Jurnal Pendidikan dan Teknologi Indonesia (JPTI)
Vol. 5, No. 9, September 2025, Hal. 2843-2855
DOI: https://doi.org/10.52436/1.jpti.1084
p-ISSN: 2775-4227
e-ISSN: 2775-4219

Perancangan Model Pengenalan Emosi Berbasis Teks Berbahasa Inggris Menggunakan Lima Algoritma Klasifikasi Machine Learning

Dorkas Tri Oktavena Soebroto[1], Nina Setiyawati[*2], Dwi Hosanna Bangkalang[3]

[1,2]Teknik Informatika, Fakultas Teknologi Informasi, Universitas Kristen Satya Wacana, Indonesia
[3]Sistem Informasi, Fakultas Teknologi Informasi, Universitas Kristen Satya Wacana, Indonesia

Email: [1]672020127@student.uksw.edu, [2]nina.setiyawati@uksw.edu, [3]dwihosanna.bangkalang@uksw.eduss

Abstrak

Saat ini, interaksi melalui platform teks marak dilakukan seperti media sosial, layanan pelanggan daring, dan forum diskusi. Oleh karena itu analisis *Text Emotion Recognition* (TER) menjadi sangat krusial. Namun, emosi dalam teks bersifat kompleks, kontekstual, dan sering kali bersifat implisit, sehingga tidak dapat dengan mudah diidentifikasi menggunakan pendekatan konvensional. TER merupakan proses identifikasi dan klasifikasi emosi dalam teks secara otomatis menggunakan teknik *machine learning*. Penelitian ini bertujuan untuk merancang model pengklasifikasian emosi berbasis teks menggunakan berbagai algoritma *classification machine learning*. Data teks dikategorikan ke dalam enam emosi utama: *sadness, joy, love, anger, fear, dan surprise*. Tahapan *preprocessing* meliputi tokenisasi, *stop word removal*, dan *lemmatization*, sedangkan fitur teks direpresentasikan dalam bentuk numerik menggunakan *Term Frequency-Inverse Document Frequency* (TF-IDF). Lima algoritma *machine learning*, yaitu *Support Vector Machine* (SVM), *K-Nearest Neighbor* (KNN), *Random Forest, Logistic Regression*, dan *Multinomial Naïve Bayes*, digunakan dalam pelatihan dan evaluasi model. Hasil penelitian menunjukkan bahwa model SVM dan *Random Forest* memberikan performa terbaik dengan akurasi mencapai 85% pada data uji. Dengan demikian, pendekatan *machine learning*, khususnya SVM dan *Random Forest*, terbukti efektif dalam klasifikasi teks emosi dan dapat diterapkan dalam berbagai aplikasi analisis sentimen. Model TER hasil eksperimental akan dilanjutkan untuk diimplementasikan pada aplikasi penanganan psikologis korban kekerasan berbasis gender (KBG) yang telah dibangun pada penelitian sebelumnya.

Kata kunci: *Klasifikasi Teks, Machine Learning, Text Emotion Recognition, TF-IDF.*

Design of English Text-Based Emotion Recognition Model Using Five Machine Learning Classification Algorithms

Abstract

Nowadays, interactions through text platforms are rampant such as social media, online customer service, and discussion forums. Therefore, Text Emotion Recognition (TER) analysis becomes very crucial. However, emotions in text are complex, contextual, and often implicit, so they cannot be easily identified using conventional approaches. TER is the process of automatically identifying and classifying emotions in text using machine learning techniques. This study aims to design a text-based emotion classification model using various classification machine learning algorithms. Text data is categorized into six main emotions: sadness, joy, love, anger, fear, and surprise. The preprocessing stages include tokenization, stop word removal, and lemmatization, while text features are represented in numeric form using Term Frequency-Inverse Document Frequency (TF-IDF). Five machine learning algorithms, namely Support Vector Machine (SVM), K-Nearest Neighbor (KNN), Random Forest, Logistic Regression, and Multinomial Naïve Bayes, are used in model training and evaluation. The results show that the SVM and Random Forest models provide the best performance with an accuracy of 85% on the test data. Thus, the machine learning approach, especially SVM and Random Forest, has proven effective in classifying emotional text and can be applied in various sentiment analysis applications. The TER model from the experimental results will be continued to be implemented in the psychological handling application for victims of gender-based violence (GBV) that has been built in previous research.

Keywords: *Machine Learning, Text Classification, Text Emotion Recognition, TF-IDF.*

p-ISSN: 2775-4227
e-ISSN: 2775-4219

1. PENDAHULUAN

Analisis sentimen adalah metode untuk mengidentifikasi sikap, perasaan, dan emosi seseorang terhadap suatu objek tertentu, seperti terhadap individu, aktivitas, organisasi, layanan, subjek, atau produk. Pengenalan emosi atau *Emotion Recognition* (ER) merupakan bagian dari analisis sentimen karena bertujuan untuk mengenali emosi secara spesifik, bukan hanya menyatakan apakah sentimen tersebut positif, negatif, atau netral . Informasi yang terkandung dalam ekspresi emosional pengguna dapat dimanfaatkan untuk berbagai tujuan strategis, mulai dari peningkatan pengalaman pengguna, deteksi krisis psikologis, hingga pengambilan keputusan bisnis yang lebih tepat sasaran.*Emotion Recognition* (ER) dapat diambil dari berbagai sumber, seperti ekspresi wajah, suara, gestur tubuh dan juga teks.

Salah satu bentuk penerapan *Emotion Recognition* (ER) dalam bentuk teks dengan menggunakan teknik *machine learning* adalah *Text Emotion Recognition* (TER). TER bertujuan untuk mengidentifikasi emosi atau perasaan yang terkandung dalam sebuah teks, dimana teks akan dibagi dalam beberapa kategori emosi spesifik, seperti *sadness, joy, love, anger, fear,* dan *surprise*. Saat ini, interaksi melalui *platform* teks marak dilakukan seperti media sosial, layanan pelanggan daring, dan forum diskusi. Oleh karena itu analisis TER menjadi sangat krusial. Namun, emosi dalam teks bersifat kompleks, kontekstual, dan sering kali bersifat implisit, sehingga tidak dapat dengan mudah diidentifikasi menggunakan pendekatan konvensional.

Deteksi emosi pada hakikatnya merupakan masalah klasifikasi *multi-label,* karena satu kalimat dapat mengandung lebih dari satu emosi secara bersamaan, seperti marah dan sedih. Hal ini meningkatkan kompleksitas masalah dan membuat proses klasifikasi menjadi lebih sulit dalam konteks bahasa alami [2]. Meskipun banyak penelitian telah dilakukan dalam klasifikasi emosi berbasis teks, masih terdapat beberapa tantangan yang perlu diatasi seperti pemilihan algoritma yang paling optimal serta teknik *preprocessing* yang efektif untuk meningkatkan akurasi model. Oleh karena itu, penelitian ini bertujuan untuk merancang model klasifikasi emosi dalam teks menggunakan berbagai algoritma *machine learning* seperti *Support Vector Machine* (SVM) [3], *K-Nearest Neighbors* (KNN) [3], *Random Forest* [4], *Logistic Regression*, dan *Multinomial Naïve Bayes* yang telah banyak digunakan secara luas dalam klasifikasi teks karena efektivitasnya dalam menangani data tidak terstruktur [5], [6], [7]. Kelima algoritma tersebut dipilih untuk memberikan pemahaman yang menyeluruh mengenai performa metode berbasis jarak, probabilistik, linier, dan ansambel dalam menyelesaikan masalah klasifikasi emosi pada teks. Perbandingan performa antar algoritma ini bertujuan untuk mengidentifikasi model paling optimal dalam konteks TER pada teks berbahasa Inggris. Pemilihan algoritma yang tepat akan sangat mempengaruhi hasil analisis sentimen terutama pada konteks teks yang kompleks [8]. Selain algoritma, *preprocessing* dan pemilihan fitur ekstraksi juga sangat mempengaruhi hasil klasifikasi [9]. Teknik ekstraksi fitur seperti *Term Frequency-Inverse Document Frequency* (TF-IDF) meningkatkan akurasi klasifikasi dengan memberikan bobot lebih tinggi pada kata-kata yang lebih signifikan dalam korpus data [10].

Penelitian mengenai *Text Emotion Recognition* (TER) telah banyak dilakukan dengan berbagai pendekatan. Agastya dan Aripin [11] menggunakan *Multinomial Naïve Bayes* untuk pemetaan emosi dalam teks berbahasa Indonesia, di mana pendekatan probabilistik ini terbukti cukup efektif dalam klasifikasi teks berdasarkan sentimen. Sementara itu, Mahing et al. [12] membandingkan performa *Support Vector Machine* (SVM) dan *Random Forest* dalam klasifikasi teks yang mengandung stres. Hasil penelitian mereka menunjukkan bahwa SVM unggul dalam menemukan pola teks yang kompleks, sedangkan *Random Forest* lebih efektif dalam menangani fitur yang lebih bervariasi. Zamsuri et al. [13] menerapkan metode *K-Nearest Neighbor* (KNN) dalam klasifikasi multi-emosi pada teks bahasa Indonesia, yang menunjukkan bahwa KNN dapat bekerja dengan baik terutama ketika parameter yang digunakan dioptimalkan. Di sisi lain, penelitian oleh Basbeth dan Fudholi [14] mengeksplorasi model *deep learning* seperti BERT, RoBERTa, dan Distil-BERT dalam klasifikasi emosi, dengan hasil yang menunjukkan bahwa model *deep learning* memiliki keunggulan dalam memahami konteks bahasa alami dibandingkan dengan metode tradisional.

Dalam berbagai penelitian, SVM terbukti efektif dalam menemukan *hyperplane optimal* untuk pemisahan kelas, sedangkan KNN menunjukkan keunggulan dalam menangani data berbasis jarak [15]. *Naïve Bayes* dengan pendekatan probabilistiknya banyak diterapkan dalam tugas seperti deteksi spam dan analisis emosi [11], [16]. Dalam konteks klasifikasi emosi dan berita, penelitian terdahulu telah mengeksplorasi berbagai pendekatan untuk meningkatkan akurasi model. Penggunaan *Multinomial Naïve Bayes* dalam pemetaan emosi menunjukkan hasil yang baik dalam mengklasifikasikan teks berdasarkan sentimen [12]. Studi lain membandingkan efektivitas SVM dan *Random Forest* dalam klasifikasi teks stres, serta penerapan KNN dalam klasifikasi multi-emosi, yang menunjukkan keunggulan masing-masing metode dalam menangani data tertentu [13], [17].

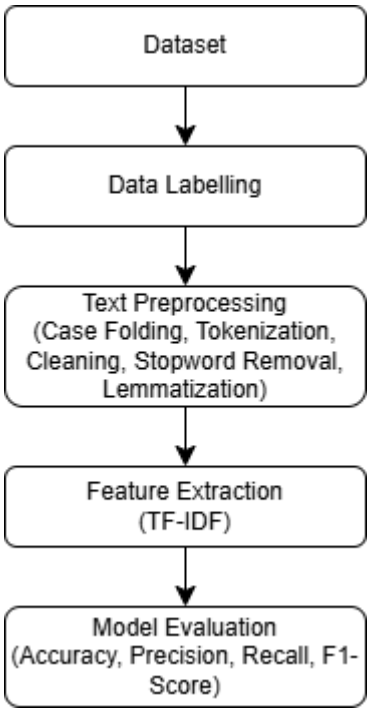
Penelitian ini memiliki beberapa perbedaan dan inovasi dibandingkan dengan penelitian terdahulu. Jika penelitian sebelumnya hanya membandingkan satu atau dua algoritma, penelitian ini menguji lima algoritma klasifikasi sekaligus, yaitu SVM, KNN, *Random Forest*, *Logistic Regression*, dan *Multinomial Naïve Bayes*, sehingga memberikan analisis yang lebih komprehensif mengenai performa masing-masing metode. Selain itu,

p-ISSN: 2775-4227
e-ISSN: 2775-4219

penelitian ini menggunakan teknik ekstraksi fitur *Term Frequency-Inverse Document Frequency* (TF-IDF), yang memberikan bobot lebih tinggi pada kata-kata signifikan dalam korpus data, sehingga diharapkan mampu meningkatkan akurasi klasifikasi emosi. Dalam konteks multi-emosi, TF-IDF berperan dalam membedakan kata-kata yang menjadi indikator khas dari emosi tertentu, sehingga model menjadi lebih peka terhadap perbedaan makna kata yang mirip atau digunakan dalam konteks yang berbeda. Banyak penelitian sebelumnya hanya fokus pada klasifikasi sentimen (positif, negatif, netral) atau kategori emosi yang lebih sedikit, sementara penelitian ini mengklasifikasikan teks ke dalam enam kategori emosi utama, yaitu *sadness, joy, love, anger, fear,* dan *surprise*. Dari sisi pengolahan data, penelitian ini menerapkan tahapan *preprocessing* yang mencakup *tokenization, stop word removal*, dan *lemmatization* untuk memastikan kualitas data yang lebih baik sebelum proses klasifikasi dilakukan. Dengan adanya perbedaan tersebut, penelitian ini diharapkan dapat memberikan kontribusi baru dalam pengembangan model klasifikasi emosi teks berbasis *machine learning* khususnya *dalam Text Emotion Recognition* berbasis teks berbahasa Inggris. Selain itu, model TER hasil eksperimental akan dilanjutkan untuk diimplementasikan pada aplikasi penanganan psikologis korban kekerasan berbasis gender (KBG) yang telah dibangun pada penelitian sebelumnya [18].

2. METODE PENELITIAN

Penelitian ini dilakukan dengan beberapa tahapan, yaitu pengumpulan data, pelabelan, pra-pemrosesan teks, ekstrasi fitur (menggunakan TF-IDF), pemodelan, dan evaluasi model.



Gambar 1. Tahapan Penelitian

2.1. Dataset

Penelitian ini menggunakan dataset emosi teks berbahasa Inggris yang diambil dari *Kaggle* [19]. Dataset ini dibagi menjadi dua bagian, yaitu *training set dan test set*, yang detailnya disajikan pada Tabel 2.

Training set berisi 16.000 data teks dan label emosi yang digunakan untuk melatih model dan mempelajari pola-pola dalam data. *Test set* berisi 2.000 data teks yang digunakan untuk menguji performa model yang belum pernah dilihat sebelumnya, sehingga memberikan evaluasi yang objektif terhadap kemampuan generalisasi model.

Setiap data teks dalam dataset diberi label dengan emosi dasar, yaitu: sedih (*sadness*), senang (*joy*), cinta (*love*), marah (*anger*), takut (*fear*), dan terkejut (*surprise*). Label-label ini direpresentasikan dengan angka sebagai berikut: 0 untuk sedih, 1 untuk senang, 2 untuk cinta, 3 untuk marah, 4 untuk takut, dan 5 untuk terkejut. Beberapa contoh data dan labelnya dapat dilihat pada Tabel 1.

p-ISSN: 2775-4227
e-ISSN: 2775-4219

Tabel 1. Contoh Data dan Label

No	Teks	Label
1	i didnt feel humiliated	0
2	i can go from feeling so hopeless to so damned hopeful just from being around someone who cares and is awake	0
3	im grabbing a minute to post i feel greedy wrong	3
4	i am ever feeling nostalgic about the fireplace i will know that it is still on the property	2
5	i am feeling grouchy	3
6	ive been taking or milligrams or times recommended amount and ive fallen asleep a lot faster but i also feel like so funny	5
7	i feel as confused about life as a teenager or as jaded as a year old man	4
8	i do not feel reassured anxiety is on each side	1

Tabel 2. Deskripsi Dataset

Nama Dataset	Deskripsi Data	Jumlah Data	Data Teks	Label
training.csv	Data untuk melatih model	16.000	Teks	Emosi (sadness, joy, love, anger, fear, surprise)
test.csv	Data untuk menguji model	2.000	Teks	Emosi (sadness, joy, love, anger, fear, surprise)

2.2. Text Preprocessing

Tahap *text preprocessing* bertujuan untuk membersihkan dan standarisasi data teks mentah sebelum diproses lebih lanjut. Membersihkan data dilakukan agar data yang digunakan menjadi seragam sehingga tidak menimbulkan kerancuan atau menghilangkan data yang tidak valid. Standarisasi data digunakan untuk menyaring kata yang tidak diperlukan dan dapat meminimalisir kesalahan kata, dengan menyamakan atau mengelompokkan kata. Tahapan ini berpengaruh dalam data dan performa model.
Proses tahapan yang dilakukan dalam *text preprocessing* yaitu:

a. *Case Folding*
   Proses ini dilakukan untuk menyamakan semua teks menjadi huruf kecil menggunakan fungsi *lower()* pada string Python, sehingga model tidak membedakan antara huruf kapital dan huruf kecil.
b. *Tokenization*
   Proses ini dilakukan untuk memecah teks menjadi kata-kata individual.
c. *Cleaning*
   Pada proses ini dilakukan penghapusan karakter-karakter yang tidak relevan, seperti simbol, tanda baca, URL, dan angka, menggunakan *regular expression* dalam Python.
d. *Stop Word Removal*
   Proses ini dilakukan untuk menghapus kata-kata umum dalam bahasa Inggris yang sering muncul dalam teks tetapi tidak membawa banyak informasi, seperti "*the*", "*a*", "*is*".
e. *Lemmatization*
   Proses ini dilakukan untuk mengubah kata menjadi bentuk dasarnya (*lemma*).

2.3. Ekstraksi Fitur (TF-IDF)

Setelah tahap *preprocessing*, teks yang telah dibersihkan dan distandarisasi diubah menjadi representasi numerik yang dapat diproses oleh algoritma *machine learning*. Ekstraksi fitur adalah teknik reduksi dimensi yang mengubah sekumpulan besar data mentah menjadi kategori-kategori yang lebih kecil untuk mempercepat proses pengolahan data [1]. Dalam penelitian ini, metode *Term Frequency-Inverse Document Frequency* (TF-IDF) digunakan untuk mengekstraksi fitur dari teks. Implementasi TF-IDF dilakukan menggunakan *TfidfVectorizer* dari library scikit-learn dalam Python.

TF-IDF mengukur pentingnya suatu kata dalam sebuah dokumen relatif terhadap seluruh koleksi dokumen, sehingga memungkinkan model untuk membedakan antara kata-kata yang umum dan kata-kata yang spesifik untuk kategori emosi tertentu. Metode ini telah digunakan secara luas dalam berbagai penelitian klasifikasi teks, termasuk analisis sentimen dan klasifikasi topik [14].

p-ISSN: 2775-4227
e-ISSN: 2775-4219

TF-IDF dihitung dengan mengalikan frekuensi kemunculan suatu kata dalam dokumen dengan kebalikan dari frekuensi kemunculan kata tersebut di seluruh koleksi dokumen. Secara matematis, penulisan rumus TF-IDF dapat dilihat dalam Rumus 1 [17]:

$$tf - idf(t,d) = tf(t,d) \times idf(t) \quad (1)$$

Dimana:

- **tf(t,d)** adalah *Term Frequency*, yang mengukur frekuensi kemunculan *term* (t) dalam dokumen (d). Biasanya dihitung sebagai jumlah kemunculan *term* (t) dalam dokumen (d) dibagi dengan jumlah total *term* (t) dalam dokumen (d).
- **idf(t)** adalah *Inverse Document Frequency*, yang mengukur seberapa umum atau langka suatu *term* (t) di seluruh koleksi dokumen (d) yang dihitung menggunakan Rumus 2.

$$idf_t = log \frac{N}{df_t} \quad (2)$$

Dimana:

- **N** adalah jumlah total dokumen dalam koleksi.
- **df_t** adalah *Document Frequency*, yaitu jumlah dokumen yang mengandung kata t. Semakin kecil df_t, semakin besar idf_t.

2.4. Algoritma Klasifikasi (Modelling)

Penelitian ini menggunakan beberapa algoritma klasifikasi *machine learning* untuk mengklasifikasikan emosi dalam teks, yaitu: *Support Vector Machine* (SVM), *K-Nearest Neighbors* (KNN), *Random Forest*, *Logistic Regression*, dan *Multinomial Naïve Bayes*. Algoritma-algoritma ini telah digunakan secara luas dalam penelitian sebelumnya untuk pengklasifikasian teks [11], [20].

2.4.1. Support Vector Machine (SVM)

SVM adalah algoritma yang digunakan untuk klasifikasi dan regresi [3]. SVM bertujuan untuk menemukan *hyperplane optimal* yang memisahkan data ke dalam kelas-kelas yang berbeda dengan margin terbesar. Implementasi SVM dalam penelitian ini menggunakan *svm*.SVC dari library scikit-learn. Persamaan dasar SVM dapat dilihat pada Rumus 3.

$$f(x) = wt^T x + b \quad (3)$$

Dimana:

- **wt** adalah vektor bobot.
- **x** adalah vektor fitur.
- **b** adalah bias [21].

Detail hyperparameter dan hasil tuning dapat dilihat pada Tabel 3.

2.4.2. K-Nearest Neighbors (KNN)

K-Nearest Neighbors (KNN) adalah algoritma yang sederhana dan intuitif. KNN mengklasifikasikan data baru berdasarkan kelas dari *k* tetangga terdekatnya [3]. Jarak antara data dihitung menggunakan metrik jarak, seperti jarak *Euclidean* yang didefinisikan dalam Rumus 4.

$$d(x,y) = \sqrt{\sum_{k=1}^{n} (xi - yi)^2} \quad (4)$$

Dimana:

- **x** dan **y** adalah dua titik data
- **xi** dan **yi** adalah nilai fitur ke-i dari x dan y [22].

Implementasi KNN dalam penelitian ini menggunakan *KNeighborsClassifier* dari library scikit-learn. Detail parameter tuning dapat dilihat pada Tabel 3.

p-ISSN: 2775-4227
e-ISSN: 2775-4219

2.4.3. Random Forest

Random Forest adalah algoritma yang menggabungkan beberapa *decision tree* untuk meningkatkan akurasi dan mengurangi *overfitting* [23]. *Random Forest* menggunakan teknik *bagging (bootstrap aggregating)* untuk membuat *subset* data dan *random subspace* untuk memilih fitur secara acak untuk setiap *decision tree*. Implementasi dilakukan menggunakan *RandomForestClassifier* dari library scikit-learn. Rentang parameter yang diuji serta hasil tuning dapat dilihat pada Tabel 3.

2.4.4. Logistic Regression

Logistic Regression adalah algoritma yang digunakan untuk klasifikasi biner. *Logistic Regression* memodelkan probabilitas kelas menggunakan fungsi logistik (*sigmoid*) yang didefinisikan dalam Rumus 5.

$$P(y = 1|x) = \frac{1}{1 + e^{\beta 0 + \beta_1 x_1 + \beta_2 x_2 + \cdots \beta_n x_n}} \qquad (5)$$

Dimana:

- **P(y=1|x)** adalah probabilitas bahwa hasil *y* bernilai 1 ketika diberikan sekumpulan variabel *predictor* x_1, x_2, x_3, … x_n.
- $\beta_0, \beta_1, \beta_2, \ldots \beta_n$ adalah koefisien yang diperkirakan oleh model [24].

Implementasi Logistic Regression dalam penelitian ini menggunakan *Logistic Regression* dari library scikit-learn. Rincian parameter tuning tersedia pada Tabel 3.

2.4.5. Multinomial Naïve Bayes

Multinomial Naïve Bayes adalah algoritma yang mengasumsikan bahwa fitur-fitur data independen secara kondisional dan mengikuti distribusi multinomial. Rumus *Multinomial Naïve Bayes* didefinisikan dalam Rumus 6.

$$P(c|x) = \frac{P(x|c) \times P(c)}{P(x)} \qquad (6)$$

Dimana:

- **P(c|x)** adalah probabilitas hipotesis berdasarkan kondisi *(*probabilitas *posterior)*.
- **P(x|c)** adalah probabilitas berdasarkan kondisi dari hipotesis
- **P(x)** adalah probabilitas *predictor*[25].

Model ini diimplementasikan menggunakan *MultinomialNB* dari scikit-learn. Parameter tuning dapat dilihat dalam Tabel 3.

2.4.6. Tabel Hyperparameter Search Space

Tabel 3 merupakan rangkuman hyperparameter yang diuji pada masing-masing algoritma serta kombinasi parameter terbaik yang diperoleh dari proses tuning.

Tabel 3. Hyperparameter Search Space

Algoritma	Hyperparameter	Nilai yang Diuji	Parameter Terbaik
SVM	kernel	['linear', 'rbf']	'rbf'
	C	[0.1, 1, 10, 100]	100
	gamma	[1, 0.1, 0.01, 0.001]	0.01
KNN	n_neighbors	[3, 5, 7, 9, 11]	11
	weights	['uniform', 'distance']	'distance'
	metric	['euclidean', 'manhattan']	'euclidean'
Random Forest	n_estimators	[50, 100, 200]	200
	max_depth	[None, 10, 20]	None
	min_samples_split	[2, 5, 10]	2
	min_samples_leaf	[1, 2, 4]	1
Logistic Regression	penalty	['l1', 'l2']	'l2'

p-ISSN: 2775-4227
e-ISSN: 2775-4219

	C	[0.001, 0.01, 0.1, 1, 10, 100]	1
	solver	['liblinear']	'liblinear'
Multinomial Naïve Bayes	alpha	[0.1, 0.5, 1.0, 1.5, 2.0]	0.1
	fit_prior	[True, False]	True

2.5. Evaluasi Model

Setelah dilakukan percobaan klasifikasi teks menggunakan algoritma *machine learning* dan ekstraksi fitur, maka dilakukan pengujian tingkat akurasi pada tiap-tiap model [5]. Performa model dievaluasi menggunakan beberapa metrik evaluasi, yaitu [12]:

a. *Accuracy*: Mengukur persentase prediksi yang benar dari total prediksi.
b. *Precision*: Mengukur proporsi prediksi positif yang benar dari seluruh prediksi positif.
c. *Recall*: Mengukur proporsi data aktual positif yang berhasil diprediksi dengan benar.
d. *F1-score*: Rata-rata harmonik antara *precision* dan *recall*.

Metrik-metrik ini dihitung menggunakan fungsi *accuracy_score* dan *classification_report* dari library scikit-learn dalam Python. Metrik-metrik ini dievaluasi pada *test set* untuk mengukur kemampuan generalisasi model terhadap data yang belum pernah dilihat sebelumnya.

3. HASIL DAN PEMBAHASAN

3.1. Hasil Klasifikasi Model SVM

Model algoritma SVM memiliki akurasi sebesar 0.85. Hal tersebut menunjukkan bahwa dari total prediksi yang dilakukan model ini, 85% prediksi adalah benar.

Dari Tabel 4 terlihat bahwa terdapat 2 label dengan *recall* tertinggi, yaitu label *sadness* dan label *joy*. Label *sadness* memiliki nilai 0.90. Hal ini menunjukkan bahwa model mampu mengidentifikasi 90% label *sadness* dengan benar. Namun, nilai yang ditampilkan pada Tabel 4 merupakan hasil pembulatan dari nilai yang terdapat pada *Confusion Matrix* sebesar 0.8982. Artinya, masih terdapat sekitar 0.1018 data *sadness* yang tidak terklasifikasi dengan benar (*False Negative*), yang kemudian terprediksi ke label *joy* sebesar 0.0397, *love* 0.0075, *anger* 0.0257, *fear* 0.0214, dan *surprise* 0.0075 seperti terlihat pada Gambar 2.

Selain itu *recall* pada label *joy* juga memiliki nilai 0.90. Hal ini menunjukkan bahwa model mampu mengidentifikasi 90% label *joy* dengan benar. Namun, nilai yang ditampilkan pada Tabel 4 merupakan hasil pembulatan dari nilai yang terdapat pada *Confusion Matrix* sebesar 0.9011. Artinya, masih terdapat sekitar 0.0989 data *joy* yang tidak terklasifikasi dengan benar (*False Negative*), yang kemudian terprediksi ke label *sadness* sebesar 0.0364, *love* 0.0308, *anger* 0.0140, *fear* 0.121, dan *surprise* 0.0056 seperti yang terlihat juga pada Gambar 2.

Tabel 4. SVM

label	precision	recall	f1-score	support
sadness	0.88	0.90	0.89	933
joy	0.86	0.90	0.88	1072
love	0.79	0.67	0.73	261
anger	0.86	0.84	0.85	432
fear	0.82	0.82	0.82	387
surprise	0.77	0.65	0.70	115
accuracy			0.85	3200
macro avg	0.83	0.80	0.81	3200
weighted avg	0.85	0.85	0.85	3200

p-ISSN: 2775-4227
e-ISSN: 2775-4219

Gambar 2. *Confusion Matrix SVM*

3.2. Hasil Klasifikasi Model KNN

Model algoritma KNN memiliki akurasi sebesar 0.79. Hal tersebut menunjukkan bahwa dari total prediksi yang dilakukan oleh model ini, 79% prediksi adalah benar.

Dari Tabel 5 terlihat bahwa *recall* pada label *joy* adalah 0.90. Hal ini menunjukkan bahwa model mampu mengidentifikasi 90% label *joy* dengan benar. Namun, nilai yang ditampilkan pada Tabel 5 merupakan hasil pembulatan dari nilai yang terdapat pada *Confusion Matrix* sebesar 0.8983. Artinya, masih terdapat sekitar 0.1017 data *joy* yang tidak terklasifikasi dengan benar (*False Negative*), yang kemudian terprediksi ke label *sadness* sebesar 0.0625, *love* 0.0121, *anger* 0.0131, *fear* 0.0131, dan *surprise* 0.0009.

Tabel 5. KNN

label	precision	recall	f1-score	support
sadness	0.79	0.87	0.83	933
joy	0.77	0.90	0.83	1072
love	0.80	0.45	0.58	261
anger	0.84	0.73	0.78	432
fear	0.82	0.70	0.75	387
surprise	0.84	0.44	0.58	115
accuracy			0.79	3200
macro avg	0.81	0.68	0.72	3200
weighted avg	0.79	0.79	0.78	3200

3.3. Hasil Klasifikasi Model Random Forest

Model algoritma *Random Forest* memiliki akurasi sebesar 0.85. Hal tersebut menunjukkan bahwa dari total prediksi yang dilakukan model ini, 85% prediksi adalah benar.

Dari Tabel 6 terlihat bahwa *recall* pada label *joy* adalah 0.91. Hal ini menunjukkan bahwa model mampu mengidentifikasi 91% label *joy* dengan benar. Namun, nilai yang ditampilkan pada Tabel 6 merupakan hasil pembulatan dari nilai yang terdapat pada *Confusion Matrix* sebesar 0.9076. Artinya, masih terdapat sekitar 0.0924

p-ISSN: 2775-4227
e-ISSN: 2775-4219

data *joy* yang tidak terklasifikasi dengan benar (*False Negative*), yang kemudian terprediksi ke label *sadness* sebesar 0.0326, *love* 0.0243, *anger* 0.0168, *fear* 0.0131, dan *surprise* 0.0056 seperti terlihat pada Gambar 3.

Tabel 6. *Random Forest*

label	precision	recall	f1-score	support
sadness	0.91	0.89	0.90	933
joy	0.83	0.91	0.87	1072
love	0.85	0.66	0.74	261
anger	0.86	0.84	0.85	432
fear	0.82	0.83	0.83	387
surprise	0.81	0.69	0.74	115
accuracy			0.85	3200
macro avg	0.85	0.80	0.82	3200
weighted avg	0.86	0.86	0.86	3200

Gambar 3. *Confusion Matrix Random Forest*

3.4. Hasil Klasifikasi Model Logistic Regression

Model algoritma *Logistic Regression* memiliki akurasi 0.84. Hal tersebut menunjukkan bahwa dari total prediksi yang dilakukan model ini, 84% prediksi adalah benar.

Dari Tabel 7 terlihat bahwa *recall* pada label *joy* adalah 0.95. Hal ini menunjukkan bahwa model mampu mengidentifikasi 95% label *joy* dengan benar. Namun, nilai yang ditampilkan pada Tabel 7 merupakan hasil pembulatan dari nilai yang terdapat pada *Confusion Matrix* sebesar 0.9534. Artinya, masih terdapat sekitar 0.0466 data *joy* yang tidak terklasifikasi dengan benar (*False Negative*), yang kemudian terprediksi ke label *sadness* sebesar 0.0205, *love* 0.0159, *anger* 0.065, *fear* 0.0028, dan *surprise* 0.0009.

p-ISSN: 2775-4227
e-ISSN: 2775-4219

Tabel 7. *Logistic Regression*

label	precision	recall	f1-score	support
sadness	0.87	0.92	0.89	933
joy	0.79	0.95	0.86	1072
love	0.86	0.52	0.65	261
anger	0.91	0.76	0.83	432
fear	0.84	0.74	0.79	387
surprise	0.92	0.38	0.54	115
accuracy			0.84	3200
macro avg	0.86	0.71	0.76	3200
weighted avg	0.84	0.84	0.83	3200

3.5. Hasil Klasifikasi Model MultinomialNB

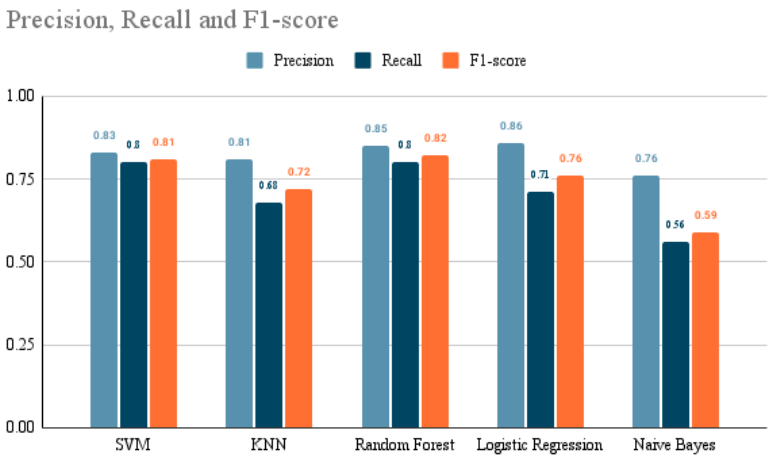
Model algoritma *MultinomialNB* memiliki akurasi 0.74. Hal tersebut menunjukkan bahwa dari total prediksi yang dilakukan model ini, 74% prediksi adalah benar.

Dari Tabel 8 terlihat bahwa *recall* pada label *joy* adalah 0.92. Hal ini menunjukkan bahwa model mampu mengidentifikasi 92% label *joy* dengan benar. Namun, nilai yang ditampilkan pada Tabel 8 merupakan hasil pembulatan dari nilai yang terdapat pada *Confusion Matrix* sebesar 0.9179 Artinya, masih terdapat sekitar 0.0821 data *joy* yang tidak terklasifikasi dengan benar (*False Negative*), yang kemudian terprediksi ke label *sadness* sebesar 0.0494, *love* 0.0121, *anger* 0.0131, *fear* 0.056, dan *surprise* 0.0019.

Tabel 8. *MultinomialNB*

label	precision	recall	f1-score	support
sadness	0.74	0.87	0.80	933
joy	0.71	0.92	0.80	1072
love	0.80	0.29	0.42	261
anger	0.82	0.62	0.71	432
fear	0.79	0.54	0.64	387
surprise	0.72	0.11	0.20	115
accuracy			0.74	3200
macro avg	0.76	0.56	0.59	3200
weighted avg	0.75	0.74	0.72	3200

3.6. Perbandingan Performa Model



Gambar 4. Grafik Perbandingan *Precision, Recall, dan F1-Score*

p-ISSN: 2775-4227
e-ISSN: 2775-4219

Gambar 4 menunjukkan bahwa *Random Forest* dan SVM memberikan hasil yang paling baik diantara yang lain dalam klasifikasi emosi, dengan nilai *F1-score* yang tinggi dan seimbang antara *precision* dan *recall*. *Logistic Regression* memiliki *precision* yang tinggi, tetapi *recall*-nya lebih rendah, sedangkan KNN menunjukkan *trade-off* antara *precision* dan *recall*. *Naive Bayes* memiliki performa yang paling rendah di antara semua model. SVM unggul karena memiliki kemampuan membentuk hyperplane optimal. Sedangkan random forest unggul karena stabilitasnya terhadap noise dan kemampuannya menangani fitur yang tidak linier.

Jika dibandingkan dengan studi Mahing et al. [12], di mana SVM juga menunjukkan performa terbaik dalam klasifikasi teks yang mengandung stres, hasil penelitian ini mendukung bahwa SVM efektif dalam menemukan pola kompleks dalam teks emosi. Sementara itu, penelitian Zamsuri et al. [13] menunjukkan bahwa KNN bekerja baik jika parameter dioptimalkan, hal ini juga terlihat dalam penelitian ini meskipun performanya tetap di bawah SVM dan Random Forest. Penelitian oleh Basbeth dan Fudholi [14] menggunakan pendekatan deep learning dan menunjukkan keunggulan, namun pendekatan *machine learning* seperti yang digunakan dalam penelitian ini juga masih memiliki performa kompetitif dengan sumber daya yang lebih efisien.

Tabel 9. Perbandingan Akurasi

Algoritma	Accuracy (before tuning)	Accuracy (after tuning)
SVM	0.85	0.85
KNN	0.79	0.79
Random Forest	0.85	0.85
Logistic Linier	0.84	0.83
MultinomialNB	0.66	0.73

Dari Tabel 9, dapat dilihat bahwa *hyperparameter tuning* sangat penting dalam pengoptimalan performa model klasifikasi emosi, karena memberikan dampak positif terhadap model. Akurasi model yang meningkat secara signifikan adalah *MultinomialNB*, sedangkan model lain relatif stabil.

4. KESIMPULAN

Dari hasil penelitian ini, perbandingan lima algoritma *machine* dalam klasifikasi emosi teks berbahasa Inggris, menunjukkan bahwa SVM dan *Random Forest* memberikan performa terbaik secara keseluruhan. Kedua model tersebut unggul dalam mengenali emosi yang lebih dominan seperti joy dan sadness. Hal tersebut bisa disebabkan oleh faktor seperti ekspresi emosi yang jelas, frekuensi kemunculan dan jumlah data training, serta karasteristik model.

Penggunaan *hyperparameter tuning* terbukti efektif dalam meningkatkan performa model, khususnya *MultinomialNB*. Penelitian selanjutnya dapat difokuskan pada peningkatan performa model dalam mengenali emosi *love* dan *surprise*, serta mengeksplorasi dataset dan algoritma lain.

DAFTAR PUSTAKA

[1] S. K. Bharti *et al.*, "Text-Based Emotion Recognition Using Deep Learning Approach," *Comput. Intell. Neurosci.*, vol. 2022, pp. 1–8, Aug. 2022, doi: 10.1155/2022/2645381.

[2] N. Ashraf, L. Khan, S. Butt, H.-T. Chang, G. Sidorov, and A. Gelbukh, "Multi-label emotion classification of Urdu tweets," *PeerJ Comput. Sci.*, vol. 8, p. e896, Apr. 2022, doi: 10.7717/peerj-cs.896.

[3] M. J. Al Dujaili, A. Ebrahimi-Moghadam, and A. Fatlawi, "Speech emotion recognition based on SVM and KNN classifications fusion," *Int. J. Electr. Comput. Eng. IJECE*, vol. 11, no. 2, p. 1259, Apr. 2021, doi: 10.11591/ijece.v11i2.pp1259-1264.

[4] S. Hamsa, Y. Iraqi, I. Shahin, and N. Werghi, "An Enhanced Emotion Recognition Algorithm Using Pitch Correlogram, Deep Sparse Matrix Representation and Random Forest Classifier," *IEEE Access*, vol. 9, pp. 87995–88010, 2021, doi: 10.1109/ACCESS.2021.3086062.

[5] L. Efrizoni, S. Defit, M. Tajuddin, and A. Anggrawan, "Komparasi Ekstraksi Fitur dalam Klasifikasi Teks Multilabel Menggunakan Algoritma Machine Learning," *MATRIK J. Manaj. Tek. Inform. Dan Rekayasa Komput.*, vol. 21, no. 3, pp. 653–666, Jul. 2022, doi: 10.30812/matrik.v21i3.1851.

[6] A. Desiani *et al.*, "Perbandingan Klasifikasi Penyakit Kanker Paru-Paru menggunakan Support Vector Machine dan K-Nearest Neighbor," *J. Process.*, vol. 18, no. 1, Apr. 2023, doi: 10.33998/processor.2023.18.1.700.

p-ISSN: 2775-4227
e-ISSN: 2775-4219

[7] M. Sari and G. Mahalisa, "Naive Bayes Classifier Untuk Deteksi Email Spam," *Technol. J. Ilm.*, vol. 15, no. 4, p. 675, Oct. 2024, doi: 10.31602/tji.v15i4.15944.

[8] R. F. P. Pratama and W. Maharani, "Comparative Analysis of Naive Bayes and SVM for Improved Emotion Classification on Social Media," *Edumatic J. Pendidik. Inform.*, vol. 9, no. 1, pp. 11–20, Apr. 2025, doi: 10.29408/edumatic.v9i1.29087.

[9] R. A. Nugroho and I. Cholissodin, "Implementasi Naïve Bayes Classifier untuk Klasifikasi Emosi Tweet Berbahasa Indonesia pada Spark," *J. Pengemb. Teknol. Inf. Dan Ilmu Komput.*, vol. 5, no. 1, pp. 301–310, Jan. 2021.

[10] D. Septiani and I. Isabela, "ANALISIS TERM FREQUENCY INVERSE DOCUMENT FREQUENCY (TF-IDF) DALAM TEMU KEMBALI INFORMASI PADA DOKUMEN TEKS," vol. 1, no. 2, pp. 81–88, 2022.

[11] W. Agastya and Aripin, "Pemetaan Emosi Dominan pada Kalimat Majemuk Bahasa Indonesia Menggunakan Multinomial Naïve Bayes," *J. Nas. Tek. Elektro Dan Teknol. Inf.*, vol. 9, no. 2, pp. 171–179, May 2020, doi: 10.22146/jnteti.v9i2.157.

[12] N. Fathirachman Mahing, A. Lazuardi Gunawan, A. Foresta Azhar Zen, F. Abdurrachman Bachtiar, and S. Agung Wicaksono, "Klasifikasi Tingkat Stress dari Data Berbentuk Teks dengan Menggunakan Algoritma Support Vector Machine (SVM) dan Random Forest," *J. Teknol. Inf. Dan Ilmu Komput.*, vol. 11, no. 5, pp. 1067–1076, Oct. 2024, doi: 10.25126/jtiik.1078010.

[13] A. Zamsuri, S. Defit, and G. W. Nurcahyo, "Classification of Multiple Emotions in Indonesian Text Using The K-Nearest Neighbor Method," *J. Appl. Eng. Technol. Sci. JAETS*, vol. 4, no. 2, pp. 1012–1021, Jun. 2023, doi: 10.37385/jaets.v4i2.1964.

[14] F. Basbeth and D. H. Fudholi, "Klasifikasi Emosi Pada Data Text Bahasa Indonesia Menggunakan Algoritma BERT, RoBERTa, dan Distil-BERT," *J. MEDIA Inform. BUDIDARMA*, vol. 8, no. 2, p. 1160, Apr. 2024, doi: 10.30865/mib.v8i2.7472.

[15] R. Nanda, E. Haerani, S. K. Gusti, and S. Ramadhani, "Klasifikasi Berita Menggunakan Metode Support Vector Machine," *J. Nas. Komputasi Dan Teknol. Inf. JNKTI*, vol. 5, no. 2, pp. 269–278, Apr. 2022, doi: 10.32672/jnkti.v5i2.4193.

[16] E. I. Setiawan, S. Johanes, A. T. Hermawan, and Y. Yamasari, "Deteksi Validitas Berita pada Media Sosial Twitter dengan Algoritma Naive Bayes," *J. Intell. Syst. Comput.*, vol. 3, no. 2, pp. 55–60, Oct. 2021, doi: 10.52985/insyst.v3i2.164.

[17] I. M. Karo Karo, R. Romia, S. Dewi, and P. M. Fadilah, "Hoax Detection on Indonesian Tweets using Naïve Bayes Classifier with TF-IDF," *J. Inf. Syst. Res. JOSH*, vol. 4, no. 3, pp. 914–919, Apr. 2023, doi: 10.47065/josh.v4i3.3317.

[18] D. H. Bangkalang, N. Setiyawati, R. Tanone, H. P. Chernovita, and Y. T. B. Tacoh, "A REQUIREMENT ENGINEERING IN REPORTING AND COUNSELING-BASED ASSISTANCE APPLICATION FOR VICTIMS OF VIOLENCE AGAINST WOMEN," *J. Ris. Inform.*, vol. 3, no. 4, pp. 311–318, Sep. 2021, doi: 10.34288/jri.v3i4.256.

[19] "Emotion Dataset for Emotion Recognition Tasks." Accessed: Nov. 07, 2024. [Online]. Available: https://www.kaggle.com/datasets/parulpandey/emotion-dataset

[20] K. Ciptady, M. Harahap, J. Jonvin, Y. Ndruru, and I. Ibadurrahman, "Prediksi Kualitas Kopi Dengan Algoritma Random Forest Melalui Pendekatan Data Science," *Data Sci. Indones. DSI*, vol. 2, no. 1, Sep. 2022, doi: 10.47709/dsi.v2i1.1708.

[21] L. K. Ramasamy, S. Kadry, Y. Nam, and M. N. Meqdad, "Performance analysis of sentiments in Twitter dataset using SVM models," *Int. J. Electr. Comput. Eng. IJECE*, vol. 11, no. 3, p. 2275, Jun. 2021, doi: 10.11591/ijece.v11i3.pp2275-2284.

[22] A. B. P. Negara, H. Muhardi, and F. Sajid, "Perbandingan Algoritma Klasifikasi terhadap Emosi Tweet Berbahasa Indonesia," *J. Edukasi Dan Penelit. Inform. JEPIN*, vol. 7, no. 2, p. 242, Aug. 2021, doi: 10.26418/jp.v7i2.48198.

[23] D. N. I. Huda, C. Prianto, and R. M. Awangga, "ANALISIS SENTIMEN PERBANDINGAN LAYANAN JASA PENGIRIMAN KURIR PADA ULASAN PLAY STORE MENGGUNAKAN METODE DECISION TREE DAN RANDOM FOREST," *J. Ilm. Inform.*, vol. 11, no. 02, pp. 150–158, Sep. 2023, doi: 10.33884/jif.v11i02.7952.

p-ISSN: 2775-4227
e-ISSN: 2775-4219

[24] L. Ketsbaia, B. Issac, X. Chen, and S. M. Jacob, "A Multi-Stage Machine Learning and Fuzzy Approach to Cyber-Hate Detection," *IEEE Access*, vol. 11, pp. 56046–56065, 2023, doi: 10.1109/ACCESS.2023.3282834.

[25] B. Wahyu Andrian, F. Adline Twince Tobing, I. Zuhdi Pane, and A. Kusnadi, "Implementation of Naïve Bayes Algorithm in Sentiment Analysis of Twitter Social Media Users Regarding Their Interest to Pay the Tax," *Int. J. Sci. Technol. Manag.*, vol. 4, no. 6, pp. 1733–1742, Nov. 2023, doi: 10.46729/ijstm.v4i6.1015.