITIAN

Metode Klasifikasi	Jenis Ekstraksi Fitur	Parameter Terbaik			F1-Score
		Mn	Mx	NG	
MNB	Potongan kata	0	90%	(5,5)	85.36%
	Gabungan kata	0	90%	(1,3)	91.30%
CNB	Potongan kata	0	90%	(5,5)	85.43%
	Gabungan kata	0	90%	(1,3)	91.98%

Dengan keterangan terkait dengan parameter-parameter pada Tabel 4 dapat dilihat pada bagian berikut.

1. Parameter **Mn** menunjukkan nilai frekuensi kemunculan minimum dari *terms* yang digunakan;
2. Parameter **Mx** menunjukkan nilai frekuensi kemunculan maksimum dari *terms* yang digunakan; dan
3. Parameter **NG (x,y)** menunjukkan konfigurasi nilai *terms* target ekstraksi fitur dengan nilai **x** menunjukkan konfigurasi nilai *terms* minimum dan nilai **y** menunjukkan konfigurasi nilai *terms* maksimum. Sebagai contoh, nilai **NG (1,3)** dan jenis ekstraksi potongan kata mengartikan

bawa konfigurasi *terms* yang digunakan terdiri dari *terms* yang dibentuk berdasarkan satu karakter, dua karakter, dan tiga karakter. Saat nilai *x* sama dengan *y* maka nilai *terms* yang digunakan hanya terbentuk dari potongan/ gabungan kata berdasarkan nilai *x*.

Berdasarkan Tabel 4, terlihat bahwa proses penghilangan *terms* berdasarkan jumlah frekuensi minimum (*M<sub>n</sub>*) dapat mempengaruhi performa model klasifikasi baik untuk metode MNB ataupun CNB dan baik dengan metode jenis ekstraksi fitur potongan ataupun gabungan kata. Hal ini dibuktikan dengan F1-Score terbaik untuk setiap konfigurasi dihasilkan saat jumlah frekuensi minimum (*M<sub>n</sub>*) sama dengan nol. Hal ini menunjukkan bahwa pada proses klasifikasi teks dengan panjang berbatas (*tweets*), model klasifikasi berbasis Naïve Bayes tetap dapat memanfaatkan kata-kata dengan jumlah kemunculan yang sedikit sebagai fitur untuk meningkatkan performa klasifikasi.

Kemudian, untuk proses penghilangan *terms* berdasarkan jumlah frekuensi maksimum (*M<sub>x</sub>*), dapat disimpulkan bahwa proses ini mempengaruhi performa dari seluruh konfigurasi model yang ada. Hal ini dibuktikan dengan untuk setiap konfigurasi, F1-score terbaik dihasilkan saat nilai (*M<sub>x</sub>*) sama dengan 90%. Hal ini dikarenakan *terms* yang muncul di lebih dari 90% mungkin saja merepresentasikan kata-kata yang sering muncul (*stopwords*) dan tidak memiliki arti signifikan untuk kelas tertentu.

Di sisi lain, berdasarkan Tabel 4, terlihat bahwa model terbaik dihasilkan oleh algoritma CNB dengan jenis ekstraksi fitur gabungan kata dengan nilai parameter **NG** sebesar **(1,3)**. Model ini berhasil menghasilkan performa F1-Score sebesar 91.98%. Akan tetapi, model CNB ini tidak memiliki perbedaan performa yang signifikan jika dibandingkan dengan model MNB dengan konfigurasi parameter yang sama (perbedaan F1-Score antar model kurang dari 0.58%).

## V. KESIMPULAN

Berdasarkan eksperimen yang telah dilakukan dapat disimpulkan bahwa model CNB dapat digunakan untuk memprediksi konten dengan indikasi depresi dengan tingkat F1-Score sebesar 91.98%. Model terbaik dihasilkan dengan metode ekstraksi fitur gabungan kata dan proses penghilangan kata-kata dengan frekuensi kemunculan di 90% dokumen yang ada. Selain berhasil membuktikan kemampuan model CNB dan MNB dalam memprediksi konten dengan indikasi depresi, penelitian ini juga menghasilkan dataset yang dapat dijadikan ajukan untuk penelitian lainnya.

Berdasarkan penelitian yang telah dilakukan, dapat dilakukan penelitian lebih lanjut terkait dengan proses pembentukan model gabungan *ensemble model* dengan menggunakan mekanisme *bagging* [21, 22] ataupun *stacking* [23] untuk meningkatkan performa klasifikasi. Selain itu, berdasarkan dataset yang dihasilkan dapat diterapkan teknik *topic modeling* [24, 25] untuk mendeteksi kumpulan kata kunci dan pola frasa di dalamnya.

## UCAPAN TERIMA KASIH

Ucapan terima kasih ditujukan kepada Laboratorium Riset bidang ilmu *Artificial Intelligence* dan departemen Lembaga Penelitian dan Pengabdian Masyarakat milik Universitas Multimedia Nusantara atas seluruh dukungan baik dalam hal-

hal yang bersifat administratif, konsultasi ataupun teknikal.

## DAFTAR PUSTAKA

- [1] Badan Pengembangan dan Pembinaan Bahasa, "Kamus Besar Bahasa Indonesia". [Online]. Available: <https://kbki.kemdikbud.go.id/entri/fakta>. [Accessed: 04-Feb-2020].
- [2] Badan Pengembangan dan Pembinaan Bahasa, "Kamus Besar Bahasa Indonesia". [Online]. Available: <https://kbki.kemdikbud.go.id/entri/opini>. [Accessed: 04-Feb-2020].
- [3] B. Clinten, "Pengguna Aktif Harian Twitter Indonesia Diklaim Terbanyak" [Online]. Available: <https://teknokompas.com/read/2019/10/30/16062477/pengguna-aktif-harian-twitter-indonesia-diklaim-terbanyak#:~:text=JAKARTA%2C%20KOMPAS.com%20%2D%20Jumlah,ke%20angka%20145%20juta%20pengguna>. [Accesses: 04-Feb-2020].
- [4] Kementrian Kesehatan, "Laporan Hasil Riset Kesehatan Dasar (Risksdas) Provinsi 2018". Available: <https://www.litbang.kemkes.go.id/laporan-riset-kesehatan-dasar-risksdas/>. [Accessed: 04-Feb-2020].
- [5] C. D. Manning, P. Raghavan, and H. Schütze, "Introduction to Information Retrieval". Cambridge University Press. 2008.
- [6] X. Shuo, "Bayesian Naïve Bayes classifiers to text classification," *Journal of Information Science*, vol. 44, no. 1, pp. 48–59.
- [7] G. Singh, B. Kumar, L. Gaur, and A. Tyagi, "Comparison between Multinomial and Bernoulli Naïve Bayes for Text Classification," in *International Conference on Automation, Computational and Technology Management (ICACTM)*, 2019, pp. 593–596.
- [8] H. J. Kim, J. Kim, J. Kim, and P. Lim, "Towards perfect text classification with Wikipedia-based semantic Naïve Bayes learning," *Neurocomputing*, vol. 315, pp. 128–134.
- [9] X. Shuo, Y. Li, and Z. Wang, "Bayesian multinomial Naïve Bayes classifier to text classification," in Park J., Chen SC., Raymond Choo KK. (eds) *Advanced Multimedia and Ubiquitous Engineering*. Singapore: Springer. 2017.
- [10] L. Ximing, and B. Yang, "A pseudo label based dataless naive bayes algorithm for text classification with seed words," in *Proceedings of the 27th International Conference on Computational Linguistics*, 2018, pp. 1908–1917.
- [11] Z. Qu, X. Song, S. Zheng, X. Wang, X. Song, and Z. Li, "Improved Bayes method based on TF-IDF feature and grade factor feature for chineese information classification" in *IEEE International Conference on Big Data and Smart Computing (BigComp)*, 2018, pp. 677–680.
- [12] A. Aninditya, M. A. Hasibuan, and E. Sutoyo, "Text Mining Approach Using TF-IDF and Naive Bayes for Classification of Exam Questions Based on Cognitive Level of Bloom's Taxonomy" in *2019 IEEE International Conference on Internet of Things and Intelligence System (IoTIS)*, 2019, pp. 112–117.
- [13] D. Kim, D. Seo, S. Cho, and P. Kang, "Multi-co-training for document classification using various document representations: TF-IDF, LDA, and Doc2Vec," *Information Sciences*, vol. 477, pp. 15–29, 2019.
- [14] A. A. Jalal, and B. H. Ali, "Text documents clustering using data mining techniques," *International Journal of Electrical & Computer Engineering*, vol. 11, no. 1, pp. 2088–8708, 2021.
- [15] A. Aizawa. "An information-theoretic perspective of tf–idf measures." *Information Processing & Management*, vol. 39, no. 1, pp. 45–65, 2003.
- [16] A. K. Uysal and S. Gunal, "The impact of preprocessing on text classification," *Information Processing & Management*, vol. 50, no. 1, pp. 104–112, 2014.
- [17] S. Vijayarani, M. J. Ilamathi, and M. Nithya, "Preprocessing techniques for text mining-an overview," *International Journal of Computer Science & Communication Networks*, vol. 5, no. 1, pp. 7–16, 2015.
- [18] J. D. Rennie, L. Shih, J. Teevan, & D. R. Karger, "Tackling the poor assumptions of naive bayes text classifiers," in *International Conference on Machine Learning*, 2003, pp. 616–623.
- [19] A. M. Kibriya, E. Frank, B. Pfahringer, and G. Holmes, "Multinomial naive bayes for text categorization revisited," in *Australasian Joint Conference on Artificial Intelligence*, pp. 488–499, 2004.
- [20] T. Wood, "What is the F-Score" [Online]. Available: <https://deeppai.org/machine-learning-glossary-and-terms/f-score>. [Accessed at: 04-Feb-2020]

- [21] L. Breiman, "Pasting small votes for classification in large databases and on-line", *Machine Learning*, vol. 36 no. 1, pp. 85-103, 1999.
- [22] G. Louppe and P. Geurts, "Ensembles on Random Patches", *Machine Learning and Knowledge Discovery in Databases*, pp. 346 –361, 2012.
- [23] D. H. Wolpert, "Stacked generalization," *Neural networks*, vol. 5 no. 2, pp. 241-259, 1992.
- [24] M. D. Hoffman, D. M. Blei, C. Wang, and J. Paisley, "Stochastic variational inference," *Journal of Machine Learning Research*, vol. 14 no. 5, 2013.
- [25] J. Park and H. J. Oh, "Comparison of topic modeling methods for analyzing research trends of archives management in Korea: focused on LDA and HDP," *Journal of Korean Library and Information Science Society*, vol. 48 no. 4, pp. 235–258, 2017.