JITET (Jurnal Informatika dan Teknik Elektro Terapan)
Vol. 13 No. 3, pISSN: 2303-0577 eISSN: 2830-7062 http://dx.doi.org/10.23960/jitet.v13i3.6905

PENGGUNAAN *RANDOM FOREST* DALAM SISTEM KLASIFIKASI KECEMASAN PADA GENERASI Z

Raina Rahmawati Fitri[1*], Asriyanik[2], Winda Apriandari[3]

[1,2,3] Program Studi Teknik Informatika, Universitas Muhammadiyah Sukabumi; Jl. R. Syamsudin No. 50, Kota Sukabumi, Jawa Barat

Keywords:
Generasi Z, Kecemasan, *Random Forest*, Situs *Web*

Corespondent Email:
rrahmawatif.14@ummi.ac.id

Abstrak. Kesehatan mental merupakan masalah yang semakin berkembang, terutama bagi Generasi Z yang lebih banyak menderita kecemasan karena mereka memiliki tekanan sosial dan akademis yang lebih besar. Di Indonesia, hal ini masih belum dikenal, oleh karen aitu harus ada sistem yang efisien untuk mengidentifikasi kecemasan. Penelitian ini bertujuan untuk mengidentifikasi dan menginvesitasi kecemasan pada Generasi Z dengan membuat sebuah *website* sebagai alat identifikasi. Metode yang dilakukkan adalah ekstraksi data melalui *scraping tweet* dari pengguna aplikasi X dan kemudian dianalisis melalui algoritma *Random Forest*. Hasil *model Random Forest* mampu mendapatkan akurasi 97,71%. Situs *web* yang dikembangkan dapat mengumpulkan informasi keluhan dari pengguna dan memberikan penilaian status kecemasan mereka secara *real-time*. Sebagai tindak lanjut dari hasil analisis, situs *web* ini dapat membantu pengguna untuk menghindari kesalahan diagnosis. Dengan demikian, penelitian ini berkontribusi dalam mengembangkan sistem informasi yang lebih baik untuk mendukung kesehatan mental, serta memberikan rekomendasi yang dapat ditindaklanjuti bagi individu Generasi Z untuk memahami dan mengatasi kecemasan.

Abstract. *Mental health is a growing problem, especially for Generation Z who suffer more from anxiety as they have greater social and academic pressures. In Indonesia, it is still not well known, hence there should be an efficient system to identify anxiety. This research aims to identify and infer anxiety in Generation Z by creating a website as an identification tool. The method used is data extraction through scraping tweets from X application users and then analyzed through the Random Forest algorithm. The results of the Random Forest model were able to get 97.71% accuracy. The developed website can collect complaint information from users and provide real-time assessment of their anxiety status. As a follow-up to the analysis results, this website can help users to avoid misdiagnosis. Thus, this research contributes to developing a better information system to support mental health, as well as providing actionable recommendations for Generation Z individuals to understand and overcome anxiety.*

1. PENDAHULUAN

Generasi Z yang lahir dari tahun 1997 sampai 2012, yang juga dikenal sebagai Gen Z, *iGeneration* atau *Internet Generation*. Gen Z tumbuh di era digitalisasi yang terkakit langsung dengan penggunaan gawai dan media sosial, sehingga mempengaruhi cara mereka berkomunikasi dan menghabiskan kehidupan sehari-hari [1]. Tingkat insensitas penggunaan media sosial pada generasi ini tidak hanya membawa dampak positif, tetapi juga membawa risiko psikologis yang serius, terutama terkait tekanan sosial, kecemasan, dan kesehatan mental. Kesehatan mental

merupakan aspek penting dalam kehidupan, karena menentukan seberapa baik seseorang mengenali kemampuannya, mengatasi stres, dan berkontribusi secara positif terhadap lingkungan di sekitarnya. Ketika potensi ini hilang, individu akan kesulitan untuk menyesuaikan diri dengan tuntutan hidup, sehingga gangguan mental pun muncul [2]. Kondisi ini ditunjukkan oleh survei yang dilakukan melalui aplikasi Jakpat pada 15 November 2022, yang melaporkan bahwa 59,1% Gen Z mengaku mengalami masalah kesehatan mental. Statistik ini menjadikan Gen Z sebagai generasi dengan tingkat gangguan kesehatan jiwa tertinggi dibandingkan Generasi X dan milenial [3]. Hal ini sejalan dengan temuan APA yang mengidentifikasi bahwa hampir 90% Gen Z di Amerika Serikat mengalami gejala stres seperti merasa kewalahan dan khawatir yang berlebihan [4]. Data tersebut menunjukkan bahwa paparan media sosial dan tekanan sosial yang terus-menerus berkontribusi besar terhadap penurunan kesehatan mental di kalangan Gen Z.

Sebagai perpanjangan dari fenomena ini, media sosial tidak hanya menjadi pendorong, tapi juga saluran utama bagi Gen Z untuk mengekspresikan kecemasan mereka. Jika melihat hasil survei mengenai kecemasan Gen Z di Amerika Serikat, keluhan yang sama juga dapat ditemukan pada Gen Z di Indonesia yang lebih memilih untuk mengekspresikan perasaan mereka melalui aplikasi X. Platform ini banyak digunakan untuk berinteraksi, mengekspresikan diri, dan hiburan bagi Gen Z, terutama sebagai cara untuk memantau tren dan membangun jejaring sosial [5]. Dalam konteks ini, aplikasi X dapat digunakan sebagai sumber data yang sesuai untuk menganalisis gejala kecemasan dengan cara mengklasifikasikan cuitan sebagai merasa atau tidak merasa kecemasan. Hal ini juga didukung oleh penelitian Rahayu dkk. [6] yang menunjukkan bagaimana individu yang memiliki kecenderungan depresi dan kecemasan lebih cenderung beralih ke media sosial seperti Twitter/X untuk mendapatkan dukungan emosional daripada berinteraksi langsung dengan lingkungan sekitar. Dengan demikian, media sosial tidak hanya menjadi indikator, tetapi juga menifestasi dari kondisi psikologis penggunanya, terutama Gen Z.

Melanjutkan urgensi tersebut, penelitian ini berusaha mengembangkan sistem klasifikasi kecemasan untuk Gen Z melalui algoritma *Random Forest*. Penelitian ini menggunakan *dataset* yang berasal dari aplikasi media sosial X karena aplikasi ini secara jelas merepresentasikan ekspresi dan opini Gen Z mengenai kecemasan. Hasil dari penelitian ini adalah sebuah aplikasi *web* yang mampu memprediksi apakah seseorang mengalami kecemasan sebagai fungsi dari cuitan yang dimasukkan. Penelitian ini menjadi penting karena masih terbatasnya penelitian yang membangun sistem klasifikasi kecemasan berbasis *machine learning* secara *real-time* di Indonesia.

2. TINJAUAN PUSTAKA

2.1. Kecemasan

Kecemasan adalah perasaan tidak nyaman atau takut yang samar-samar disertai dengan respon otonom dan biasanya sesuatu yang tidak spesifik atau sesuatu yang tidak disadari. Kecemasan adalah hal normal, jika disadari dan dapat memicu perilaku adatif seseorang untuk mengkondisikan dirinya agar mampu menghadapi apa yang ditakutkannya. Namun, kecemasan menjadi sebuah hal yang tidak normal, jika direspon secara berlebihan. Hal ini dapat menyebabkan ketidaknyamanan, mengganggu kegiatan sehari-hari, distres, atau menghindari interaksi sosial yang penuh tekanan bagi individu [7].

2.2. Metode SEMMA

a. *Sample*
   Pengambilan *sample* merupakan langkah awal dalam proses pengumpulan data dimana sebagian data dikumpulkan dari populasi yang besar untuk dijadikan representasi dari keseluruhan informasi yang dibutuhkan [8].
b. *Explore*
   Tahap eksplorasi data juga bertanggung jawab untuk memahami struktur dan kualitas data serta mengidentifikasi duplikasi atau kesalahan dalam data. Proses ini mencakup pemilihan data yang representatif dengan menggunakan metode pengambilan *sample linier* sehingga masih mewakili seluruh populasi [8].

c. *Modify*
Modifikasi dilakukan untuk mengubah data yang tidak terstuktur menjadi data yang terstruktur. Pembersihan data, kapitalisasi, tokenisasi, penghilang *stopword*, dan penyaringan dilakukan di sini untuk membersihkan data dan membuatnya siap untuk tahap analisis selanjutnya [9].
d. *Modeling*
Model dikembangkan sebagai replika yang tepat dari suatu proses atau sistem, yang dapat bertindak sebagai dasar untuk keputusan atau tindakan oleh organisasi atau orang [10].
e. *Assess*
Evaluasi untuk menguji keakuratan dan kinerja model. Langkah ini diperlukan untuk memahami area perbaikan dan memastikan model yang dibangun dapat diandalkan [10].
f. *Deployment*
Deployment merupakan tahap dimana model dijadikan sebagai bagian dari sistem operasi atau aplikasi yang digunakan, sehingga dapat membantu dalam otomatisasi dan penerapan model pada situasi dunia nyata [11].

2.3. TF-IDF

Term Frequency – Inverse Document Frequency atau biasa disebut TF-IDF adalah metode pembobotan kata dengan menghitung frekuensi kata yang ada dalam sebuah dokumen. TF-IDF memberikan bobot pada kata di setiap dokumen tergantung pada frekuensi atau beberapa kali kata tersebut muncul di dalam sebuah dokumen [12]. Berikut rumus untuk menghitung TF-IDF:

a. *Term Frequency* (TF) adalah frekuensi kemunculan fitur t dalam dokumen, jika fitur sering muncul maka nilai TF semakin besar.
$$TF_t = (t, d) \quad (1)$$
b. IDF adalah logaritma dari total dokumen n dibagi jumlah dokumen df yang mengandung fitur t, jika fitur muncul di sedikit dokumen maka nilai IDF semakin besar.
$$IDF_t = log\frac{n}{df(t)} \quad (2)$$
c. TF-IDF adalah hasil kali TF dan IDF, jika nilainnya tinggi maka fitur memiliki bobot yang berat, sebaliknya jika nilainya rendah maka bobotnya ringan.
$$W_t = TF_t \times IDF_t \quad (3)$$

2.4. SMOTE

Synthetic Minority Oversampling Technique atau biasa disebut SMOTE merupakan metode *oversampling* yang paling banyak digunakan untuk menangani masalah distribusi kelas yang tidak seimbang dalam pemodelan *machine learning.* SMOTE mencoba menyeimbangkan distribusi kelas dengan melakukan *oversampling* pada kelas minoritas melalui pembangkitan data sintetis secara acak [13].

2.5. *Random Forest*

Random Forest adalah metode untuk meningkatkan akurasi hasil melalui pembangkitan fitur secara acak untuk setiap node. RF terdiri dari beberapa pohon keputusan yang digunakan untuk mengklasifikasikan data [14]. Berikut rumus yang dikembangkan oleh Claude Shannon dalam teori informasi untuk menghitung *entropy* dan *information gain* [15].
$$Entropi(A) = -\sum_{i=1}^{k} -p_i\, log_2(p_i) \quad (4)$$
$$IG = Entropi(A) - \sum_{i=1}^{k} \frac{|A_{i|}}{|A|} \times Entropi(A_i) \quad (5)$$
Selain entropi, ukuran lain yang digunakan ketika mendistribusikan data adalah indeks Gini. Nilai Gini diperhitungkan pada setiap langkah distribusi dalam menghitung pengurangan nilai Gini. Nilai Gini dihitung melalui rumus berikut [15].
$$Gini(A) = 1 - \sum_{i=1}^{k} (p_i^2) \quad (6)$$
$$GD = Gini(A) - \sum_{i=1}^{k} \frac{|A_i|}{|A|} \times Gini(A_i) \quad (7)$$

2.6. *Confusion Matrix*

Confusion matrix adalah tabel yang banyak digunakan dalam menjelaskan kinerja model *machine learning* [16]. Tabel ini menggambarkan prediksi atau status aktual dari data yang dihasilkan oleh algoritma *machine learning*, khususnya model klasifikasi. Alat ini juga digunakan sebagai pengukuran akurasi, berupa matriks berukuran 2×2 yang dapat mengestimasi seberapa tepat algoritma yang digunakan [17]. Rumus di bawah digunakan dalam upaya memperkirakan akurasi, presisi, sensitivitas, dan F1-*score* dari hasil *data mining*.

a. Akurasi adalah proporsi prediksi yang benar dan mencerminkan seberapa baik klasifikasi memprediksi keadaan.

$$Akurasi = \frac{(TP + TN)}{(TP + TN + FP + FN)} \quad (8)$$

b. Presisi adalah ukuran dari contoh yang benar dikembalikan, yaitu prediksi positif yang benar untuk semua contoh positif yang benar.

$$Presisi = \frac{TP}{TP + FP} \quad (9)$$

c. Sensitivitas mengacu pada kemampuan model untuk mengenali contoh positif, dihitung dari pembagian prediksi positif yang benar dengan kumpulan prediksi positif yang benar.

$$Sensitivitas = \frac{TP}{TP + FN} \quad (10)$$

d. F1-*score* adalah ukuran rata-rata yang seimbang antara akurasi dan sensitivitas, nilai yang mendekati 1 mengindikasi bahwa model berkinerka baik.

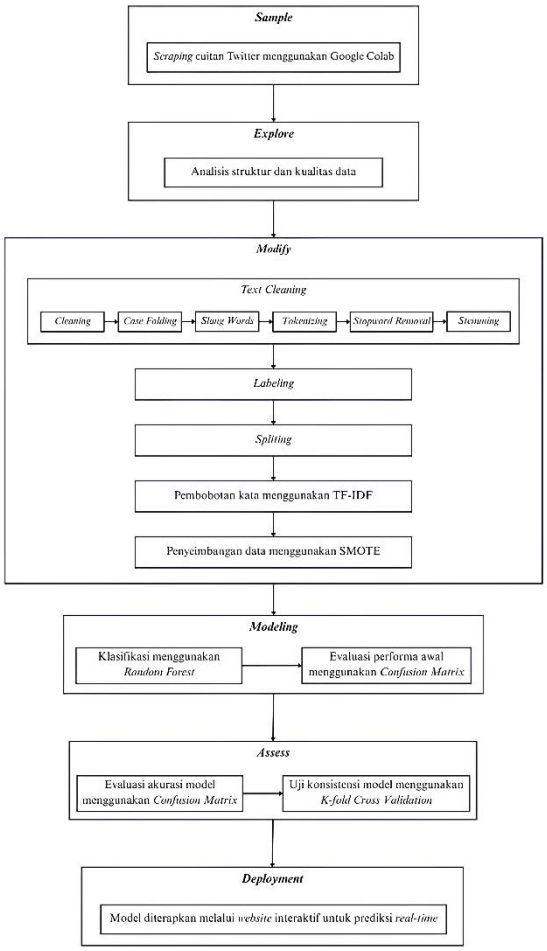
$$F1score = 2 \times \frac{presisi \times sensitivitas}{presisi + sensitivitas} \quad (11)$$

2.7. *K-Fold Cross Validation*

Cross validation merupakan cara untuk memprediksi akurasi model *machine learning* dengan tujuan memperoleh validasi yang mendekati kenyataan melalui perputaran data latih dan data uji. Teknik ini menggunakan *K-fold cross validation* untuk menentukan rata-rata pencapaian sistem dengan memasukkan atribut *input* ke dalam iterasi acak [18].

3. METODE PENELITIAN

Metode penelitian ini dimaksudkan untuk memberikan proses yang sistematis dalam pengumpulan data, persiapan alat, dan pengolahan data secara bertahap sesuai tahapan. Penelitian ini menggunakan metode SEMMA (*Sample, Explore, Modify, Model, Assess*) yang diformulasikan agar sesuai dengan analisis data yang sistematis. Dengan metode ini, penulis dapat memperoleh informasi yang penting dari data melalui pengambilan sampel, mengeksplorasi, memodifikasi, memodelkan, dan mengevaluasi.



Gambar 1. Alur penelitian

Penjelasan dari gambar 1:

a. Pada tahap *sample*, data dikumpulkan dengan melakukan *scraping tweet* menggunakan Google Colab.
b. Tahap *explore*, dilakukan untuk memeriksa struktur dan kualitas data untuk mencari *noise* dalam data.
c. Tahap *modify* meliputi tahapan *text preprocessing,* seperti *text cleaning, case folding,* normalisasi kata tidak baku, tokenisasi, *stopword removal*, dan *stemming*. Kemudian data diberi label, dibagi menjadi data latih dan data uji, dilakukan pembobotan kata menggunakan TF-IDF, dan diseimbangkan menggunakan teknik SMOTE.
d. Tahap *model*, model klasifikasi dikembangkan dengan menggunakan algoritma *Random Forest*. Evaluasi awal dilakukan dengan menggunakan *confusion matrix*.

e. Tahap *assess*, akurasi dan konsistensi model dievaluasi dengan menggunakan *confusion matrix* dan *K-fold cross validation*.
f. Tahap *deployment*, model diterapkan pada situs *web* interaktif untuk prediksi secara *real-time*.

4. HASIL DAN PEMBAHASAN

4.1. *Sample* (Pengumpulan Data)

Pengumpulan data dilakukan dengan melakukan *scraping* terhadap *tweet* pada platform media sosial X dengan menggunakan kata kunci yang berhubungan dengan kecemasan pada generasi Z seperti "*anxiety gen z*", "cemas", "gelisah", "*burnout* kuliah", dan lain-lain. *Scraping* dilakukan dengan bantuan *tweet-harvest* di *Google Colaboratory* menggunakan Node.js. Data yang terkumpul disimpan dalam format CSV untuk dianalisis lebih lanjut.

4.2. *Explore* (Deskripsi Data)

Pada tahap ekslorasi data, penulis meneliti distribusi data berdasarkan kata kunci yang digunakan pada saat melakukan *scraping*. Setiap kata kunci menghasilkan bermacam-macam *tweet* sesuai dengan seberapa sering kata kunci tersebut terlihat di platform X. *Scraping* selama beberapa hari menghasilkan data sebanyak 16.314 data dengan 15 kolom. Di antara eksplorasi awal, ada 460 duplikasi yang perlu dihilangkan. Selain itu, ada beberapa kolom dengan nilai kosong yang harus diisi pada tahap *preprocessing* data selanjutnya. Hasil dari pengecekan jumlah data kosong perkolom adalah kolom image_ url memiliki 14.839 nilai kosong, in_reply_to_screen_name memiliki 6.559, location memiliki 8.833, dan full_text memiliki 4 nilai kosong. Semua kolom yang tersisa terisi dengan baik tanpa ada data yang hilang. Selain itu, dari hasil fungsi info(), terlihat bahwa semua kolom memiliki tipe data yang sesuai dan sebagian besar kolom memiliki data yang lengkap, sehingga memberikan gambaran tentang struktur dan karakteristik kumpulan data sebelum pembersihan dan analisis lebih lanjut.

4.3. *Modify* (Modifikasi Data)

Tahap *modify* memberikan data sebelum melakukan pemodelan dengan berbagai pra-pemprosesasn teks dan pemrosesan lanjutan. Proses-proses dalam tahap ini adalah adalah pra-pemrosesasn teks, pelabelan, pemisahan, rekayasa fitur, dan menyeimbangkan kelas dengan SMOTE. Setiap proses dijelaskan secara rinci dalam subbagian berikut.

a. *Text Preprocessing*

Pra-pemrosesan teks dilakukan untuk membersihkan dan memproses data teks sebelum pemodelan. Proses ini sangat penting dalam menghilangkan *noise*, menormalkan struktur bahasa, dan meningkatkan efesiensi dan akurasi dalam analisis lebih lanjut. Langkah pertama dari *preprocessing* adalah pembersihan, yang menghilangkan item yang tidak relevan seperti simbol, angka, tanda baca, emoji, URL, dan spasi. Tujuan dari proses ini adalah untuk menghilangkan gangguan dari data teks sehingga hasil analisis tidak bias dengan karakter yang tidak memiliki nilai semantik.

Tabel 1. Hasil proses *cleaning*

Dokumen	*Input Process*	*Output Proces*
D1	gais cemas akan skripsi wajar gasii??	gais cemas akan skripsi wajar gasii
D2	Beratnya tidak seberapa tetapi rasa cemas yang terpendam dalam dada membuatku merasa terbebani	Beratnya tidak seberapa tetapi rasa cemas yang terpendam dalam dada membuatku merasa terbebani
D3	Asli suka banget! Udah lama deh gue gak ngerasa se-excited ini	Asli suka banget Udah lama deh gue gak ngerasa seexcited ini

Setelah pembersihan data, ada juga *case folding*, yang berarti semua huruf diubah menjadi huruf kecil. Hal ini dilakukan untuk menstandarisasi representasi kata sehingga "Cemas" dan "cemas" tidak dianggap sebagai dua hal yang berbeda.

Tabel 2. Hasil proses *case folding*

Dokumen	*Input Process*	*Output Proces*
D1	gais cemas akan skripsi wajar gasii	gais cemas akan skripsi wajar gasii
D2	Beratnya tidak seberapa tetapi rasa cemas yang terpendam dalam dada membuatku merasa terbebani	beratnya tidak seberapa tetapi rasa cemas yang terpendam dalam dada membuatku merasa terbebani
D3	Asli suka banget Udah lama deh gue gak ngerasa seexcited ini	asli suka banget udah lama deh gue gak ngerasa seexcited ini

Kemudian kata-kata yang tidak baku atau bahasa gaul dinormalisasi. Kata-kata gaul, singkatan, atau ejaan informal diganti

dengan kata formal menggunakan kamus padanan kata yang telah disusun sebelumnya. Misalnya, "bgt" diganti dengan "banget" dan "ovt" dengan "*overthinking*". Hal ini membuat teks menjadi lebih seragam dan bermakna.

Tabel 3. Hasil proses *slang word*

Dokumen	*Input Process*	*Output Proces*
D1	gais cemas akan skripsi wajar gasii	gais cemas akan skripsi wajar gasii
D2	beratnya tidak seberapa tetapi rasa cemas yang terpendam dalam dada membuatku merasa terbebani	beratnya tidak seberapa tetapi rasa cemas yang terpendam dalam dada membuatku merasa terbebani
D3	asli suka banget udah lama deh gue gak ngerasa seexcited ini	asli suka banget sudah lama deh saya tidak merasa seexcited ini

Setelah normalisasi kata-kata, *tokenizing* dilakukan, yang membagi teks menjadi unit kata atau token. Analisis kemudian dapat dilakukan pada tingkat kata dengan ini dan merupakan dasar untuk tahap NLP berikutnya seperti penghitungan frekuensi dan ekstraksi fitur.

Tabel 4. Hasil proses *tokenizing*

Dokumen	*Input Process*	*Output Proces*
D1	gais cemas akan skripsi wajar gasii	['gais', 'cemas', 'akan', 'skripsi', 'wajar', 'gasii']
D2	beratnya tidak seberapa tetapi rasa cemas yang terpendam dalam dada membuatku merasa terbebani	['beratnya', 'tidak', 'seberapa', 'tetapi', 'rasa', 'cemas', 'yang', 'terpendam', 'dalam', 'dada', 'membuatku', 'merasa', 'terbebani']
D3	asli suka banget sudah lama deh saya tidak merasa seexcited ini	['asli', 'suka', 'banget', 'sudah', 'lama', 'deh', 'saya', 'tidak', 'merasa', 'seexcited', 'ini']

Tahap berikutnya adalah menghilangkan *stopwords*, yaitu kata-kata yang sering muncul namun tidak memiliki bobot informasi yang berarti, seperti "yang", "dan", "itu", dll. Menghilangkan kata-kata ini akan membuat perhatian analisis terpusat pada kata-kata yang lebih bermakna.

Tabel 5. Hasil proses *stopword removal*

Dokumen	*Input Process*	*Output Proces*
D1	['gais', 'cemas', 'akan', 'skripsi', 'wajar', 'gasii']	['gais', 'cemas', 'skripsi', 'wajar', 'gasii']
D2	['beratnya', 'tidak', 'seberapa', 'tetapi', 'rasa', 'cemas', 'yang', 'terpendam', 'dalam', 'dada', 'membuatku', 'merasa', 'terbebani']	['beratnya', 'cemas', 'terpendam', 'dada', 'membuatku', 'terbebani']
D3	['asli', 'suka', 'banget', 'sudah', 'lama', 'deh', 'saya', 'tidak', 'merasa', 'seexcited', 'ini']	['asli', 'suka', 'deh', 'seexcited']

Langkah *preprocessing* terakhir adalah *stemming*, di mana sebuah kata dikembalikan ke bentuk dasarnya. Sebagai contoh, kata "dikerjakan" diubah menjadi "kerja". *Stemming* mengurangi variabilitas kata dan menyederhanakan kompleksitas kosakata dalam teks melalui penggunaan algoritma seperti Sastrawi.

Tabel 6. Hasil proses *stemming*

Dokumen	*Input Process*	*Output Proces*
D1	['gais', 'cemas', 'skripsi', 'wajar', 'gasii']	['gais', 'cemas', 'skripsi', 'wajar', 'gasii']
D2	['beratnya', 'cemas', 'terpendam', 'dada', 'membuatku', 'terbebani']	['berat', 'cemas', 'pendam', 'dada', 'buat', 'beban']
D3	['asli', 'suka', 'deh', 'seexcited']	['asli', 'suka', 'deh', 'seexcited']

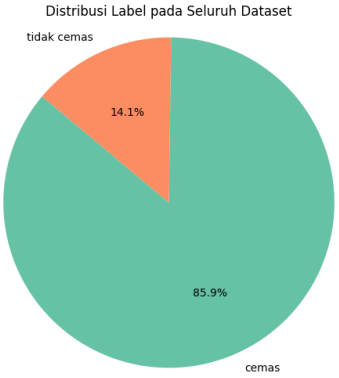
b. *Labeling*

Setelah menyelesaikan pembersihan data, pelabelan data dilakukan dengan menerapkan pendekatan *rule-based* berbasis kata kunci. Semua *tweet* diberi label berdasarkan konteks, dengan mencocokkan kata-kata dalam teks dengan daftar kata kunci yang telah dibuat sebelumnya, termasuk cemas, tidak cemas, dan tidak diketahui. Proses ini dilakukan secara otomatis tanpa melibatkan anotator manual. Distribusi hasil pelabelan menunjukkan dominasi label cemas dengan jumlah data yang cukup besar dibandingkan dengan dua label lainnya. Informasi distribusi data per label dapat dilihat pada Tabel 7.

Tabel 7. Jumlah data setiap label

No.	Label	Jumlah Data
1.	Cemas	12169
2.	Tidak cemas	1998
3.	Tidak diketahui	1555

Terdapat data yang ditandai sebagai tidak diketahui karena tidak adanya kecocokan antara isi *tweet* dengan kata kunci yang diberikan. Karena label ini tidak memberikan kontribusi yang informatif terhadap pelatihan model, maka label ini dihapus pada tahap pra-pemodelan. Gambar 3 menunjukkan distribusi visual dari label-label ini.



Gambar 2. Diagram distribusi label

c. *Spliting*

Setelah *preprocessing* dan pelabelan dilakukan, data dibagi menjadi dua set, yaitu data pelatihan dan data pengujian. Rasio 80:20 digunakan untuk pemisahan, dengan harapan sebagian besar data digunakan untuk melatih model klasifikasi dan sisanya digunakan untuk penilaian kinerja model pada data yang tidak terlihat. Pemisahan dilakukan secara bertingkat, dengan menjaga proporsi distribusi label tetap sama untuk kedua himpunan bagian. Hal ini disebabkan oleh kebutuhan untuk menjaga agarmodel tidak bias terhadap kelas mayoritas saat pelatihan dan pengujian.

Tabel 8. Hasil proses *spliting*

No	Tipe Data	Jumlah
1.	Data latih	11.333
2.	Data Uji	2.834

d. TF-IDF

Langkah selanjutnya adalah pembobotan kata menggunakan metode TF-IDF. Frekuensi kemunculan kata dalam dokumen tertentu diukur dengan TF, sedangkan signifikansi sebuah kata relatif terhadap semua dokumen dalam korpus diukur dengan IDF. Skor TF-IDF dihitung dengan membagi jumlah total dokumen dengan jumlah dokumen dengan istilah tersebut dan mengirimkan skor TF dari setiap istilah untuk mendapatkan skor TF-IDF. Hasil akhir dari pembobotan tersebut menghasilkan representasi vektor dari setiap kata terhadap dokumen. Vektor tersebut akan digunakan sebagai fitur *input* dalam proses pemodelan. Nilai-nilai hasil TF dan IDF disajikan pada tabel 9, sedangkan hasil akhir pembobotan TF-IDF ditampilkan pada tabel 10 di bawah ini.

Tabel 9. Hasil perhitungan TF dan IDF

Term	TF1	TF2	TF3	DF	IDF
gais	1	0	0	1	0,477
cemas	1	1	0	2	0,176
skripsi	1	0	0	1	0,477
wajar	1	0	0	1	0,477
gasii	1	0	0	1	0,477
berat	0	1	0	1	0,477
pendam	0	1	0	1	0,477
dada	0	1	0	1	0,477
buat	0	1	0	1	0,477
beban	0	1	0	1	0,477
asli	0	0	1	1	0,477
suka	0	0	1	1	0,477
deh	0	0	1	1	0,477
seexcited	0	0	1	1	0,477

Tabel 10. Hasil perhitungan TF-IDF

Term	TF-IDF1	TF-IDF2	TF-IDF3
gais	0,477	0	0
cemas	0,176	0,176	0
skripsi	0,477	0	0
wajar	0,477	0	0
gasii	0,477	0	0
berat	0	0,477	0
pendam	0	0,477	0
dada	0	0,477	0
buat	0	0,477	0
beban	0	0,477	0
asli	0	0	0,477
suka	0	0	0,477
deh	0	0	0,477
seexcited	0	0	0,477

e. SMOTE

Sebelum memasuki tahap pemodelan, penyeimbangan data, SMOTE dilakukan di setiap kelas agar distribusi seimbang di antarakelas-kelas pada data latih. Data sintetis dari kelas minoritas dibuat dan karakteristik data yang serupa digabungkan bersama sebagai cara untuk membuat distribusi yang adil. Sebelum penyeimbangan SMOTE, kelas Tidak Cemas diambil alih oleh kelas Cemas, sehingga kemungkinan menyebabkan bias

dalam model. Setelah penerapan SMOTE, kedua kelas memiliki total data yang sama. Hasil distribusi data proses SMOTE disajikan pada Tabel 11.

Tabel 11. Hasil proses SMOTE

Sebelum SMOTE		Sesudah SMOTE	
Cemas	Tidak Cemas	Cemas	Tidak Cemas
9.735	1.598	9.735	9.735

4.4. *Model* (Pemodelan Data)

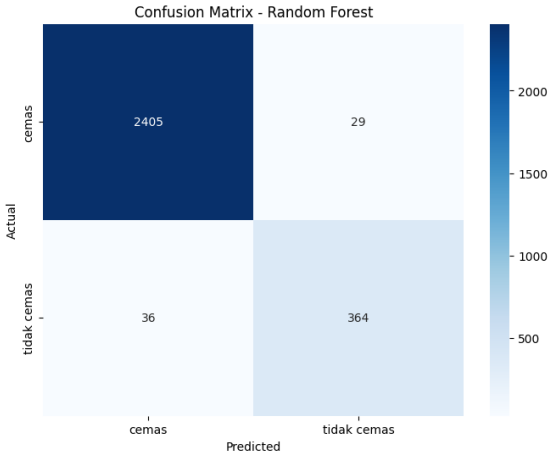
Tugas pemodelan dilakukan setelah *preprocessing*, pelabelan, pemisahan, pembobotan, dan penyeimbangan data. Model dilatih pada *data training* dan diuji pada *data testing*. Pengujian dilakukan dengan mengukur akurasi, presisi, *recall*, dan F1-*score* untuk memastikan performa model dalam mengklasifikasikan teks, sebagai teks yang cemas atau tidak cemas. Model melaporkan kinerja yang tinggi dengan akurasi 97,71%. Rincian metrik evaluasi ditunjukkan pada Tabel 12.

Tabel 12. Hasil evaluasi model

Kelas	*Precision*	*Recall*	F1-*score*
Cemas	0.99	0.99	0.99
Tidak Cemas	0.93	0.91	0.92
Accuracy			0.98

4.5. *Assess* (Evaluasi Data)

Setelah mengembangkan dan mengevaluasi model, tahap evaluasi dilakukan untuk memastikan keandalan dan konsistensi kinerja model. Evaluasi kinerja dilakukan dengan *confusion matrix* dan 10-*fold cross validation*. *Confusion matrix* digunakan untuk mengidentifikasi klasifikasi yang benar dan salah dari data uji, dan menganalisis tingkat kesalahan klasifikasi antar kelas. Hasil penelitian menunjukkan bahwa dari 2.834 data yang diuji, 2.405 data cemas dan 364 data tidak cemas diklasifikasikan dengan benar. Sementara itu, 36 data tidak cemas salah diklasifikasikan sebagai cemas dan 29 data cemas salah diklasifikasikan sebagai tidak cemas. Matriks di bawah menunjukkan akurasi kinerja model yang tepat sebesar 97,71%, dengan nilai *precision*, *recall*, dan F1-*score* yang seimbang.



Gambar 3. Hasil *confusion matrix*

Selain itu, untuk menguji ketahanan kinerja model terhadap variasi data pelatihan, dilakukan validasi silang sebanyak 10 kali. Hasilnya menunjukkan kinerja yang stabil dengan rata-rata akurasi 0.9578, presisi 0.9568, *recall* 0.9578, dan F1-*score* 0.9568. Nilai-nilai di bawah menunjukkan bahwa model *Random Forest* memiliki kemampuan generalisasi yang tinggi untuk data baru dan stabil di seluruh bagian data.

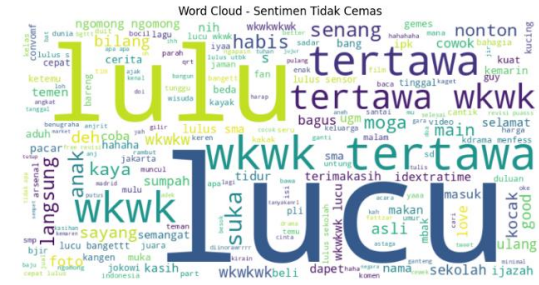
Tabel 13. Hasil 10-*fold cross validation*

Iterasi	Akurasi	Presisi	*Recall*	F1-Skor
1	95,68%	95,59%	95,68%	95,53%
2	95,68%	95,68%	95,68%	95,55%
3	97,00%	96,94%	97,00%	96,94%
4	95,41%	95,32%	95,41%	95,30%
5	95,59%	95,50%	95,59%	95,76%
6	95,85%	95,75%	95,85%	95,73%
7	95,23%	95,06%	95,23%	95,07%
8	95,59%	95,43%	95,59%	95,44%
9	96,03%	95,95%	96,03%	95,96%
10	95,50%	95,44%	95,50%	95,47%
Rata-Rata	95,78%	95,68%	95,78%	95,68%

4.6. Visualisasi Sentimen

Visualisasi dilakukan dengan menggunakan *WordCloud* untuk menampilkan kata-kata yang paling sering muncul untuk setiap label sentimen, yaitu cemas dan tidak cemas. Tujuannya adalah untuk memudahkan interpretasi data dan mengidentifikasi pola kata yang dominan.

Gambar 4. *Wordcloud* cemas

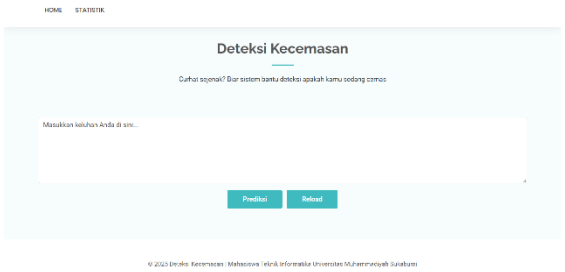


Gambar 5. *Wordcloud* tidak cemas

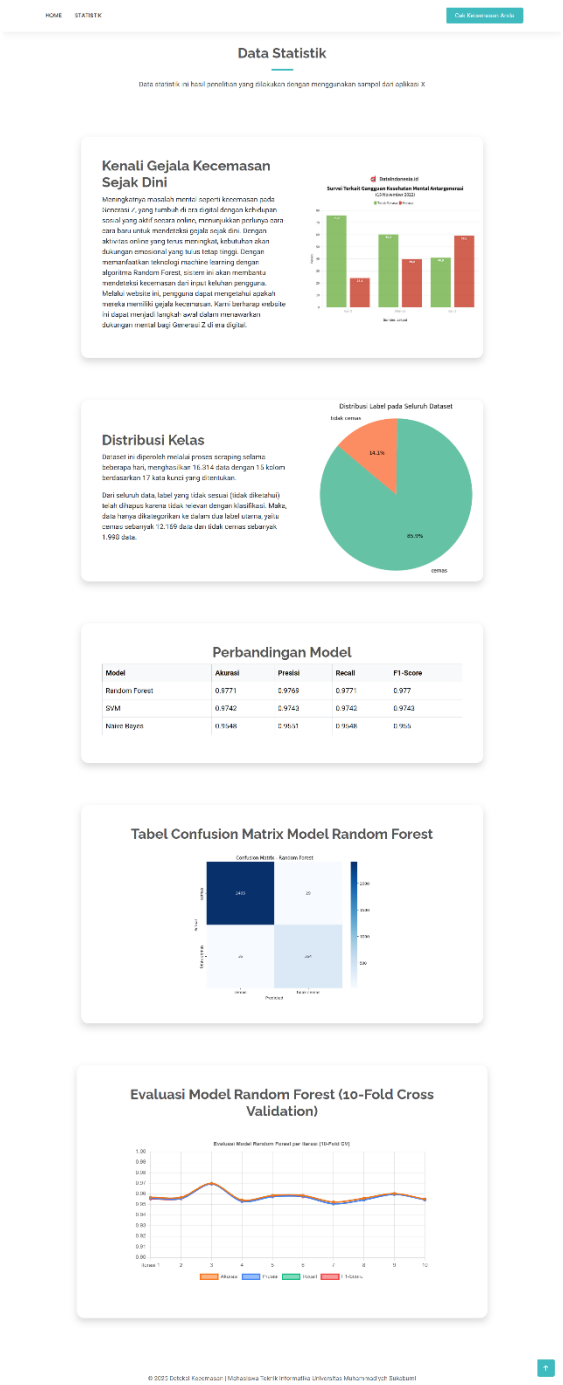
4.7. *Deployment* (Penerapan)

Tahap implementasi bertujuan untuk merealisasikan sistem pendeteksi kecemasan berbasis *web* melalui model *Random Forest*. Sistem ini memiliki antarmuka yang mudah dan responsif sehingga memudahkan pengguna untuk menggunakannya dalam melakukan deteksi secara *real-time* serta mengakses informasi statistik hasil penelitian.

Gambar 6. Tampilan halaman utama



Gambar 7. Tampilan halaman deteksi kecemasan



Gambar 8. Tampilan halaman data statistik

5. KESIMPULAN

Berdasarkan penelitian yang sudah dilakukan berikut beberapa kesimpulan yang dapat diambil adalah sebagai berikut:

a. Penelitian ini mampu mengambil 16.314 informasi *tweet* dari pengguna aplikasi X mengenai kecemasan pada Generasi Z.
b. Data yang diperoleh ditransformasi melalui tahap *preprocessing* dan penyeimbangan menggunakan metode

SMOTE agar siap untuk digunakan dalam aplikasi *model training*.

c. *Model Random Forest* berhasil digunakan untuk klasifikasi kecemasan dengan hasil akurasi yang tinggi, yaitu sebesar 97.91%.
d. Validasi model juga dilakukan dengan menggunakan 10-*fold Cross Validation*, yang menunjukkan performa model yang kuat dan stabil.
e. Sistem untuk mendeteksi kecemasan kemudian ditempatkan dalam sebuah situs *web*, yang memungkinkan deteksi kecemasan secara *real-time* dengan menggunakan *input* keluhan pengguna.
f. Kelebihan dari sistem ini adalah akurasi yang tinggi, aplikasi yang sederhana, dan akses yang mudah bagi pengguna Generasi Z untuk mendeteksi gejala kecemasan sendiri.
g. Kekurangannya adalah sumber data yang terbatas dari satu platform dan perlakuan berbasis teks tanpa memperhatikan aspek lain seperti konteks sosial atau emosional.
h. Penelitian selanjutnya dapat menambahkan fungsionalitas pembelajaran, menghubungkan ke layanan profesional, memperluas sumber data, dan bereksperimen dengan model lain untuk meningkatkan akurasi dan cakupan deteksi.

UCAPAN TERIMA KASIH

Penulis mengucapkan terima kasih kepada seluruh pihak yang telah memberikan dukungan dalam penyelesaian penelitian ini, khususnya kepada dosen pembimbing, pimpinan fakultas dan program studi, keluarga tercinta, teman-teman seperjuangan, serta semua sahabat yang telah memberikan semangat, bantuan, dan doa selama proses penelitian berlangsung.

DAFTAR PUSTAKA

[1] C. Nurlaila, Q. Aini, S. Setyawati, dan A. Laksana, "Dinamika Perilaku Gen Z Sebagai Generasi Internet," *Konsensus : Jurnal Ilmu Pertahanan, Hukum dan Ilmu Komunikasi*, vol. 1, no. 6, hlm. 95–102, Des 2024, doi: 10.62383/konsensus.v1i6.433.

[2] E. E. Pratiwi, A. R. Aisy, R. Rahmaddeni, dan N. Ananta, "KLASIFIKASI KESEHATAN MENTAL PADA USIA REMAJA MENGGUNAKAN METODE SVM," *Jurnal Informatika dan Teknik Elektro Terapan*, vol. 13, no. 2, hlm. 445–453, Apr 2025, doi: 10.23960/jitet.v13i2.6232.

[3] M. A. Rizaty, "Survei: Gen Z Paling Banyak Merasakan Masalah Kesehatan Mental," DataIndonesia.id.

[4] E. Narus, "Gen Z Rentan Terkena Gangguan Mental, Apa Penyebabnya?," Media Indonesia. Diakses: 15 Desember 2024. [Daring]. Tersedia pada: https://mediaindonesia.com/humaniora/707791/gen-z-rentan-terkena-gangguan-mental-apa-penyebabnya#google_vignette

[5] A. P. Dewi dan S. Delliana, "SELF DISCLOSURE GENERASI Z DI TWITTER," *Ekspresi dan Presepsi : Jurnal Ilmu Komunikasi*, hlm. 62–69, 2020, [Daring]. Tersedia pada: http://ejournal.upnvj.ac.id/index.php/JEP/index

[6] K. Rahayu, V. Fitria, D. Septhya, Rahmaddeni, dan L. Efrizoni, "Klasifikasi Teks untuk Mendeteksi Depresi dan Kecemasan pada Pengguna Twitter Berbasis Machine Learning," *MALCOM: Indonesian Journal of Machine Learning and Computer Science*, vol. 3, no. 2, hlm. 108–114, Sep 2023, doi: 10.57152/malcom.v3i2.780.

[7] K. I. P. Sari, S. Muthoharoh, dan R. Widiyawati, "KECEMASAN AKADEMIK MAHASISWA KEBIDANAN; LITERATURE REVIEW," *JURNAL PENGEMBANGAN ILMU DAN PRAKTIK KESEHATAN*, vol. 2, no. 3, hlm. 166–175, Jun 2023.

[8] V. Arinal dan M. Azhari, "Penerapan Regresi Linear Untuk Prediksi Harga Beras Di Indonesia," *Jurnal Sains dan Teknologi*, vol. 5, no. 1, hlm. 341–346, Sep 2023, doi: 10.55338/saintek.v5i1.1417.

[9] N. A. Salsabila, *ANALISIS SENTIMEN PADA MEDIA SOSIAL TWITTER TERHADAP TOKOH GUS DUR MENGGUNAKAN METODE NAIVE BAYES DAN SUPPORT VECTOR MACHINE (SVM)*. 2022.

[10] K. Widi dan A. D. Indriyanti, "Peramalan Penjualan Cookies pada Usaha Cookies Sweetnest Menggunakan Metode Simple Moving Average," *JEISBI (Journal of Emerging Information Systems and Business Intelligence)*, vol. 5, no. 2, hlm. 104–109, 2024.

[11] N. Hidayati, J. Suntoro, dan G. G. Setiaji, "Perbandingan Algoritma Klasifikasi untuk Prediksi Cacat Software dengan Pendekatan CRISP-DM," *Jurnal Sains dan Informatika*, vol. 7, no. 2, hlm. 117–126, Nov 2021, doi: 10.34128/jsi.v7i2.313.

[12] M. R. A. Surya, Martanto, dan U. Hayati, "ANALISIS SENTIMEN ULASAN PENGGUNA OVO MENGGUNAKAN

ALGORITMA NAIVE BAYES PADA GOOGLE PLAY STORE," *JATI (Jurnal Mahasiswa Teknik Informatika)*, vol. 8, no. 3, hlm. 2780–2786, 2024.

[13] E. Erlin, Y. Desnelita, N. Nasution, L. Suryati, dan F. Zoromi, "Dampak SMOTE terhadap Kinerja Random Forest Classifier berdasarkan Data Tidak seimbang," *MATRIK : Jurnal Manajemen, Teknik Informatika dan Rekayasa Komputer*, vol. 21, no. 3, hlm. 677–690, Jul 2022, doi: 10.30812/matrik.v21i3.1726.

[14] S. Amaliah, M. Nusrang, dan Aswi, "Penerapan Metode Random Forest Untuk Klasifikasi Varian Minuman Kopi di Kedai Kopi Konijiwa Bantaeng," *VARIANSI: Journal of Statistics and Its application on Teaching and Research*, vol. 4, no. 3, hlm. 121–127, Des 2022, doi: 10.35580/variansiunm31.

[15] N. N. Sholihah dan A. Hermawan, "IMPLEMENTATION OF RANDOM FOREST AND SMOTE METHODS FOR ECONOMIC STATUS CLASSIFICATION IN CIREBON CITY," *Jurnal Teknik Informatika (JUTIF)*, vol. 4, no. 6, hlm. 1387–1397, Des 2023, doi: 10.52436/1.jutif.2023.4.6.1135.

[16] Kristiawan dan A. Widjaja, "Perbandingan Algoritma Machine Learning dalam Menilai Sebuah Lokasi Toko Ritel," *Jurnal Teknik Informatika dan Sistem Informasi*, vol. 7, no. 1, hlm. 35–46, Apr 2021, doi: 10.28932/jutisi.v7i1.3182.

[17] N. A'ayunnisa, Y. Salim, dan H. Azis, "Analisis performa metode Gaussian Naïve Bayes untuk klasifikasi citra tulisan tangan karakter arab," *Indonesian Journal of Data and Science (IJODAS)*, vol. 3, no. 3, hlm. 115–121, Des 2022.

[18] M. D. Purbolaksono, M. I. Tantowi, A. I. Hidayat, dan Adiwijaya, "Perbandingan Support Vector Machine dan Modified Balanced Random Forest dalam Deteksi Pasien Penyakit Diabetes," *Jurnal RESTI (Rekayasa Sistem dan Teknologi Informasi)*, vol. 5, no. 2, hlm. 393–399, Apr 2021, doi: 10.29207/resti.v5i2.3008.