

Penerapan Support Vector Machine untuk Analisis Sentimen Fitur Layanan pada Ulasan Gojek

Jonathan Adrian Wibowo¹⁾ Viny Christanti Mawardi²⁾ Tri Sutrisno³⁾

¹⁾²⁾³⁾ Teknik Informatika, FTI, Universitas Tarumanagara

Jl. Letjen S Parman no 1, Jakarta 11440 Indonesia

¹⁾email: jonathan.535190065@stu.untar.ac.id ²⁾email: viny@fti.untar.ac.id ³⁾email: tris@fti.untar.ac.id

ABSTRAK

Perkembangan teknologi informasi, seperti yang dapat dilihat oleh platform seperti Gojek telah mengubah cara manusia dalam beraktivitas. Gojek, yang hadir sebagai aplikasi layanan transportasi dan makanan berbasis teknologi telah membawa banyak dampak signifikan dalam cara masyarakat bergerak, berbelanja, dan beraktivitas. Google Play Store sebagai salah satu gerbang bagi pengguna aplikasi untuk mengakses aplikasi seperti Gojek, menyediakan informasi penting tentang aplikasi, salah satunya ulasan pengguna untuk membantu pengguna dalam membuat keputusan sebelum mengunduh aplikasi tersebut. Oleh sebab itu, penelitian ini akan melakukan analisis sentimen berbasis fitur dengan fokus ulasan pengguna terhadap fitur layanan Gojek. Dataset yang digunakan terdiri dari 2000 data ulasan pengguna yang diambil dari website Google Play Store. Proses klasifikasi akan menggunakan algoritma *Support Vector Machine*. Dari hasil evaluasi menggunakan *Confusion Matrix*, ditemukan kernel RBF mendapatkan hasil akurasi terbaik, yaitu sebesar 95,26% untuk analisis fitur dan 88,53% untuk analisis sentimen dengan pembagian data latih dan uji yang sama, yaitu 80/20. Sebaliknya, hasil kurang baik ditemukan pada kernel *Polynomial*, yaitu 75,06% untuk analisis fitur dengan pembagian data latih dan uji 80/20. Sedangkan, untuk analisis sentimen sebesar 79,28% dengan pembagian data latih dan uji 60/40. Dari penelitian ini, diperoleh model pelatihan terbaik yang bisa digunakan adalah kernel RBF sebagai metode klasifikasi.

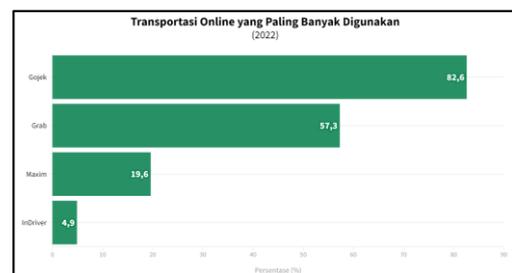
Kata Kunci

Confusion Matrix, Gojek, Google Play Store, Kernel, *Support Vector Machine*

1. Pendahuluan

Dalam era digital yang terus berkembang dari waktu ke waktu, teknologi informasi dan

komunikasi telah mengubah berbagai macam aspek kehidupan manusia. Dimulai dari cara manusia berinteraksi, bertransaksi, dan mengakses berbagai layanan yang ada. Aplikasi berbasis platform telah menjadi salah satu contohnya, dimana manusia semakin terbiasa akan pola baru dalam kehidupan sehari-hari, seperti hadirnya aplikasi layanan transportasi dan makanan berbasis teknologi, yaitu Gojek, yang telah membawa banyak dampak signifikan dalam cara masyarakat bergerak, berbelanja, dan beraktivitas.



Gambar 1. Hasil Survei Indef

Berdasarkan Gambar 1, Gojek tercatat sebagai layanan ojek online favorit masyarakat Indonesia. Tercatat ada 82% responden yang menggunakan layanan milik PT GoTo Gojek Tokopedia Tbk. tersebut, meski memiliki aplikasi lainnya [1]. Sebagai platform aplikasi berbasis online, Gojek telah memperkenalkan konsep layanan serba ada melalui platformnya yang terintegrasi.

Tidak hanya menyediakan layanan transportasi, Gojek juga menyediakan berbagai layanan lain seperti pengiriman makanan, pembayaran tagihan, dan bahkan pembelian produk. Dengan demikian, Gojek bergerak dalam berbagai sektor, menggabungkan teknologi dan layanan tradisional untuk memberikan solusi dalam kehidupan sehari-hari masyarakat. Oleh sebab itu, ulasan pelanggan yang diterima sangat penting adanya. Dimana, ulasan yang sangat beragam dari pengguna inilah yang mencerminkan pengalaman mereka dalam menggunakan layanan-layanan tersebut.

Google Play Store sendiri merupakan platform distribusi digital yang dikembangkan oleh Google. Platform ini merupakan toko resmi tempat pengguna dapat mengunduh dan memperbarui berbagai jenis aplikasi dan permainan untuk perangkat mereka. Pengguna dapat mencari berbagai aplikasi yang tersedia, termasuk aplikasi Gojek.

Google Play Store juga menyediakan informasi penting tentang aplikasi, seperti deskripsi, ulasan pengguna, peringkat, dan sebagainya yang membantu pengguna dalam membuat keputusan sebelum mengunduh aplikasi tertentu. Dengan demikian, Google Play Store berperan sebagai gerbang utama bagi pengguna aplikasi di dalamnya untuk mengakses berbagai aplikasi, serta memungkinkan mereka untuk memberikan komentar atau keluhan terhadap aplikasi tersebut dengan mudah melalui perangkat mereka.

Analisis sentimen (*sentiment analysis*) adalah sebuah teknik atau cara yang digunakan untuk mengidentifikasi bagaimana sebuah sentimen diekspresikan menggunakan teks dan bagaimana sentimen tersebut bisa dikategorikan sebagai sentimen positif maupun sentimen negatif [2].

Demikian pula adanya dengan analisis fitur atau aspek. Dimana analisis fitur ini merupakan subset dari analisis sentimen yang lebih fokus pada proses identifikasi dan analisis sentimen terkait dengan fitur-fitur tertentu dari suatu produk, layanan, atau topik lainnya.

Tujuan utama dalam analisis fitur adalah untuk memahami bagaimana orang merespons atau menilai suatu aspek-aspek spesifik suatu subjek. Misalnya, dalam konteks aplikasi Gojek, aspek-aspek yang akan diteliti mungkin mencakup performa sistem, antarmuka aplikasi, layanan dan lain sebagainya. Analisis fitur memungkinkan untuk melihat pandangan dan sentimen yang lebih rinci terkait dengan berbagai fitur atau aspek yang dianggap penting oleh pengguna. Proses analisis fitur melibatkan identifikasi aspek-aspek yang relevan dalam teks (pengenalan aspek), ekstraksi sentimen terkait aspek tersebut, dan penyajian hasil secara terstruktur.

Dalam proses analisis sentimen berbasis fitur akan dilakukan proses klasifikasi yang akan menggunakan metode algoritma *Support Vector Machine*. Analisis sentimen ini akan mengklasifikasikan sentimen dari ulasan-ulasan pada aplikasi Gojek, yang kemudian akan dilakukan proses analisis kembali terhadap fitur layanan dalam ulasan aplikasi tersebut.

Algoritma *Support Vector Machine (SVM)* sendiri merupakan metode pembelajaran mesin yang biasa digunakan untuk klasifikasi dan

regresi. SVM bekerja dengan membangun bidang pembatas terbaik (*hyperplane*) yang memaksimalkan jarak antara dua kelas data yang berbeda dalam ruang fitur. SVM bertujuan untuk memisahkan data dengan margin maksimal, yaitu jarak antara *hyperplane* dan titik-titik data terdekat (*support vectors*) dari masing-masing kelas.

Algoritma SVM akan mengajarkan model menggunakan data pelatihan yang berisi contoh data ulasan yang sudah diklasifikasikan secara manual ke dalam dua kategori sentimen dan dua kategori fitur atau aspek, SVM dapat mempelajari pola-pola dalam teks yang mengindikasikan kelas tertentu. Setelah dilatih, model SVM dapat digunakan untuk mengklasifikasikan ulasan baru ke dalam kategori kelas yang sesuai.

Algoritma ini biasa digunakan untuk menganalisis sentimen karena beberapa alasan yang diantaranya: kemampuan SVM untuk memisahkan dua kelas dengan margin maksimal, penanganan data yang kompleks atau tingkat tinggi, penanganan data yang tidak tersebar secara *linear*, dan pengendalian *overfitting* (ketidakmampuan model mengeneralisasi data dengan baik ke data yang belum pernah dilihat sebelumnya).

Penelitian yang dibuat terkait penerapan algoritma *Support Vector Machine (SVM)* untuk melakukan analisis sentimen berbasis fitur layanan pada data ulasan aplikasi Gojek. Penelitian ini bertujuan untuk memahami dan mengekstraksi sentimen positif dan negatif terhadap fitur layanan dalam ulasan, yaitu ulasan yang berisi komentar pengguna terhadap seluruh layanan pada aplikasi Gojek. Sehingga, setelah data berhasil diambil melalui proses *scraping* data akan diberi label sesuai sentimen dan fitur tersebut.

Metode yang digunakan dalam penelitian ini melibatkan pra-pemrosesan data yang berisi teks seperti *case folding*, pembersihan karakter khusus dan angka, *tokenization*, *stopword removal* (kata-kata yang dianggap penting akan dihapus terlebih dahulu dari dalam daftar *stopword*), terjemahan teks Bahasa Inggris, dan *lemmatization*.

Setelah itu, teks akan dikonversikan menjadi representasi atau vektor numerik dengan menggunakan metode TF-IDF (*Term Frequency-Inverse Document Frequency*) yang selanjutnya akan dimanfaatkan oleh algoritma SVM. Proses berikutnya, algoritma SVM diterapkan untuk melatih model klasifikasi sentimen berdasarkan fitur yang ada dalam ulasan. Model yang telah dibuat akan dilakukan pengujian untuk menguji model SVM tersebut.

Hasil prediksi data uji akan dievaluasi berdasarkan nilai sebenarnya untuk mengukur kinerja model SVM menggunakan metode evaluasi *confusion matrix*. Hal ini membantu melihat seberapa baik model dapat mengklasifikasikan sentimen dan fitur pada data ulasan pengguna.

2. Metode Penelitian

Dalam membangun sistem analisis sentimen berbasis fitur layanan pada data ulasan aplikasi Gojek dibutuhkan pembahasan mengenai teori dasar dari tahapan dan juga metode yang akan digunakan untuk proses klasifikasi. Pembahasan teoritik dasar serta metode terkait yang akan digunakan dalam penelitian akan dibahas pada subbab berikut ini.

2.1 Analisis Sentimen Berbasis Fitur

Analisis sentimen (*sentiment analysis*) adalah sebuah teknik atau cara yang digunakan untuk mengidentifikasi bagaimana sebuah sentimen diekspresikan menggunakan teks dan bagaimana sentimen tersebut bisa dikategorikan sebagai sentimen positif maupun sentimen negatif [2]. Sedangkan, analisis sentimen berbasis fitur merupakan pendekatan dalam pemrosesan bahasa alami yang bertujuan untuk memahami dan menganalisis perasaan atau pendapat yang terkandung dalam teks, dengan fokus pada fitur atau elemen tertentu dari suatu subjek atau topik [3]. Pendekatan ini digunakan untuk mengidentifikasi sentimen positif atau negatif terhadap fitur-fitur yang telah ditentukan sebelumnya dalam teks, seperti dalam ulasan produk atau komentar sosial media.

2.2 Term Frequency – Inverse Document Frequency (TF-IDF)

Metode *Term Frequency Inverse Document Frequency* (TF-IDF) merupakan teknik penentuan seberapa term mewakili konten dalam dokumen dengan memberi bobot ke masing-masing kata yang terkandung di dalamnya. TF-IDF biasa digunakan untuk analisis teks dan pengambilan informasi (*information retrieval*) untuk menilai pentingnya suatu kata atau frase dalam dokumen berdasarkan seberapa sering kata atau frase tersebut muncul dalam dokumen (*term frequency* atau TF) dan seberapa unik suatu kata atau frase tersebut dalam kumpulan dokumen (*Inverse Document Frequency* atau IDF) [4].

Tujuan utama dari TF-IDF adalah untuk membantu dalam mengidentifikasi kata-kata yang paling penting atau karakteristik dalam suatu dokumen relatif terhadap dokumen-dokumen lain dalam kumpulan.

Rumus TF-IDF adalah sebagai berikut:

$$TF - IDF(t, d) = TF(t, d) \times IDF(t) \quad (1)$$

Dimana, $TF(t, d)$ dihitung dengan rumus sebagai berikut:

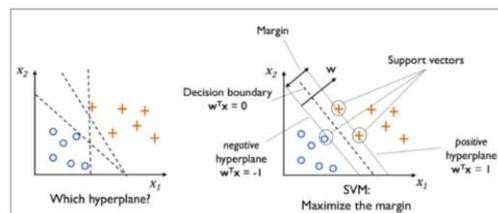
$$TF(t, d) = \frac{\text{jumlah kata } t \text{ dalam dokumen } d}{\text{total kata dalam dokumen } d} \quad (2)$$

dan, $IDF(t)$ dihitung dengan rumus sebagai berikut:

$$IDF(t) = \log\left(\frac{\text{total dokumen dalam korpus}}{\text{jumlah dokumen yang mengandung kata } t}\right) \quad (3)$$

2.3 Support Vector Machine (SVM)

Support Vector Machine (SVM) merupakan salah satu metode dalam *supervised learning* yang biasanya digunakan untuk klasifikasi (seperti *Support Vector Classification*) dan regresi (*Support Vector Regression*). SVM digunakan untuk mencari *hyperplane* terbaik dengan memaksimalkan jarak antar kelas. *Hyperplane* adalah sebuah fungsi yang dapat digunakan untuk pemisah antar kelas. Dalam 2 dimensi fungsi yang digunakan untuk klasifikasi antar kelas disebut sebagai *line whereas*, fungsi yang digunakan untuk klasifikasi antar kelas dalam 3 dimensi disebut *plane similarly*, sedangkan fungsi yang digunakan untuk klasifikasi di dalam ruang kelas dimensi yang lebih tinggi di sebut *hyperplane* [5].



Gambar 2. *Hyperplane* yang memisahkan kelas positif (+1) dan negatif (-1)

Hyperplane yang ditemukan SVM diilustrasikan seperti Gambar 2 posisinya berada ditengah-tengah antara dua kelas, artinya jarak antara *hyperplane* dengan objek-objek data berbeda dengan kelas yang berdekatan (terluar) yang diberi tanda bulat kosong dan positif. Dalam SVM objek data terluar yang paling dekat dengan *hyperplane* disebut *support vector*. Objek yang disebut *support vector* paling sulit diklasifikasikan dikarenakan posisi yang hampir tumpang tindih (*overlap*) dengan kelas lain. Mengingat sifatnya yang kritis, hanya *support vector* inilah yang diperhitungkan untuk

menemukan *hyperplane* yang paling optimal oleh SVM.

Jenis pada *Support Vector Machine* yang paling sederhana adalah *Support Vector Machine linear*. *Hyperplane* untuk klasifikasi *Support Vector Machine linear* dapat diartikan dengan rumus berikut:

$$x_i \cdot w + b = 0 \quad (4)$$

Dalam *hyperplane*, terdapat *supporting hyperlines* yang merupakan dua buah bidang pembatas, dimana bidang pembatas yang pertama merupakan pembatas kelas pertama dan bidang pembatas yang kedua adalah pembatas dari kedua sehingga didapatkan pertidaksamaan *linear* sebagai berikut:

$$x_i \cdot w + b \leq 0 \text{ untuk } y_i = -1 \quad (5)$$

$$x_i \cdot w + b \geq 0 \text{ untuk } y_i = +1 \quad (6)$$

Vektor bobot yang menggambarkan posisi dari *hyperplane* dibidang normal dilambangkan dengan w , sedangkan b merupakan bias yang menggambarkan posisi bidang relatif terhadap koordinat. Margin merupakan jarak di antara bidang pembatas dan nilai margin tersebut adalah $\frac{2}{\|w\|}$. Untuk mencari nilai margin terbesar (*maximum margin*), perlu memaksimalkan $\frac{1}{\|w\|}$ berdasarkan rumus jarak garis ke titik pusatnya. Klasifikasi kelas data pada persamaan (5) dan (6) dapat digabungkan dan direpresentasikan sebagai berikut:

$$y_i(x_i \cdot w + b) - 1 \geq 0 \quad (7)$$

Untuk menghitung margin yang optimal, jarak antara *hyperplane* dan data yang terdekat akan dimaksimalkan dengan memformulasikan masalah ini ke dalam *problem quadratic programming* (QP) yaitu dengan meminimalkan invers dari persamaan (7), berdasarkan *constraint* (syarat) sebagai berikut:

$$\frac{1}{2} \|w\|^2 \quad (8)$$

Syarat:

$$y_i(x_i \cdot w + b) - 1 \geq 0, i = 1, 2, \dots, N \quad (9)$$

Masalah ini dapat diselesaikan dengan menggunakan *Lagrange Multiplier*:

$$L(w, b, \alpha) = \frac{1}{2} \|w\|^2 - \sum_{i=1}^N \alpha_i y_i (x_i \cdot w + b) - 1 \quad (10)$$

α_i merupakan *Lagrange Multiplier* yang bernilai 0 atau positif dan berkorespondensi dengan x_i . Untuk meminimalkan Lagrangian, persamaan (10) harus diturunkan terhadap w dan b kemudian diubah dengan nilai 0 dengan syarat optimalisasi:

Syarat 1:

$$\frac{\partial L}{\partial w} = 0, w = \sum_{i=1}^N \alpha_i y_i x_i \quad (11)$$

Syarat 2:

$$\frac{\partial L}{\partial b} = 0, w = \sum_{i=1}^N \alpha_i y_i = 0 \quad (12)$$

Dengan N merupakan jumlah data yang menjadi *Support Vector*. Persamaan (10) dapat dijabarkan menjadi:

$$L(w, b, \alpha) = \frac{1}{2} \|w\|^2 - \sum_{i=1}^N \alpha_i y_i (x_i \cdot w) - b \sum_{i=1}^N \alpha_i y_i + \sum_{i=1}^N \alpha_i \quad (13)$$

Persamaan akan berubah dengan menerapkan syarat optimal yang terdapat pada persamaan (12) dalam suku ketiga persamaan (13), sehingga akan didapatkan rumus *Maximize* sebagai berikut:

$$L(w, b, \alpha) = \sum_{i=1}^N \alpha_i - \frac{1}{2} \sum_{i,j=1}^N \alpha_i \alpha_j x_i x_j y_i y_j \quad (14)$$

Hasil yang akan didapatkan adalah nilai α_i dari setiap data latih dan data latih yang memiliki nilai $\alpha_i > 0$ adalah *support vector* dan nilai α_i dari data latih sisanya adalah nol. Dengan demikian, *hyperplane* yang dihasilkan hanya akan dipengaruhi oleh *Support Vector*.

Dikarenakan formula dari pencarian *hyperplane* ini termasuk dalam *problem quadratic programming*, nilai maksimum α_i selalu dapat ditemukan. Setelah solusi dari *problem quadratic programming* ini ditemukan, kelas dari data uji x_i dapat ditentukan berdasarkan nilai dari fungsi keputusan:

$$f(x) = y \alpha'(x_i x_j) + b \quad (15)$$

Keterangan:

y = Label data latih

α' = Nilai yang didapatkan dari perhitungan *Lagrange Multiplier*

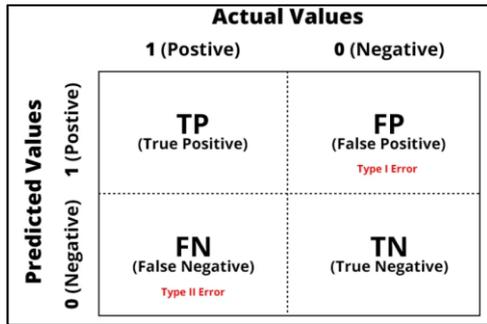
x_i = *Support Vector*

x_j = Data yang akan diklasifikasi

b = Bias yang menggambarkan posisi bidang relatif terhadap koordinat

2.4 Confusion Matrix

Confusion matrix juga sering disebut *error matrix*. Pada dasarnya *confusion matrix* memberikan informasi perbandingan hasil klasifikasi yang dilakukan oleh sistem (model) dengan hasil klasifikasi sebenarnya. *Confusion matrix* berbentuk tabel matriks yang menggambarkan kinerja model klasifikasi pada serangkaian data uji yang nilai sebenarnya diketahui. Gambar 3 dibawah ini merupakan *confusion matrix* dengan 4 kombinasi nilai prediksi dan nilai aktual yang berbeda [6].



Gambar 3. Confusion Matrix

Metode perhitungan pada *confusion matrix* dijabarkan melalui perhitungan pada nilai *accuracy*, *precision*, dan *recall*. *Accuracy* menggambarkan seberapa akurat model dapat mengklasifikasikan dengan benar. Maka, *accuracy* adalah rasio prediksi benar (positif dan negatif) dengan keseluruhan data yang ada. Dengan kata lain, *accuracy* merupakan tingkat kedekatan nilai prediksi dengan nilai aktual (sebenarnya). Nilai *accuracy* dapat diperoleh dengan persamaan:

$$accuracy = \frac{TP+TN}{TP+TN+FP+FN} \times 100\% \quad (16)$$

Precision memperlihatkan tingkat akurasi antara data yang diminta dengan hasil prediksi yang diberikan oleh sistem. Sehingga *precision*, yaitu rasio prediksi benar positif dibandingkan dengan keseluruhan hasil yang diprediksi positif. Dari semua kelas positif yang di prediksi dengan benar, akan dicari berapa banyak data yang benar-benar positif. Nilai *precision* dapat diperoleh dengan persamaan:

$$precision = \frac{TP}{TP+FP} \times 100\% \quad (17)$$

Recall memperlihatkan keberhasilan sistem dalam menemukan kembali sebuah informasi. Maka, *recall* merupakan rasio prediksi benar positif dibandingkan dengan keseluruhan data yang benar positif. Nilai *recall* didapat dengan persamaan:

$$recall = \frac{TP}{TP+FN} \times 100\% \quad (18)$$

F₁Score menggambarkan perbandingan rata-rata *precision* dan *recall* yang dibobotkan. *Accuracy* tepat digunakan sebagai acuan performansi algoritma jika dataset memiliki jumlah data *False Negatif* dan *False Positif* yang sangat mendekati (*symmetric*). Tapi jika jumlahnya tidak mendekati, sebaiknya menggunakan *F₁Score* sebagai acuan:

$$F_1Score = 2 \times \frac{precision \times recall}{precision + recall} \quad (19)$$

3. Hasil Dan Pembahasan

Pada pengujian ini, program menggunakan metode *Support Vector Machine* untuk melakukan klasifikasi sentimen terhadap fitur layanan untuk data ulasan aplikasi Gojek dari Google Play Store. Data sebelumnya diberikan label fitur layanan dan non-layanan untuk kolom feature, sedangkan untuk kolom sentimen diberikan label positif dan negatif untuk masing-masing ulasan pengguna. Terdapat 2 tahapan pelatihan dan pengujian pada data, dimana data akan dilakukan pelatihan dan pengujian terlebih dahulu pada label fitur layanan dan non layanan, lalu selanjutnya dengan data yang sama akan dilakukan pelatihan dan pengujian pada label sentimen positif dan negatif. Jumlah data ulasan yang akan digunakan di dalam penelitian sebanyak 2000 data.

Setelah melakukan tahap pengujian, akan dilakukan evaluasi dengan menggunakan metode *Confusion Matrix*. Keseluruhan data akan dilakukan pelatihan dan pengujian masing-masing 3 kali dengan rasio yang berbeda untuk setiap kernel *linear*, *RBF*, dan *polynomial*. Tabel pembagian data dapat dilihat pada Tabel 1 dan 2 dibawah.

Tabel 1. Pembagian *Data Training* dan *Data Testing* untuk tahapan fitur

Pengulangan Ke-	Data Training	Data Testing	Pembagian Data
1	800	200	80/20
2	700	300	70/30
3	600	400	60/40

Tabel 2. Pembagian *Data Training* dan *Data Testing* untuk tahapan sentimen

Pengulangan Ke-	Data Training	Data Testing	Pembagian Data
1	800	200	80/20
2	700	300	70/30
3	600	400	60/40

Hasil *confusion matrix* untuk analisis fitur pada tahapan pertama dapat dilihat pada ketiga tabel dibawah.

Tabel 3. *Confusion Matrix* pengulangan ke-1

Kernel	Pembagian Data		Akurasi
	Data Training	Data Testing	
Linear	80%	20%	94,51%
RBF			95,26%
Polynomial			75,06%

Pada pengulangan pertama, hasil akurasi terbesar yang didapatkan adalah kernel *RBF*, yaitu sebesar 95,26%. Lalu, untuk kernel *linear*, yaitu sebesar 94,51%. Sedangkan, yang paling terkecil ada pada kernel *polynomial*, yaitu sebesar 75,06%. Jumlah *data training* yang digunakan pada percobaan ini sebesar 1600 data (80%) dan *data testing* sebesar 400 data (20%).

Tabel 4. *Confusion Matrix* pengulangan ke-2

Kernel	Pembagian Data		Akurasi
	Data Training	Data Testing	
Linear	70%	30%	94,34%
RBF			94,84%
Polynomial			78,2%

Pada pengulangan kedua, hasil akurasi terbesar yang didapatkan adalah kernel *RBF*, yaitu sebesar 94,84%. Selanjutnya, untuk kernel *linear*, yaitu sebesar 94,34%. Sedangkan, hasil akurasi terkecil didapatkan pada kernel *polynomial*, yaitu sebesar 78,2%. Jumlah *data training* yang digunakan pada percobaan ini sebesar 1400 data (70%) dan *data testing* sebesar 600 data (30%).

Tabel 5. *Confusion Matrix* pengulangan ke-3

Kernel	Pembagian Data		Akurasi
	Data Training	Data Testing	
Linear	60%	40%	94,26%
RBF			93,76%
Polynomial			78,78%

Pada pengulangan ketiga, hasil akurasi terbesar yang didapatkan adalah kernel *linear*, yaitu sebesar 94,26%. Sedangkan, untuk kernel *RBF* sebesar 93,76%. Lalu, yang paling terkecil

didapatkan adalah kernel *polynomial* sebesar 78,78%. Jumlah data *training* yang digunakan pada percobaan ini sebesar 1200 data (60%) dan data *testing* sebesar 800 data (40%).

Dari ketiga hasil percobaan pada tahapan fitur layanan dan non-layanan yang dilakukan dengan mengatur jumlah data yang hendak dilatih dan diuji melalui pembagian data 80/20, 70/30, dan 60/40 didapatkan hasil yang sangat baik untuk kernel *linear* dan *RBF* dimana untuk kedua kernel tersebut mendapatkan hasil akurasi diatas 90% terhadap ketiganya. Sedangkan, untuk kernel *polynomial* hasil akurasi yang didapatkan masih dibawah 80%, sehingga terdapat perbedaan yang cukup signifikan apabila dibandingkan dengan dua kernel sebelumnya.

Hasil terbaik didapatkan pada pembagian data 80/20, dimana akurasi terbaik didapatkan oleh kernel *RBF* dengan hasil akurasi sebesar 95,26% seperti bisa dilihat pada Tabel 3. Sedangkan, hasil kurang baik didapatkan pada kernel *polynomial*, yaitu sebesar 75,06% seperti juga bisa dilihat pada Tabel 3.

Selanjutnya, hasil *confusion matrix* untuk analisis sentimen pada tahapan kedua dapat dilihat pada ketiga tabel dibawah.

Tabel 6. *Confusion Matrix* pengulangan ke-1

Kernel	Pembagian Data		Akurasi
	Data Training	Data Testing	
Linear	80%	20%	87,53%
RBF			88,53%
Polynomial			80,3%

Pada pengulangan pertama, hasil akurasi terbesar yang didapatkan adalah kernel *RBF*, yaitu sebesar 88,53%. Lalu, untuk kernel *linear*, yaitu sebesar 87,53%. Sedangkan, yang paling terkecil ada pada kernel *polynomial*, yaitu sebesar 80,3%. Jumlah *data training* yang digunakan pada percobaan ini sebesar 1600 data (80%) dan *data testing* sebesar 400 data (20%).

Tabel 7. *Confusion Matrix* pengulangan ke-2

Kernel	Pembagian Data		Akurasi
	Data Training	Data Testing	
Linear	70%	30%	87,69%
RBF			87,85%
Polynomial			81,36%

Pada pengulangan kedua, hasil akurasi terbesar yang didapatkan adalah kernel *RBF*, yaitu sebesar 87,85%. Selanjutnya, untuk kernel *linear*, yaitu sebesar 87,69%. Sedangkan, hasil akurasi terkecil didapatkan pada kernel *polynomial*, yaitu sebesar 81,36%. Jumlah *data training* yang digunakan pada percobaan ini sebesar 1400 data (70%) dan *data testing* sebesar 600 data (30%).

Tabel 8. *Confusion Matrix* pengulangan ke-3

Kernel	Pembagian Data		Akurasi
	Data Training	Data Testing	
Linear	60%	40%	86,52%
RBF			87,14%
Polynomial			79,28%

Pada pengulangan ketiga, hasil akurasi terbesar yang didapatkan adalah kernel *RBF*, yaitu sebesar 87,14%. Sedangkan, untuk kernel *linear* sebesar 86,52%. Lalu, yang paling terkecil didapatkan adalah kernel *polynomial* sebesar 79,28%. Jumlah *data training* yang digunakan pada percobaan ini sebesar 1200 data (60%) dan *data testing* sebesar 800 data (40%).

Dari ketiga hasil percobaan pada tahapan sentimen positif dan negatif yang dilakukan dengan mengatur jumlah data yang hendak dilatih dan diuji melalui pembagian data 80/20, 70/30, dan 60/40 didapatkan hasil yang cukup baik untuk kernel *linear* dan *RBF*, walaupun tidak memiliki perbedaan yang signifikan dengan kernel *polynomial* dimana terjadi pada analisis fitur layanan dan non-layanan yang telah dilakukan sebelumnya dan hasil yang didapatkan antar pembagian data juga tidak mengalami perbedaan yang signifikan.

Hasil terbaik didapatkan pada pembagian data 80/20, dimana akurasi terbaik didapatkan oleh kernel *RBF* dengan hasil akurasi sebesar 88,53% seperti bisa dilihat pada Tabel 6. Sedangkan, hasil kurang baik apabila dibandingkan dengan yang lainnya didapatkan pada kernel *polynomial*, yaitu sebesar 79,28% seperti juga bisa dilihat pada Tabel 8.

4. Kesimpulan Dan Saran

Beberapa kesimpulan yang dapat diambil setelah melakukan semua pengujian yang dilakukan, antara lain:

1. Sistem yang dibuat dapat melakukan *scraping* data ulasan sebanyak 2000 data untuk aplikasi Gojek di Google Play Store.
2. Sistem yang dibuat dapat digunakan untuk analisis sentimen berbasis fitur pada aplikasi Gojek dalam Google Play Store.
3. Semakin banyak data yang digunakan, akan membuat proses *preprocessing* juga semakin lama, dikarenakan memakan banyak memori untuk data.
4. Hasil akurasi terbaik untuk analisis fitur didapat pada kernel *RBF* sebesar 95,26% dengan pembagian data *training* dan data *testing* sebesar 80/20. Sedangkan, hasil akurasi terbaik untuk analisis sentimen juga didapat pada kernel *RBF* sebesar 88,53% dengan pembagian data sebesar 80/20. Selanjutnya, hasil kurang baik untuk analisis fitur didapatkan pada kernel *polynomial*, yaitu sebesar 75,06% pada pembagian data *training* dan data *testing* sebesar 80/20. Sedangkan, hasil kurang baik pada analisis sentimen juga didapatkan pada kernel *polynomial*, yaitu sebesar 79,28% pada pembagian data sebesar 60/40.
5. Dalam mencari hasil evaluasi terbaik menggunakan *Confusion Matrix*, hasil akurasi yang didapatkan sangat bergantung kepada banyaknya data latih dan model data yang digunakan, seperti untuk analisis sentimen berbasis fitur yang dilakukan didapatkan model latih terbaik untuk kernel *RBF* dan rasio data latih dan uji sebesar 80/20. Oleh sebab itu, pentingnya menggunakan berbagai opsi pembagian data dan kernel untuk masing-masing menganalisis model tersebut guna mendapatkan hasil akurasi terbaik.
6. Sistem yang dibuat akan menggunakan algoritma *Support Vector Machine* dengan *training model* kernel *RBF* sebagai metode klasifikasi.

Saran yang dapat diusulkan untuk peneliti lainnya adalah menggunakan lebih banyak jumlah data guna mendapatkan hasil akurasi yang lebih baik dan mencoba untuk menambahkan fitur lainnya selain layanan. Selain itu, penelitian ini dapat dijadikan acuan untuk melakukan penelitian lain untuk aplikasi sejenis, misalnya: Grab, Indrive, Maxim, dsb guna, sehingga dapat melakukan evaluasi model yang lebih optimal.

REFERENSI

- [1] S. Sadya, "Survei Indef: Gojek Pimpin Pasar Ojek Online di Indonesia," 16 Aug 2023. [Online]. Available: <https://dataindonesia.id/varia/detail/survei-indef-gojek-pimpin-pasar-ojek-online-di-indonesia>.
- [2] T. Nasukawa dan J. Yi, "Sentiment Analysis: Capturing Favorability Using Natural Language Processing," dalam *Proceedings of the 2nd International Conference on Knowledge Capture*, 2003.
- [3] U. Rofiqoh, R. S. Perdana dan M. A. Fauzi, "Analisis Sentimen Tingkat Kepuasan Pengguna Penyedia Layanan Telekomunikasi Seluler Indonesia pada Twitter dengan Metode Support Vector Machine dan Lexion Based Feature," *Jurnal Pengembangan Teknologi Informasi dan Ilmu Komputer Universitas Brawijaya*, vol. 1, no. 12, pp. 1725-1732, 2017.
- [4] A. Deviyanto dan M. D. R. Wahyudi, "Penerapan Analisis Sentimen pada Pengguna Twitter Menggunakan Metode K-Nearest Neighbor," *Jurnal Informatika Sunan Kalijaga*, vol. 3, no. 1, pp. 1-13, 2018.
- [5] T. S. Furey, N. Chrstianini, B. D. W. Duffy, M. Schummer dan D. Haussler, "Support Vector Machine Classification and Validation of Cancer Tissue Samples Using Microarray Expression Data," *Bioinformatics*, vol. 16, no. 10, pp. 906-914, 2001.
- [6] K. S. Nugroho, "Confusion Matrix untuk Evaluasi Model pada Supervised Learning," 13 November 2019. [Online]. Available: <https://ksnugroho.medium.com/confusion-matrix-untuk-evaluasi-model-pada-unsupervised-machine-learning-bc4b1ae9ae3f>. [Diakses 8 September 2023].

Jonathan Adrian Wibowo, saat ini sebagai mahasiswa Program Studi Teknik Informatika Fakultas Teknologi Informasi Universitas Tarumanagara.

Viny Christanti Mawardi, memperoleh gelar S.Kom. dari Universitas Tarumanagara tahun 2004. Kemudian memperoleh gelar M.Kom. dari Universitas Indonesia tahun 2008. Saat ini aktif sebagai Dosen Tetap Fakultas Teknologi Informasi Universitas Tarumanagara, Jakarta.

Tri Sutrisno, memperoleh gelar S.Si dari Universitas Diponegoro tahun 2011. Kemudian memperoleh gelar M.Sc.dari Universitas Gadjah Mada pada tahun 2015. Dan saat ini aktif sebagai Dosen Tetap Program Studi Teknik Informatika Universitas Tarumanagara Jakarta.