PENGARUH KERNEL PADA SVM TERHADAP KLASIFIKASI SENTIMEN *TWEET* BAHASA INDONESIA

Titin Rahayu[1], Arif Bijaksana Putra Negara[2], Enda Esyudha Pratama[3]

[1,2,3]*Universitas Tanjungpura*
E-mail : [1]Titinrahayu244@student.untan.ac.id, [2]Arifbpn@informatika.untan.ac.id, [3]Enda@informatika.untan.ac.id

ABSTRACT

Sentiment analysis of Indonesian-language Twitter data presents challenges due to the short, unstructured nature of tweets, as well as the frequent use of informal language, abbreviations, and non-standard words. This study aims to analyze the effect of Support Vector Machine (SVM) kernel selection, Synthetic Minority Over-sampling Technique (SMOTE), and hyperparameter tuning using Grid Search on the performance of sentiment classification of Indonesian-language tweets. The dataset consisted of 10,820 tweets classified into three sentiment categories: positive, negative, and neutral. The research stages included text preprocessing, TF-IDF feature extraction, data splitting, SMOTE implementation, SVM modeling using Linear, Radial Basis Function (RBF), Sigmoid, and Polynomial kernels, and model evaluation using accuracy, precision, recall, and F1-score. A total of 16 scenarios were tested based on combinations of kernels, SMOTE, and Grid Search. The results showed that the RBF kernel achieved the best performance. The combination of RBF kernel, SMOTE, and Grid Search obtained an accuracy of 75.47%, precision of 75.95%, recall of 75.47%, and F1-score of 75.54%. Therefore, the combination of RBF kernel, SMOTE, and Grid Search was the best configuration for sentiment classification of Indonesian-language tweets in this dataset.

Keywords : *Sentiment Analysis, Twitter, Support Vector Machine, SMOTE, Grid Search, Kernel.*

ABSTRAK

Analisis sentimen pada *Twitter* berbahasa Indonesia memiliki tantangan karena karakteristik data yang singkat, tidak terstruktur, serta banyak menggunakan kata tidak baku dan bahasa informal. Penelitian ini bertujuan menganalisis pengaruh pemilihan *kernel* Support Vector Machine (SVM), penerapan Synthetic Minority Over-sampling Technique (SMOTE), dan *hyperparameter tuning* menggunakan *Grid Search* terhadap performa klasifikasi sentimen *tweet* berbahasa Indonesia. Dataset yang digunakan terdiri atas 10.820 *tweet* dengan tiga kelas sentimen, yaitu positif, negatif, dan netral. Tahapan penelitian meliputi *text preprocessing*, pembobotan TF-IDF, pembagian data, penerapan SMOTE, pemodelan SVM menggunakan *kernel linear*, RBF, *sigmoid*, dan *polynomial*, serta evaluasi menggunakan *accuracy*, *precision*, *recall*, dan *F1-score*. Pengujian dilakukan pada 16 skenario berdasarkan kombinasi *kernel*, SMOTE, dan *Grid Search*. Hasil penelitian menunjukkan bahwa *kernel RBF* menghasilkan performa terbaik. Skenario dengan kombinasi RBF, SMOTE, dan *Grid Search* memperoleh *accuracy* sebesar 75,47%, *precision* 75,95%, *recall* 75,47%, dan *F1-score* 75,54%. Dengan demikian, kombinasi pemilihan *kernel* RBF, SMOTE, dan *Grid Search* merupakan konfigurasi terbaik untuk klasifikasi sentimen *tweet* berbahasa Indonesia pada dataset penelitian ini.

Kata Kunci : *Analisis Sentimen, Twitter, Support Vector Machine, SMOTE, Grid Search, Kernel.*

PENDAHULUAN

Perkembangan media sosial menjadikan *Twitter* sebagai salah satu platform yang banyak digunakan untuk menyampaikan pendapat, ekspresi, dan reaksi terhadap berbagai topik maupun peristiwa secara *real-time*. Banyaknya *tweet* berbahasa Indonesia yang tersedia dapat dimanfaatkan untuk mengetahui kecenderungan sentimen masyarakat terhadap suatu topik atau entitas. Namun, klasifikasi sentimen pada teks berbahasa Indonesia memiliki tantangan berupa penggunaan kata singkatan, bahasa gaul (*slang*), kata asing, serta ambiguitas teks. Oleh karena itu, diperlukan tahapan analisis yang sistematis, meliputi *pre-processing*, *transformation*, *data splitting*, *modelling*, dan *evaluation* untuk mengelompokkan teks ke dalam sentimen positif, negatif, dan netral (Thalita Meisya Permata Aulia, 2021; Passonneau, 2011).

Salah satu algoritma yang banyak digunakan dalam klasifikasi teks adalah *Support Vector Machine* (SVM). SVM memiliki kemampuan generalisasi yang baik dan bekerja dengan membentuk *hyperplane* optimal untuk memisahkan data berdasarkan kelas. Pada data yang tidak dapat dipisahkan secara linear, SVM dapat menggunakan fungsi *kernel* untuk mentransformasikan data ke ruang fitur berdimensi lebih tinggi. Dalam klasifikasi dengan lebih dari dua kelas, SVM dapat dikembangkan menjadi *Multi-class SVM* dengan pendekatan *One Versus One* (OVO) dan *One Versus All* (OVA) (Thalita Meisya Permata Aulia, 2021; Meidianingsih & Meganingtyas, 2022). Pemilihan *kernel* menjadi salah satu faktor penting karena dapat memengaruhi kemampuan model dalam melakukan klasifikasi.

Selain pemilihan *kernel*, performa SVM juga dapat ditingkatkan melalui *hyperparameter tuning*. Salah satu metode yang dapat digunakan adalah *Grid Search*, yaitu metode pencarian kombinasi parameter untuk memperoleh konfigurasi model yang optimal. Penelitian Pratama dan Trilaksono (2015) menunjukkan bahwa penggunaan SVM pada klasifikasi *tweet* dapat memberikan hasil yang baik, dengan parameter C dan γ sebagai parameter yang dioptimalkan. Penelitian lain yang membandingkan *kernel linear*, *sigmoid*, *RBF*, dan *polynomial* pada analisis sentimen vaksin COVID-19 memperoleh akurasi sebesar 0,87 pada *kernel linear* dan *sigmoid*, sedangkan *kernel polynomial* dan *RBF* memperoleh akurasi sebesar 0,86 (Thalita Meisya Permata Aulia, 2021).

Berdasarkan penelitian terdahulu tersebut, penelitian ini dilakukan untuk menganalisis pengaruh pemilihan *kernel* pada SVM terhadap performa klasifikasi sentimen *tweet* berbahasa Indonesia. Penelitian membandingkan empat jenis *kernel*, yaitu *linear*, *RBF*, *sigmoid*, dan *polynomial*, serta membandingkan model dengan nilai *default* dan model yang telah melalui *hyperparameter tuning* menggunakan *Grid Search*. Selain itu, metode *Synthetic Minority Over-sampling Technique* (SMOTE) diterapkan untuk menangani ketidakseimbangan kelas pada *dataset*. *Dataset* yang digunakan merupakan modifikasi dari dataset Indonesia untuk analisis sentimen yang dikembangkan oleh Ferdiana et al. (2019), dengan tiga kelas sentimen, yaitu positif, negatif, dan netral. Performa model selanjutnya dievaluasi untuk mengetahui konfigurasi yang memberikan hasil klasifikasi terbaik.

METODE

Data Penelitian

Penelitian ini menggunakan *Dataset Indonesia untuk Analisis Sentimen* yang dimodifikasi dari penelitian Ferdiana et al. (2019). Dataset berupa teks *tweet* berbahasa Indonesia yang terdiri

atas 10.820 data dengan tiga kelas sentimen, yaitu positif sebanyak 3.228 data, negatif sebanyak 4.036 data, dan netral sebanyak 3.556 data. Dataset digunakan untuk proses pelatihan dan pengujian model klasifikasi sentimen.

Tahapan Penelitian

Tahapan penelitian meliputi *data collection*, *text preprocessing*, vektorisasi data, *dataset splitting*, *dataset balancing*, *modelling*, *validation*, dan *evaluation*. Pada tahap *text preprocessing*, dilakukan *data cleaning* untuk menghilangkan data duplikat dan *case folding* untuk mengubah seluruh huruf menjadi huruf kecil. Selanjutnya, data teks diubah menjadi bentuk numerik menggunakan metode *Term Frequency-Inverse Document Frequency* (TF-IDF).

Dataset kemudian dibagi menjadi data latih sebesar 90% dan data uji sebesar 10%. Untuk mengatasi ketidakseimbangan jumlah data antarkelas, diterapkan metode *Synthetic Minority Over-sampling Technique* (SMOTE) pada data latih. Selanjutnya, dilakukan pemodelan menggunakan algoritma *Support Vector Machine* (SVM) dengan empat jenis *kernel*, yaitu *linear*, *RBF*, *sigmoid*, dan *polynomial*.

Pengujian dilakukan melalui 16 skenario yang merupakan kombinasi dari empat *kernel*, penerapan dan tanpa penerapan SMOTE, serta penggunaan *hyperparameter tuning* dan nilai *default*. *Hyperparameter tuning* dilakukan menggunakan metode *Grid Search* untuk memperoleh kombinasi parameter terbaik pada model SVM.

Tabel 1.

Kernel	SMOTE	Grid Search	Jumlah Skenario
Linear	- / ✓	- / ✓	4
RBF	- / ✓	- / ✓	4
Sigmoid	- / ✓	- / ✓	4
Polynomial	- / ✓	- / ✓	4
Total			**16**

Validasi model dilakukan menggunakan *10-fold cross validation* untuk menilai konsistensi model dan mengurangi kemungkinan *overfitting*. Pada metode ini, data latih dibagi menjadi 10 bagian dan proses pelatihan serta validasi dilakukan secara bergantian pada setiap bagian.

Evaluasi performa model dilakukan menggunakan *confusion matrix* dengan parameter *accuracy*, *precision*, *recall*, dan *F1-score*. Hasil evaluasi dari seluruh skenario dibandingkan untuk mengetahui pengaruh pemilihan *kernel*, SMOTE, dan *hyperparameter tuning* terhadap performa klasifikasi sentimen *tweet* berbahasa Indonesia.

Seluruh proses penelitian diimplementasikan menggunakan bahasa pemrograman Python dengan bantuan *library Scikit-learn* untuk pemodelan SVM dan *Grid Search*, serta *Imbalanced-learn* untuk penerapan SMOTE.

HASIL PENELITIAN DAN PEMBAHASAN

Hasil Pengolahan Data

Penelitian menggunakan *Indonesian General Sentiment Analysis Dataset* yang terdiri atas 10.820 *tweet* berbahasa Indonesia dengan tiga kelas sentimen, yaitu negatif sebanyak 4.036 data, positif sebanyak 3.228 data, dan netral sebanyak 3.556 data. Data tersebut selanjutnya melalui tahap *text preprocessing* sebelum digunakan dalam proses klasifikasi.

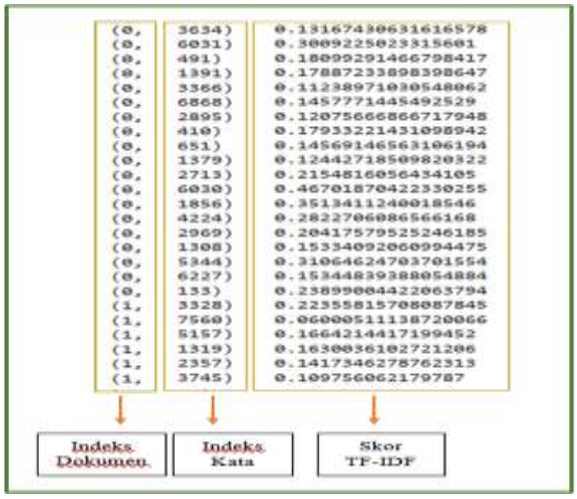
Tahap *text preprocessing* meliputi *case folding*, *cleaning*, *normalization*, *tokenizing*, *stemming*, dan *rejoin*. Contoh hasil setiap tahapan pengolahan data ditunjukkan pada Gambar 1.

	Tweet	Label	Case Folding	Data Cleaning	Normalisasi	Tokenisasi	Stemming	Rejoin
1	aku selalu per...	netral	aku selalu p...	aku selalu p...	aku selalu p...	['aku', 'selalu'...	['aku', 'selalu'...	aku selalu p...
2	Laki laki sejati...	positif	laki laki seja...	laki laki sejat...	laki laki sejat...	['laki', 'laki', 's...	['laki', 'laki', '...	laki laki seja...
3	aku personal ...	netral	aku persona...	aku personal...	aku personal...	['aku', 'perso...	['aku', 'perso...	aku person...
4	sheilaon7 sa...	positif	sheilaon7 sa...	sheilaon sa...	sheilaon sa...	['sheilaon', 's...	['sheilaon', 's...	sheilaon sa...
5	ya allah jantu...	negatif	ya allah jant...	ya allah jant...	ya allah jant...	['ya', 'allah', ']...	['ya', 'allah', ']...	ya allah jant...
6	tapi gue gak p...	negatif	tapi gue gak...	tapi gue gak ...	tapi saya en...	['tapi', 'saya', ...	['tapi', 'saya',...	tapi saya en...
7	hai kak stiky s...	netral	hai kak stiky...	hai kak stiky ...	hai kakak sti...	['hai', 'kakak',...	['hai', 'kakak',...	hai kakak st...
8	terlihat enak d...	positif	terlihat enak...	terlihat enak ...	terlihat enak ...	['terlihat', 'en...	['lihat', 'enak'...	lihat enak d...
9	ayo kita otw b...	netral	ayo kita otw ...	ayo kita otw ...	ayo kita jala...	['ayo', 'kita', ']...	['ayo', 'kita', ']...	ayo kita jala...
10	sudah hilangk...	positif	sudah hilan...	sudah hilang...	sudah hilang...	['sudah', 'hila...	['sudah', 'hila...	sudah hilan...
11	sedikit Banya...	netral	sedikit bany...	sedikit bany...	sedikit bany...	['sedikit', 'ban...	['sedikit', 'ba...	sedikit bany...
12	Klo tidak ma...	positif	klo tidak ma...	klo tidak ma...	kalau tidak ...	['kalau', 'tidak...	['kalau', 'tida...	kalau tidak ...
13	mau ganti pre...	negatif	mau ganti pr...	mau ganti pr...	mau ganti pr...	['mau', 'ganti'...	['mau', 'ganti'...	mau ganti p...
14	terkadang ka...	negatif	terkadang k...	terkadang ka...	terkadang k...	['terkadang', '...	['terkadang', ...	terkadang k...
15	tidak mau mel...	netral	tidak mau m...	tidak mau m...	tidak mau m...	['tidak', 'mau'...	['tidak', 'mau'...	tidak mau le...

Gambar 1. Hasil *Text Preprocessing*

Berdasarkan Gambar 1, data awal yang masih mengandung huruf kapital, tanda baca, serta kata tidak baku diubah menjadi bentuk yang lebih terstruktur. Pada tahap *normalization*, kata tidak baku seperti "ga" diubah menjadi "tidak", sedangkan pada tahap *stemming* kata diubah menjadi bentuk dasarnya. Setelah seluruh tahapan selesai, diperoleh 7.617 *vocabulary* dengan total 148.961 token.

Setelah *text preprocessing*, data ditransformasikan menjadi representasi numerik menggunakan metode TF-IDF. Proses ini dilakukan untuk memberikan bobot pada setiap kata berdasarkan tingkat kepentingannya dalam dokumen. Hasil transformasi TF-IDF ditunjukkan pada Gambar 2.



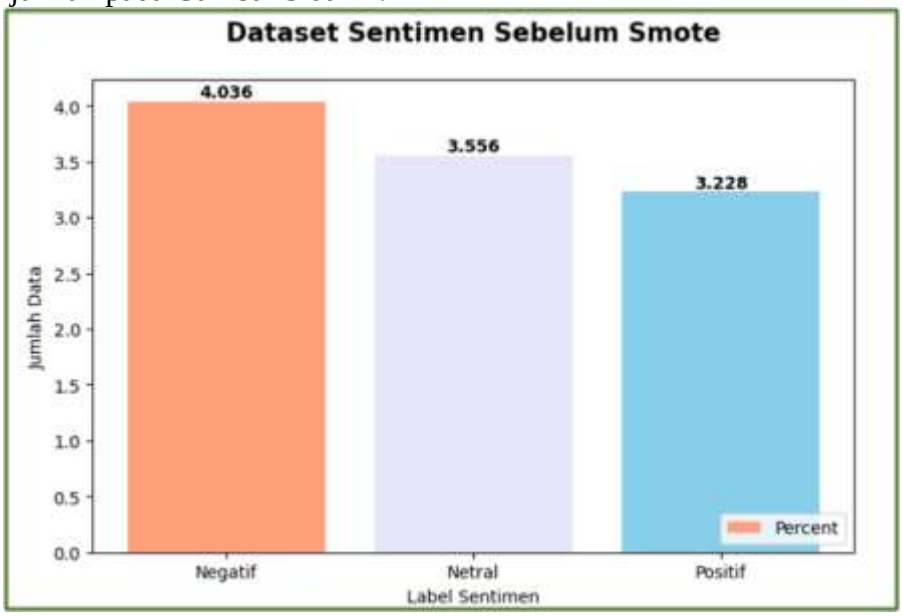
Gambar 2. Hasil Transformasi TF-IDF

Gambar 2 menunjukkan bahwa data teks telah berhasil diubah menjadi matriks numerik yang dapat digunakan sebagai masukan pada algoritma SVM. Selanjutnya, data dibagi menjadi 90% data latih dan 10% data uji.

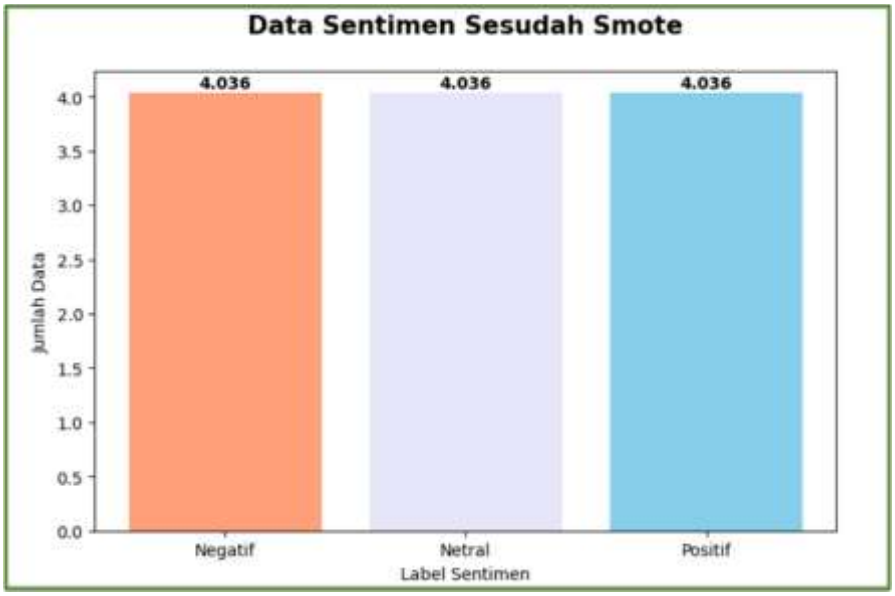
Penerapan SMOTE

Distribusi data menunjukkan bahwa kelas negatif merupakan kelas mayoritas, sedangkan kelas positif dan netral merupakan kelas minoritas. Untuk mengatasi ketidakseimbangan tersebut, penelitian menerapkan *Synthetic Minority Over-sampling Technique* (SMOTE).

Sebelum penerapan SMOTE, data latih terdiri atas 9.738 data, sedangkan setelah SMOTE jumlah data latih meningkat menjadi 10.897 data. Perubahan distribusi kelas sebelum dan sesudah SMOTE ditunjukkan pada Gambar 3 dan 4.



Gambar 3. Distribusi Data Sebelum SMOTE



Gambar 4. Distribusi Data Sesudah SMOTE

Berdasarkan Gambar 3 dan 4, penerapan SMOTE meningkatkan jumlah sampel pada kelas minoritas sehingga distribusi data menjadi lebih seimbang. Data hasil SMOTE kemudian digunakan pada skenario pengujian yang menerapkan SMOTE.

Hasil *Hyperparameter Tuning*

Pengujian dilakukan menggunakan empat *kernel* SVM, yaitu *linear*, *RBF*, *sigmoid*, dan *polynomial*. Untuk memperoleh kombinasi parameter yang optimal, dilakukan *hyperparameter tuning* menggunakan *Grid Search* dengan parameter C dan gamma.

Proses pencarian parameter terbaik pada masing-masing *kernel* ditunjukkan pada Tabel 1.

Tabel 1. Hasil *Grid Search Hyperparameter Tuning*

Skenario	Kernel	*Grid Search*	*Hyperparameter*	
			C	Gamma
3	Linear	v	0.615848211066026	0.0001
7	RBF	v	1.623776739188721	0.615848211066026
11	Sigmoid	v	5.0	0.21000000000000002
15	Polynomial	v	78.47599703514607	10000

Hasil pencarian menunjukkan bahwa setiap *kernel* memiliki kombinasi parameter optimal yang berbeda. Parameter tersebut selanjutnya digunakan pada skenario yang menerapkan *Grid Search*. Ringkasan parameter terbaik disajikan pada Tabel 2.

Tabel 2. Parameter Terbaik Hasil *Grid Search*

Kernel	Skenario	C	Gamma
Linear	3	0,6158	0,0001
Linear + SMOTE	4	2,3	0,00001
RBF	7	1,6238	0,6158
RBF + SMOTE	8	4,2813	1,6238
Sigmoid	11	5,0	0,21
Sigmoid + SMOTE	12	10.000	0,0001
Polynomial	15	78,4760	10.000
Polynomial + SMOTE	16	0,0002	18,0

Hasil Klasifikasi SVM

Sebanyak 16 skenario pengujian dilakukan berdasarkan kombinasi empat *kernel*, penggunaan SMOTE, dan *Grid Search*. Evaluasi model dilakukan menggunakan *10-fold cross validation* dan pengukuran *accuracy*, *precision*, *recall*, serta *F1-score*.

Hasil *accuracy* dari seluruh skenario ditunjukkan pada Gambar 5.



Gambar 5. Perbandingan *Accuracy* 16 Skenario

Berdasarkan Gambar 5, skenario 8 menghasilkan *accuracy* tertinggi dibandingkan seluruh skenario lainnya. Skenario tersebut menggunakan *kernel RBF* dengan SMOTE dan *Grid Search*, dengan *accuracy* sebesar 75,47%.

Untuk melihat performa masing-masing skenario secara lebih rinci, hasil evaluasi disajikan pada Tabel 2.

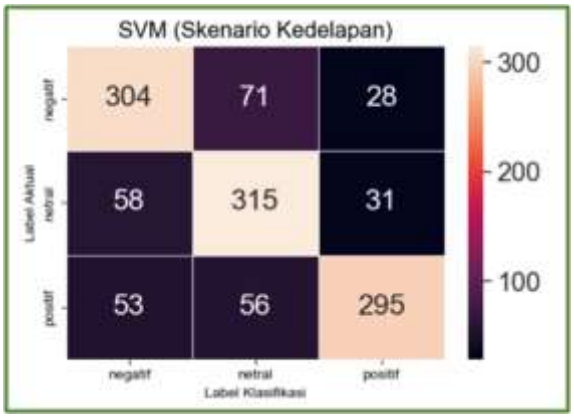
Tabel 2. Hasil Evaluasi 16 Skenario

Kernel	Skenario	SMOTE	*Grid Search*	*Accuracy*
Linear	1	–	–	70,70%
Linear	2	✓	–	72,09%
Linear	3	–	✓	70,89%
Linear	4	✓	✓	72,17%
RBF	5	–	–	71,63%
RBF	6	✓	–	74,40%
RBF	7	–	✓	72,18%
RBF	8	✓	✓	**75,47%**
Sigmoid	9	–	–	70,79%
Sigmoid	10	✓	–	70,11%
Sigmoid	11	–	✓	71,07%
Sigmoid	12	✓	✓	72,09%
Polynomial	13	–	–	62,01%
Polynomial	14	✓	–	72,83%
Polynomial	15	–	✓	63,49%
Polynomial	16	✓	✓	73,16%

Berdasarkan Tabel 2, *kernel RBF* menghasilkan performa rata-rata tertinggi dibandingkan *kernel* lainnya. Skenario 8 memberikan hasil terbaik dengan *accuracy* 75,47%. Pada skenario tersebut, nilai *precision* sebesar 75,95%, *recall* sebesar 75,47%, dan *F1-score* sebesar 75,54%.

Hasil *Confusion Matrix*

Untuk melihat kemampuan model dalam mengklasifikasikan data pada masing-masing kelas sentimen, digunakan *confusion matrix*. Visualisasi *confusion matrix* pada skenario terbaik ditampilkan pada Gambar 6.



Gambar 6. *Confusion Matrix* Skenario 8

Gambar 6 menunjukkan hasil klasifikasi skenario 8 pada data uji. Model mampu mengklasifikasikan sebagian besar data dengan benar pada kelas positif, negatif, dan netral. Dari 1.211 data uji, sebanyak 914 data berhasil diklasifikasikan dengan benar.

Pembahasan

Hasil penelitian menunjukkan bahwa tahapan *text preprocessing* berperan penting dalam mempersiapkan data *tweet* sebelum dilakukan klasifikasi. Tahapan *case folding*, *cleaning*, *normalization*, *tokenizing*, dan *stemming* menghasilkan data yang lebih terstruktur serta mengurangi kata tidak baku dan karakter yang tidak diperlukan. Khairunnisa et al. (2021) menyatakan bahwa *text preprocessing* dapat memengaruhi performa analisis sentimen karena kualitas data masukan berpengaruh terhadap proses klasifikasi. Normalisasi kata tidak baku juga diperlukan pada data *Twitter* berbahasa Indonesia karena penggunaan singkatan dan bahasa informal cukup tinggi (Rozi et al., 2019). Selain itu, stemming membantu mengubah kata menjadi bentuk dasarnya sehingga variasi kata dapat dikurangi dan proses klasifikasi menjadi lebih efektif (Utomo, 2013).

Transformasi data menggunakan TF-IDF menghasilkan representasi numerik yang digunakan sebagai masukan dalam proses klasifikasi SVM. TF-IDF memberikan bobot berdasarkan tingkat kepentingan suatu kata dalam dokumen sehingga karakteristik teks dapat direpresentasikan dalam bentuk numerik. Darwis et al. (2020) menunjukkan bahwa TF-IDF dapat digunakan untuk merepresentasikan teks *Twitter* berdasarkan bobot kata sebelum dilakukan proses analisis lebih lanjut. Dengan demikian, penggunaan TF-IDF dalam penelitian ini sesuai untuk mengubah data teks menjadi fitur yang dapat diproses oleh algoritma klasifikasi.

Hasil pengujian terhadap empat *kernel* menunjukkan bahwa *kernel RBF* memberikan performa rata-rata terbaik dengan *accuracy* sebesar 73,42%, diikuti *linear* sebesar 71,46%, *sigmoid* sebesar 71,02%, dan *polynomial* sebesar 67,87%. Perbedaan tersebut menunjukkan bahwa pemilihan *kernel* memengaruhi kemampuan SVM dalam membentuk batas pemisah antar kelas. SVM memiliki kemampuan dalam menangani klasifikasi data berdimensi tinggi, termasuk data teks yang direpresentasikan dalam bentuk fitur numerik (Jumeilah, 2017). Hasil penelitian ini juga sejalan dengan Thalita Meisya Permata Aulia (2021) yang menunjukkan adanya perbedaan performa pada penggunaan beberapa *kernel* SVM untuk analisis sentimen, sehingga pemilihan *kernel* perlu disesuaikan dengan karakteristik data.

Penerapan SMOTE menunjukkan pengaruh yang berbeda pada setiap *kernel*. Pada *kernel RBF*, penggunaan SMOTE meningkatkan *accuracy* dari 71,63% menjadi 74,40%. Peningkatan yang lebih besar terlihat pada *kernel polynomial*, yaitu dari 62,01% menjadi 72,83%. Hal tersebut menunjukkan bahwa ketidakseimbangan kelas dapat memengaruhi kemampuan model dalam mengenali kelas minoritas. SMOTE bekerja dengan membentuk sampel sintetis pada kelas minoritas sehingga distribusi data menjadi lebih seimbang. Sutoyo dan Fadlurrahman (2020) juga menunjukkan bahwa penerapan SMOTE dapat digunakan untuk mengatasi masalah *imbalance class* dan meningkatkan kemampuan model klasifikasi.

Penggunaan *Grid Search* menghasilkan kombinasi nilai C dan gamma yang berbeda pada masing-masing *kernel*. Pada *kernel RBF*, penggunaan *Grid Search* tanpa SMOTE menghasilkan *accuracy* sebesar 72,18%, sedangkan kombinasi *Grid Search* dan SMOTE menghasilkan *accuracy* sebesar 75,47%. Hasil tersebut menunjukkan bahwa penentuan *hyperparameter* yang sesuai dapat meningkatkan performa model, terutama ketika dikombinasikan dengan penyeimbangan kelas. Pratama dan Trilaksono (2015) menunjukkan bahwa pemilihan parameter pada SVM dapat menghasilkan performa klasifikasi yang berbeda pada data *Twitter*. Oleh karena itu, proses

hyperparameter tuning diperlukan untuk memperoleh konfigurasi model yang lebih sesuai dengan karakteristik data.

Skenario terbaik diperoleh pada skenario 8, yaitu *kernel RBF* dengan penerapan SMOTE dan *Grid Search*. Model tersebut menghasilkan *accuracy* sebesar 75,47%, *precision* 75,95%, *recall* 75,47%, dan *F1-score* 75,54%. Hasil tersebut menunjukkan bahwa kombinasi pemilihan *kernel* yang tepat, penyeimbangan kelas, dan optimasi *hyperparameter* mampu menghasilkan performa klasifikasi yang lebih baik dibandingkan konfigurasi lainnya. Dengan demikian, tujuan penelitian untuk mengetahui pengaruh pemilihan *kernel*, penerapan *Grid Search*, dan SMOTE terhadap klasifikasi sentimen *tweet* berbahasa Indonesia dapat tercapai.

KESIMPULAN

Berdasarkan hasil penelitian, dapat disimpulkan bahwa pemilihan *kernel* SVM berpengaruh terhadap performa klasifikasi sentimen *tweet* berbahasa Indonesia. Dari empat *kernel* yang diuji, *kernel RBF* menghasilkan performa terbaik dengan *accuracy* sebesar 75,47% pada kombinasi SMOTE dan *Grid Search*, serta rata-rata *accuracy* sebesar 73,42%.

Penerapan *Grid Search* memberikan peningkatan performa pada beberapa konfigurasi *kernel* dengan menentukan kombinasi *hyperparameter* yang sesuai. Sementara itu, penerapan SMOTE mampu meningkatkan performa terutama pada *kernel RBF* dan *polynomial* dengan membantu mengatasi ketidakseimbangan kelas. Secara keseluruhan, kombinasi *kernel RBF*, SMOTE, dan *Grid Search* merupakan konfigurasi terbaik dalam penelitian dengan *accuracy* 75,47%, *precision* 75,95%, *recall* 75,47%, dan *F1-score* 75,54%.

UCAPAN TERIMA KASIH

Penulis mengucapkan terima kasih kepada semua pihak yang telah memberikan dukungan dan bantuan dalam pelaksanaan penelitian ini. Ucapan terima kasih secara khusus disampaikan kepada dosen pembimbing atas arahan dan masukan yang diberikan selama proses penelitian, serta kepada seluruh pihak yang telah membantu dalam penyediaan data dan pelaksanaan penelitian. Semoga hasil penelitian ini dapat memberikan manfaat dan menjadi referensi bagi penelitian selanjutnya.

DAFTAR PUSTAKA

Aulia, T. M. P. (2021). Perbandingan kernel Support Vector Machine (SVM) dalam penerapan analisis sentimen vaksinisasi COVID-19. *SINTECH (Science and Information Technology) Journal, 4*(2), 139–145. https://doi.org/10.31598/sintechjournal.v4i2.762

Darwis, M., Pranoto, G. T., Wicaksana, Y. E., & Yaddarabullah, Y. (2020). Implementation of TF-IDF algorithm and K-mean clustering method to predict words or topics on Twitter. *JISA (Jurnal Informatika dan Sains), 3*(2), 49–55. https://doi.org/10.31326/jisa.v3i2.831

Ferdiana, R., Jatmiko, F., Purwanti, D. D., Tri Ayu, A. O., & Dicka, R. (2019). Dataset Indonesia untuk analisis sentimen. *Proceedings of the 2019 International Conference on Information and Communication Technology*.

Jumeilah, F. S. (2017). Penerapan Support Vector Machine (SVM) untuk pengkategorian penelitian. *Jurnal RESTI (Rekayasa Sistem dan Teknologi Informasi)*, *1*(1), 19–25. https://doi.org/10.29207/resti.v1i1.11

Khairunnisa, S., Adiwijaya, A., & Faraby, S. Al. (2021). Pengaruh *text preprocessing* terhadap analisis sentimen komentar masyarakat pada media sosial Twitter (Studi kasus pandemi COVID-19). *Jurnal Media Informatika Budidarma*, *5*(2), 406. https://doi.org/10.30865/mib.v5i2.2835

Meidianingsih, Q., & Meganingtyas, D. E. W. (2022). Analisis perbandingan performa metode ensemble dalam menangani *imbalanced multi-class classification*. *Jurnal Aplikasi Statistika & Komputasi Statistik*, *14*(2), 13–21. https://doi.org/10.34123/jurnalasks.v14i2.335

Passonneau, R. (2011). Sentiment analysis of Twitter data. 30–38.

Pratama, E. E., & Trilaksono, B. R. (2015). Klasifikasi topik keluhan pelanggan berdasarkan tweet dengan menggunakan penggabungan feature hasil ekstraksi pada metode Support Vector Machine (SVM). *Jurnal Edukasi dan Penelitian Informatika (JEPIN)*, *1*(2). https://doi.org/10.26418/jp.v1i2.11023

Rozi, I. F., Ardiansyah, R., & Rebeka, N. (2019). Penerapan normalisasi kata tidak baku menggunakan Levenshtein Distance pada analisa sentimen layanan PT. KAI di Twitter. *Seminar Informatika Aplikatif*, 106–112.

Sutoyo, E., & Fadlurrahman, M. A. (2020). Penerapan SMOTE untuk mengatasi *imbalance class* dalam klasifikasi Television Advertisement Performance Rating menggunakan Artificial Neural Network. *Jurnal Edukasi dan Penelitian Informatika (JEPIN)*, *6*(3), 379. https://doi.org/10.26418/jp.v6i3.42896

Utomo, M. S. (2013). Implementasi Stemmer Tala pada aplikasi berbasis web. *Jurnal Teknologi Informasi DINAMIK*, *18*(1), 41–45.