

Implementasi Algoritma *Support Vector Machine* dalam Klasifikasi Deteksi Depresi dari Postingan pada Media Sosial

Kameliya Putri^{a1}, Made Agung Raharja^{a2}

^aProgram Studi Informatika, Fakultas Matematika dan Ilmu Pengetahuan Alam
Universitas Udayana, Bali
Jln. Raya Kampus UNUD, Bukit Jimbaran, Kuta Selatan, Badung, 08261, Bali, Indonesia
¹kameliyaputri8748@gmail.com
²made.agung@unud.ac.id

Abstract

Mental health issues, such as depression, have significant impacts on individuals and society. Early identification and detection of these conditions are crucial steps in providing appropriate interventions and supporting better recovery. With the increasing use of social media, many people have started sharing their thoughts, feelings, and experiences online. Social media provides an abundant platform for users to express themselves and interact with others. Posts on social media often reflect individuals' emotional states. Therefore, analyzing the content of these posts can provide valuable insights for monitoring and early detection of depressive symptoms. Machine learning has been widely used for automated text mining and classification tasks. A classification method that can be used to classify social media posts into depression and normal classes is the support vector machine. Based on the testing results of the Support Vector Machine algorithm in classifying posts on social media, the highest accuracy value obtained was 95.5% using a parameter value of C equal to 0.25. The Precision, recall, and F-1 score values were 96%.

Keywords: *Mental health issues, Depression, Support Vector Machine, Classification*

1. Pendahuluan

Masalah kesehatan mental, seperti depresi, memiliki dampak yang signifikan pada individu dan masyarakat. Dimana gangguan kesehatan mental adalah situasi di mana seseorang menghadapi kesulitan dalam beradaptasi dengan lingkungan sekitarnya dan mengalami kesulitan dalam menyelesaikan masalah, sehingga menyebabkan tingkat stres yang berlebihan [1]. Depresi adalah gangguan kesehatan mental yang menjadi alasan utama bunuh diri global. Menurut survei perhitungan beban penyakit di Indonesia pada tahun 2017, terdapat prevalensi gangguan kesehatan mental sebesar 13,4%, dimana depresi secara khusus menempati posisi teratas sebagai gangguan kesehatan mental selama tiga dekade terakhir, yaitu dari tahun 1990 hingga 2017 [2]. Identifikasi dan deteksi dini kondisi-kondisi ini merupakan langkah penting dalam memberikan intervensi yang tepat dan mendukung pemulihan yang lebih baik. Seiring dengan meningkatnya penggunaan media sosial, banyak orang mulai berbagi pikiran, perasaan, dan pengalaman mereka secara online. Menurut temuan survei "We Are Social" pada tahun 2017, jumlah pengguna media sosial di Indonesia mencapai 106 juta dari total populasi masyarakat yang berjumlah 262 juta [3]. Media sosial menyediakan platform yang melimpah bagi pengguna untuk mengekspresikan diri dan berinteraksi dengan orang lain. Postingan pada media sosial sering kali mencerminkan kondisi emosional individu. Oleh karena itu, analisis konten dari postingan tersebut dapat memberikan wawasan yang berharga dalam pemantauan dan klasifikasi deteksi dini gejala depresi.

Untuk mempermudah dalam klasifikasi deteksi depresi tentunya diperlukan teknik atau metode untuk dapat mengelompokkan dengan sebuah perhitungan yaitu dengan menggunakan metode text mining yaitu dengan klasifikasi. Klasifikasi digunakan karena klasifikasi adalah sebuah metode yang melakukan proses dalam mendapatkan suatu pola untuk menyatakan suatu teks

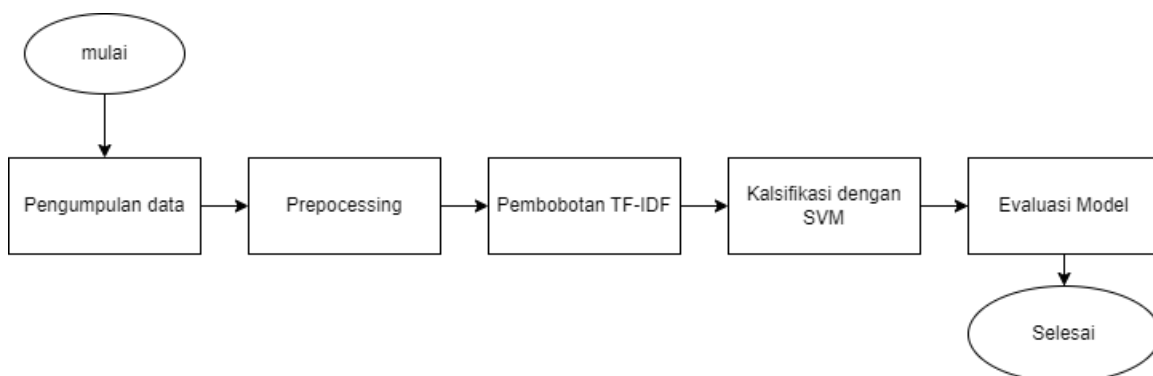
tersebut masuk pada kelompok tertentu yang telah ditentukan [4]. Salah satu algoritma yang dapat digunakan dalam *text mining* khususnya pada klasifikasi adalah *Support Vector Machine* (SVM). SVM merupakan suatu algoritma yang memiliki prinsip mencari *hyperplane* yang memiliki margin terbesar. *Hyperplane* yaitu suatu garis yang memisahkan data antar kelas atau kategori. Sedangkan margin merupakan jarak antara *hyperplane* dengan data terdekat yang berada pada masing-masing kelas. Data yang paling dekat dengan *hyperplane* disebut support vector [5]. Metode SVM ini memiliki tujuan utama yaitu untuk membangun OSH (Optimal Separating *Hyperplane*), yang membuat fungsi pemisahan optimum yang dapat digunakan untuk klasifikasi.

Pada penelitian sebelumnya yang terkait klasifikasi mengenai deteksi depresi yaitu penelitian dilakukan oleh Andre Budiman dkk yaitu melakukan klasifikasi konten Twitter dengan indikasi depresi dengan menggunakan metode *Multinomial Naïve Bayes* (MNB) dan *Complement Naïve Bayes* (CNB). Dari hasil eksperimen yang dilakukan, kombinasi metode TF-IDF dan MNB berhasil mencapai tingkat F-score sebesar 91,30%. Sementara itu, gabungan metode TF-IDF dan CNB berhasil mencapai tingkat performa sebesar 91,98% [6]. Selanjutnya penelitian yang dilakukan oleh Arianti Primadhani dkk yaitu melakukan analisis sentimen deteksi depresi pada pengguna media sosial Twitter dengan menggunakan metode KNN dengan menggunakan *confusion matrix* hasil akurasi yang didapatkan sebesar 78.18% [7].

Berdasarkan dari permasalahan yang ada dan penelitian-penelitian terdahulu yang terkait yang menjadi dasar dalam melakukan penelitian, maka penulis tertarik untuk melakukan penelitian mengenai klasifikasi deteksi depresi dari postingan pada media sosial dengan menggunakan algoritma *Support Vector Machine*. Penelitian ini bertujuan untuk mempelajari postingan media sosial yang terindikasi mengalami gangguan depresi atau normal. Dari metode yang digunakan untuk penelitian diatas diperlukan data teks berupa cuitan dari postingan media sosial yang diperoleh datanya melalui website Kaggle. Dengan menerapkan metode *Support Vector Machine*, diharapkan mampu menghasilkan akurasi yang baik dan luaran yang dihasilkan mampu digunakan untuk mengklasifikasikan deteksi depresi dari postingan pada media sosial dengan benar.

2. Metode Penelitian

Bagian ini akan menggambarkan secara umum langkah-langkah yang akan dilakukan dalam penelitian yang dilakukan oleh peneliti. Penelitian akan dimulai dengan mengumpulkan data teks berupa postingan pada media sosial. Setelah itu, akan dilakukan tahap preprocessing untuk mempersiapkan data tersebut. Selanjutnya, data akan diolah dengan menggunakan metode pembobotan kata menggunakan Term Frequency Inverse Document Frequency (TF-IDF). Setelah mendapatkan hasil pembobotan kata, langkah selanjutnya adalah melakukan proses klasifikasi dengan menggunakan metode Support Vector Machine. Pada tahap akhir penelitian, dilakukan pengujian dan evaluasi terhadap kinerja metode yang digunakan. Berikut adalah alur metodologi penelitian yang akan dilakukan oleh peneliti.



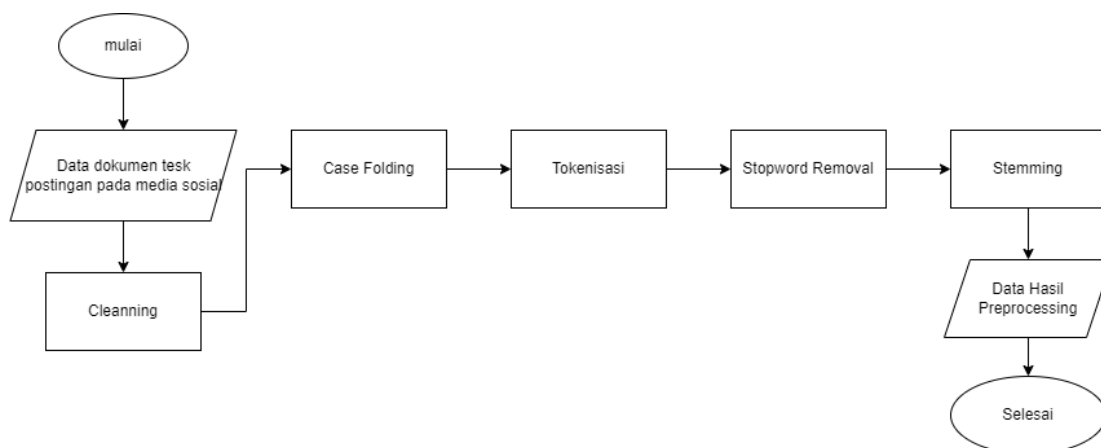
Gambar 1. Alur Metode Pnelitian

2.1 Pengumpulan Data

Pada proses pengumpulan data, dimana data yang akan digunakan berupa teks postingan pada sosial media yang didapatkan dari situs kaggle. Pada data tersebut terdapat lima variabel yaitu text, category, Age, Gender, Age Category. Namun variabel yang digunakan hanya variabel text dan Category. Data yang digunakan berjumlah 2454 record yang terdiri dari 2 kelas yaitu kelas 1 (Depresi) dan 0 (Normal). Distribusi antara kelas 1 (Depresi) dan 0 (Normal) masing-masing sebanyak 1227 record. Dari data tersebut digunakan data training dan data testing dengan perbandingan 80%: 20% atau sebanyak 1963 record untuk data *training* dan 491 record untuk data *testing*.

2.2 Text Preprocessing

Text Preprocessing merupakan tahap awal dalam membangun sebuah model *machine learning* dalam *text mining*. Pada langkah ini, dilakukan pra-pemrosesan data teks yang telah dikumpulkan sebelumnya. *Preprocessing* teks adalah suatu proses untuk mengubah data teks yang tidak terstruktur menjadi data yang terstruktur, atau dengan kata lain, mengubah teks menjadi indeks kata sesuai kebutuhan [8]. Tujuan dari proses ini adalah untuk mempersiapkan teks agar siap digunakan dan diolah. Pra-pemrosesan melibatkan serangkaian langkah, meliputi *cleaning*, *tokenization*, *case folding*, *stopwords removal*, dan *stemming*. Diharapkan dengan *preprocessing text* dapat mengurangi informasi yang tidak relevan atau tidak berarti dalam dokumen tersebut dengan menghilangkan kata atau teks yang tidak perlu. Semua langkah ini bertujuan untuk mempermudah proses selanjutnya. Ilustrasi alur dari tahap pra-pemrosesan dapat dilihat pada Gambar 2.



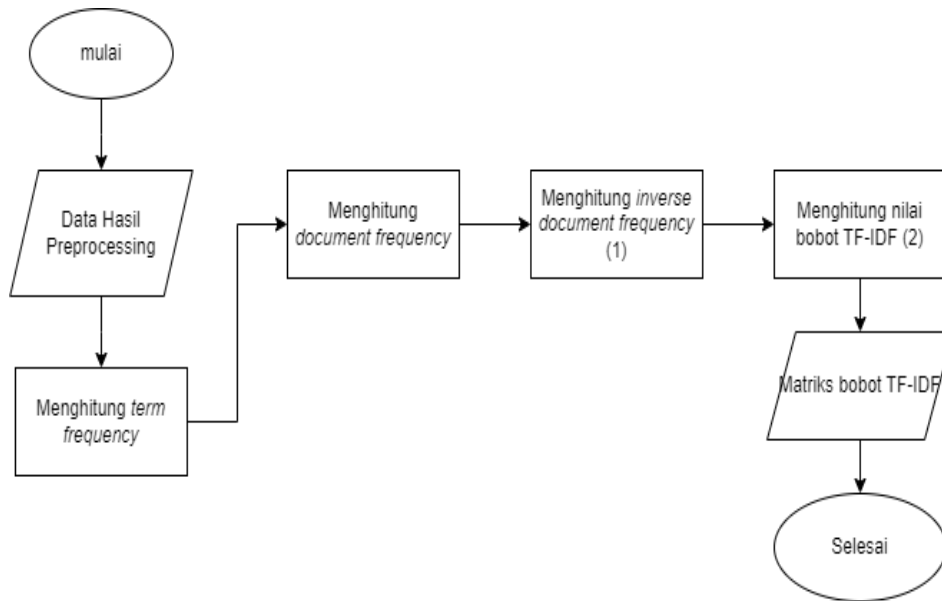
Gambar 2. Alur *Text Preprocessing*

Berdasarkan Gambar 2, tahap pra-pemrosesan terdiri dari beberapa langkah. Tahap pertama data dokumen teks postingan pada media sosial dilakukan *cleaning*, yang bertujuan untuk menghilangkan karakter selain huruf seperti tanda baca, emotikon, dan angka. Tahap kedua adalah *case folding*, yang mengubah semua kata menjadi huruf kecil. Tahap ketiga adalah *tokenization*, yang membagi kalimat menjadi token atau kata-kata tunggal berdasarkan tanda spasi. Tahap keempat adalah *stopword removal*, yaitu menghilangkan kata-kata *stopword* yang dianggap tidak memiliki makna. Terakhir, tahap kelima adalah *stemming*, yang berfungsi untuk mengubah kata-kata ke dalam bentuk dasarnya.

2.3 Ekstraksi Fitur TF-IDF

Setelah melalui tahap *preprocessing*, data akan masuk ke tahap pembobotan atau ekstraksi fitur. Dalam pengolahan teks, perlu dilakukan ekstraksi kata-kata menjadi bentuk numerik karena komputer pada dasarnya hanya dapat memproses data dalam bentuk numerik. Ekstraksi fitur digunakan untuk mengungkapkan informasi yang berpotensi dan mewakili kata-kata sebagai vektor fitur. Vektor ini kemudian digunakan sebagai input untuk metode klasifikasi pada tahap

selanjutnya. Salah satu teknik ekstraksi fitur yang umum digunakan adalah menggunakan metode TF-IDF (*Term Frequency-Inverse Document Frequency*). TF-IDF adalah metode pembobotan yang digunakan untuk mengukur pentingnya suatu kata dalam sebuah dokumen atau korpus. Metode ini menggunakan term atau kata-kata yang telah melalui proses preprocessing sebagai inputnya. Dengan menggunakan TF-IDF, kita dapat menentukan bobot atau nilai penting dari setiap kata dalam dokumen berdasarkan frekuensi kemunculannya dalam dokumen tersebut dan dalam keseluruhan korpus [9]. Hasil akhir dari ekstraksi fitur direpresentasikan dalam bentuk matriks yang berisi kata-kata unik dan nilai-nilai fitur TF-IDF dari setiap kata dalam semua data review. Berikut merupakan tahapan dari ekstraksi fitur.



Gambar 3. Alur Ekstraksi Fitur TF-IDF

Berdasarkan gambar 3 tersebut, proses pembobotan TF-IDF dimulai dengan menghitung frekuensi kata atau *term frequency* dalam dokumen ($tf_{i,j}$), kemudian menghitung frekuensi dokumen yang mengandung kata tersebut atau *document frequency* (df), dan dilanjutkan dengan perhitungan bobot inverse document frequency dengan menggunakan persamaan (1). Terakhir menghitung bobot TF-IDF dengan menggunakan persamaan (2).

$$idf_i = \log\left(\frac{N}{df_i}\right) \quad (1)$$

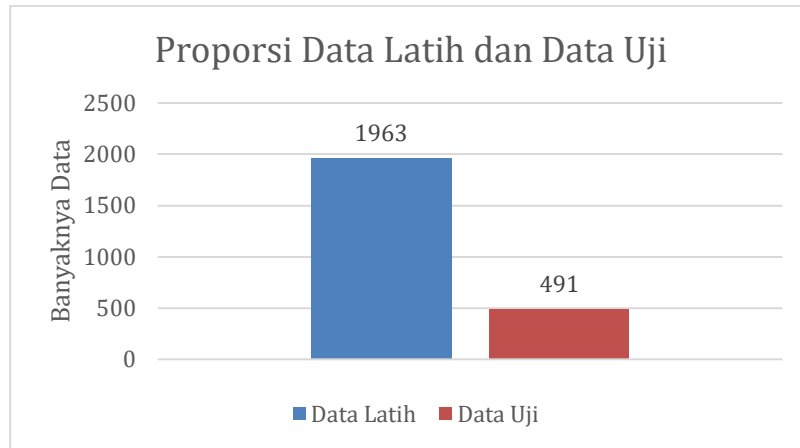
$$w_{i,j} = tf_{i,j} \times idf_i \quad (2)$$

Keterangan:

- N = jumlah dokumen secara keseluruhan
- $w_{i,j}$ = bobot *term* t terhadap dokumen d
- $tf_{i,j}$ = frekuensi *term* i pada dokumen j
- idf_i = nilai bobot IDF *term* i

2.4 Pemisahan Data

Langkah berikutnya adalah memisahkan data menjadi data latih dan data uji. Dalam penelitian ini, digunakan 80% data sebagai data latih dan 20% data sebagai data uji dari total 2454 record.



Gambar 4. Proporsi Data Latih dan Data Uji

Berdasarkan gambar 4, diperoleh 1963 data sebagai data latih dan 491 data sebagai data uji. Proses pemisahan data latih dan data uji dilakukan secara acak untuk menjaga proporsi yang seimbang antara kelas-kelas tersebut.

2.5 Klasifikasi *Support Vector Machine*

Support Vector Machine (SVM) adalah salah satu metode klasifikasi dalam pembelajaran mesin (*supervised learning*) yang menggunakan model hasil pelatihan untuk memprediksi kelas [10]. SVM bekerja dengan mencari *hyperplane* (bidang pembatas) yang memiliki margin terbesar. *Hyperplane* ini adalah garis yang memisahkan data antara kelas atau kategori. Margin adalah jarak antara *hyperplane* dengan data terdekat dari setiap kelas, dan data yang paling dekat dengan *hyperplane* disebut support vector. SVM melakukan klasifikasi dengan mencari *hyperplane* atau kernel maksimal yang dapat memisahkan dua kelas. Dalam konteks ini, kelas yang dipisahkan adalah sentimen positif (diberi label +1) dan sentimen negatif (diberi label -1). SVM menggunakan bidang pembatas paralel atau kernel untuk memisahkan kedua kelas tersebut. Berikut adalah langkah-langkah perhitungan yang terlibat dalam SVM [11]:

1. Diasumsikan bahwa kedua kelas -1 dan +1 dapat dipisahkan secara sempurna oleh *hyperplane* atau bidang pembatas berdimensi d , yang didefinisikan.

$$w \cdot x + b = 0 \tag{3}$$

2. Pattern x_i yang termasuk dalam kelas -1 (kelas negatif) dapat dirumuskan sebagai pattern yang memenuhi persamaan tertentu.

$$w \cdot x_i + b \leq -1 \tag{4}$$

3. Pattern x_i yang termasuk dalam kelas +1 (kelas positif) dapat dirumuskan sebagai pattern yang memenuhi persamaan tertentu.

$$w \cdot x_i + b \geq 1 \tag{5}$$

4. Fungsi digunakan untuk mencari nilai Lagrange Multiplier α dengan syarat tertentu.

$$\max_{\alpha} L_D = \sum_{i=1}^n \alpha_i - \frac{1}{2} \sum_{i,j=1}^n \alpha_i \alpha_j y_i y_j K(x_i, x_j) \tag{6}$$

$$\text{Syarat: } \alpha_i \geq 0 (i = 1, 2, \dots, n) \text{ dan } \sum_{i=1}^n \alpha_i y_i = 0$$

Setelah nilai dari fungsi di atas ditemukan, langkah selanjutnya adalah mencari nilai w , nilai bias, dan fungsi keputusan klasifikasi $\text{sign}(f(x))$:

5. Persamaan digunakan untuk mencari nilai w.

$$W = \sum_{i=1}^n \alpha_i y_i x_i \quad (7)$$

6. Persamaan digunakan untuk mencari nilai bias.

$$b = -\frac{1}{2}(x_i^+ \cdot W + x_i^- \cdot W) \quad (8)$$

7. Persamaan digunakan untuk mencari fungsi keputusan klasifikasi $\text{sign}(f(x))$.

$$f(x) = \text{sign}[\sum_{i,j=1}^n \alpha_i y_i K(x, x_i) + b] \quad (9)$$

Fungsi $\text{sign}()$ adalah fungsi normalisasi yang memberikan nilai 1 (kelas positif) jika nilai x dalam fungsi tersebut lebih dari 0, dan memberikan nilai -1 (kelas negatif) jika nilai x dalam fungsi tersebut kurang dari 0.

2.6 Pengujian dan Evaluasi

Evaluasi dilakukan untuk mengukur performa dari model yang telah dibangun. Performa ini dapat diukur menggunakan tabel *confusion matrix*. *Confusion matrix* adalah sebuah tabel yang umumnya digunakan untuk memvisualisasikan kinerja model klasifikasi pada sejumlah data uji di mana nilai sebenarnya sudah diketahui. Berikut ini adalah contoh tabel *confusion matrix* untuk sebuah model klasifikasi dengan dua kelas.

Tabel 1. Table *Confusion matrix*

	<i>Classifier positive</i>	<i>classifier negative</i>
<i>Actual positive</i>	TP	FN
<i>Actual negative</i>	FP	TN

Berdasarkan Tabel 1 *confusion matrix*, kita dapat mengukur performa *Precision*, *recall*, *F-1 score*, dan akurasi. *Precision* yaitu mengukur sejauh mana model dapat mengidentifikasi secara akurat dokumen-dokumen yang relevan dari dokumen yang dianggap relevan oleh model [12], dan persamaannya menjadi:

$$\text{Precision} = \frac{TP}{(TP+FP)} \quad (10)$$

Recall yaitu mengukur performa dokumen yang relevan dan bernilai positif dari seluruh dokumen yang benar, dan persamaannya menjadi:

$$\text{Recall} = \frac{TP}{(TP+FN)} \quad (11)$$

F-1 score mengukur gabungan dari *Precision* dan *recall*, yang dimana menggabungkan kedua metrik ini menjadi satu angka untuk memberikan gambaran keseluruhan tentang performa model, dan persamaannya menjadi:

$$\text{F-1 score} = 2 \times \frac{(\text{Precision} \times \text{recall})}{(\text{Precision} + \text{recall})} \quad (12)$$

Akurasi mengukur sejauh mana model dapat mengklasifikasikan dengan benar seluruh dokumen, baik relevan maupun tidak relevan. dan persamaannya menjadi:

$$\text{Akurasi} = \frac{(TP + TN)}{(TP + TN + FP + FN)} \quad (13)$$

3. Hasil dan Pembahasan

3.1. Hasil *Preprocessing*

Hasil dari seluruh proses preprocessing pada dataset disajikan dalam bentuk tabel yang dapat ditemukan pada Tabel 2. Di bawah ini adalah beberapa contoh kalimat yang diambil dari dataset.

Tabel 2. Hasil *Preprocessing*

Tahap Preprocessing Teks Postingan Media Sosial	
Tanpa Preprocessing	I hate being alive when I feel so dead inside.
Cleanning	I hate being alive when I feel so dead inside
Case Folding	I hate being alive when I feel so dead inside.
Tokenisasi	['i', 'hate', 'being', 'alive', 'when', 'i', 'feel', 'so', 'dead', 'inside']
Stopword Removal	['hate', 'alive', 'feel', 'dead', 'inside']
Stemming	hate alive feel dead inside

3.2. Hasil Pembobotan TF-IDF

Dalam penelitian ini, digunakan pembobotan TF-IDF untuk menghasilkan representasi vektor dari data teks. Vektorisasi adalah proses mengubah data teks menjadi data numerik, di mana setiap angka dalam vektor merepresentasikan kemunculan kata dalam dokumen. Pembobotan ini penting karena komputer hanya dapat memahami dan memproses data dalam bentuk numerik.

3.3. Hasil Klasifikasi SVM

Pada klasifikasi menggunakan SVM dengan kernel linear, dilakukan percobaan dengan beberapa nilai parameter C yang berbeda. Nilai C digunakan untuk mengontrol *trade-off* antara margin dan jumlah kesalahan klasifikasi. Berikut adalah hasil dari percobaan tersebut:

Tabel 3. Percobaan dengan Beberapa Nilai Parameter C

C	Akurasi
0.01	90,8%
0.05	94,3%
0.25	95,9%
0.5	95,5%
1	95,1%

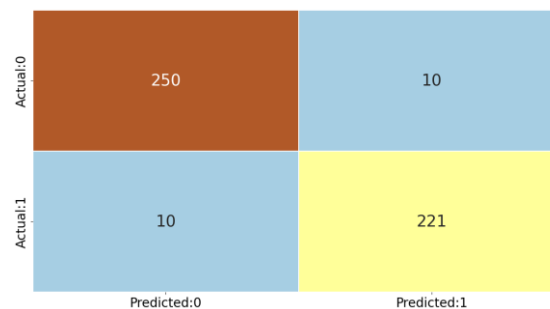
Berdasarkan tabel 3 tersebut, hasil pengujian menunjukkan bahwa penggunaan parameter C mempengaruhi akurasi klasifikasi. Ketika menggunakan parameter C = 0.01, diperoleh akurasi sebesar 90,8%, kemudian, dengan nilai C = 0.05, akurasi meningkat menjadi 94,3%, menunjukkan peningkatan performa. Penggunaan parameter C = 0.25 menghasilkan akurasi sebesar 95,9%, menunjukkan peningkatan signifikan dalam performa klasifikasi. Meskipun akurasi turun sedikit, dengan C = 0.5 dengan akurasi 95,5% dan C = 1, yaitu 95,1%, model SVM tetap memberikan performa yang baik. Nilai C yang lebih tinggi cenderung memberikan performa yang lebih baik, tetapi perlu diingat bahwa peningkatan yang signifikan juga bisa berdampak pada *overfitting* atau kompleksitas model yang lebih tinggi. Oleh karena itu, pemilihan nilai C yang optimal perlu mempertimbangkan *trade-off* antara akurasi dan kompleksitas model. Sehingga untuk nilai C yang digunakan dalam penelitian ini yaitu dengan parameter C = 0.25.

3.4. Evaluasi

Evaluasi dilakukan menggunakan *Confusion matrix* untuk mengukur akurasi yang diperoleh dari metode yang digunakan. Pada pemodelan menggunakan *Support Vector Machine* (SVM), diperoleh akurasi yang tinggi pada tahap pelatihan dan pengujian. Informasi evaluasi yang lebih rinci disajikan dalam Tabel 4 dan Gambar 5.

Tabel 4. Hasil Evaluasi

	<i>Precision</i>	<i>Recall</i>	<i>F-1 Score</i>	Akurasi
depresi	96%	96%	96%	96%
normal				95,9%



Gambar 5. *Confusion matrix* SVM

Dalam evaluasi yang menggunakan parameter $C = 0.25$, yang menghasilkan akurasi terbaik sebesar 96%, juga dilakukan perhitungan nilai *Precision*, *recall*, dan *F-1 score*. Hasil evaluasi menunjukkan bahwa model klasifikasi SVM memiliki kinerja yang sangat baik dalam mengklasifikasikan data. *Precision* yang mencapai 96% menunjukkan tingkat ketepatan yang tinggi dalam mengidentifikasi dokumen-dokumen yang relevan dan bernilai positif. *Recall* sebesar 96% menunjukkan bahwa model mampu mengenali sebagian besar dokumen yang sebenarnya relevan. Selain itu, nilai *F-1 score* sebesar 96% menunjukkan keseimbangan yang baik antara *Precision* dan *recall*. Dengan demikian, dapat disimpulkan bahwa penggunaan parameter $C = 0.25$ pada SVM memberikan kinerja yang optimal dalam mengklasifikasikan data, dengan akurasi, *Precision*, *recall*, dan *F-1 score* yang sangat memuaskan. Model ini dapat diandalkan dalam mengidentifikasi dokumen-dokumen yang relevan dan memberikan hasil prediksi yang akurat.

4. Kesimpulan

Berdasarkan penelitian ini, deteksi depresi dari postingan pada media sosial dilakukan dengan menggunakan metode klasifikasi *Support Vector Machine* (SVM) dengan kernel linear. Dataset terdiri dari dua kelas, yaitu depresi dan normal. Hasil evaluasi menunjukkan bahwa penggunaan SVM dengan nilai parameter C sebesar 0.25 menghasilkan akurasi terbaik sebesar 95.5%. Selain itu, *Precision*, *recall*, dan *F-1 score* juga mencapai nilai yang tinggi, yaitu sebesar 96%. Penelitian ini menunjukkan bahwa metode SVM dengan kernel linear sangat efektif dalam mengklasifikasikan postingan pada media sosial menjadi kategori depresi atau normal. Penggunaan nilai parameter C sebesar 0.25 memberikan kinerja optimal dengan tingkat akurasi yang tinggi. *Precision*, *recall*, dan *F-1 score* yang tinggi menunjukkan bahwa model ini mampu mengenali dengan baik postingan-postingan yang relevan dengan kondisi depresi dan normal.

Daftar Pustaka

- [1] Putri, A. W., Wibhawa, B. dan Gutama, A. S. 2015. Kesehatan Mental Masyarakat Indonesia (Pengetahuan, Dan Keterbukaan Masyarakat Terhadap Gangguan Kesehatan Mental). *Prosiding Penelitian dan Pengabdian Kepada Masyarakat*. 2, 2, 252–258. doi:

- 10.24198/jppm.v2i2.13535
- [2] Indrayani, Y.A., Wahyudi, T. 2019. Situasi Kesehatan Jiwa di Indonesia. InfoDatin Pusat Data dan Informasi Kementerian Kesehatan RI. 1-12
 - [3] J. Degenhard, "Forecast of the number of Twitter users in Indonesia from 2017 to 2026," Statista.com, 2020.
<https://www.statista.com/forecasts/1145550/twitter-users-in-indonesia>.
 - [4] Setio, P. B. N., Saputro, D. R. S., and Winarno. B., "Klasifikasi Dengan Pohon Keputusan Berbasis Algoritme C4.5," Prism. Pros. Semin. Nas. Mat., vol. 3, pp. 64–71, 2020.
 - [5] Irmanda, H.N. and Astriratma, R. (2020). Klasifikasi Jenis Pantun dengan Metode Support Vector Machines (SVM). Jurnal RESTI (Rekayasa Sistem dan Teknologi Informasi), 4(5), pp.915–922.
 - [6] A. Budiman, A. Suryadibrata, and J. Young, "Implementasi Algoritma Naïve Bayes untuk Klasifikasi Konten Twitter dengan Indikasi Depresi," *Jurnal Informatika: Jurnal Pengembangan IT*, vol. 6, no. 1, 2021.
 - [7] A.P.Tirtopangarsa and W. Maharani, "Sentiment Analysis of Depression Detection on Twitter Social Media Users Using the K-Nearest Neighbor Method," 2021. In *Seminar Nasional Informatika (SEMNASIF)* (Vol. 1, No. 1, pp. 247-258).
 - [8] E. Indrayuni. "Klasifikasi Text Mining Review Produk Kosmetik Untuk Teks Bahasa Indonesia Menggunakan Algoritma Naive Bayes." vol. VII, no. 1, pp. 29-36, 2019.
 - [9] P. M. Prihatini, "Implementasi Ekstraksi Fitur Pada Pengolahan Dokumen Berbahasa Indonesia," *Matrix J. Manaj. Teknol. dan Inform.*, vol. 6, no. 3, pp. 174–178, 2017.
 - [10] R. Maulana & S. Redjeki. (2017). Analisis Sentimen Pengguna Twitter Menggunakan Metode *Support Vector Machine* Berbasis Cloud Computing. *Jurnal TAM*, 6, 23-28.
 - [11] A. S. Nugroho, A. B. Wiarto, & D. Handoko. (2003). *Support Vector Machine*. *Proceesing Indones. Sci. Meetiting Cent. Japan*.
 - [12] Zizka, Jan., Darena, Frantisek., and Svoboda, Arnost. 2020. *Text Mining with Machine Learning Principles and Techniques*. CRC Press. New York.

Halaman ini sengaja dibiarkan kosong