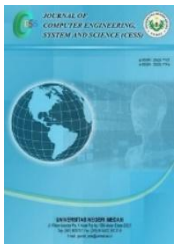
CESS (Journal of Computer Engineering, System and Science)
Vol. 11, No. 1, Januari 2026, pp.51-63

CESS
(Journal of Computer Engineering, System and Science)
Available online: https://jurnal.unimed.ac.id/2012/index.php/cess
ISSN: 2502-714x (Print) | ISSN: 2502-7131 (Online)



Perbandingan Kinerja Klasifikasi Sentimen Ulasan Aplikasi *Over the Top* di *Google Play Store*

Comparison of Classification Performance for Sentiment of Over the Top Application Reviews on Google Play Store

Tutus Pandam Pradipta[1*], Sri Huning Anwariningsih[2], Dahlan Susilo[3]

[1,2,3]Program Studi Informatika, Fakultas Sains, Teknologi dan Kesehatan,
Universitas Sahid Surakarta, Indonesia
Jl. Adi Sucipto No. 154 Jajar, Kecamatan Laweyan, Kota Surakarta, Jawa Tengah 57144
Email: [1]tutus.pandam@yahoo.com, [2]srihuning@usahidsolo.ac.id, [3]dahlan.susilo@usahidsolo.ac.id
**Corresponding Author*

A B S T R A K

Persaingan ketat layanan *Over The Top* (OTT) di Indonesia menuntut pemahaman terkait kepuasan pengguna sebagai kunci retensi pelanggan. Ulasan *Google Play Store* menjadi sumber data penting untuk klasifikasi sentimen. Meskipun metode SVM dan Naive Bayes populer untuk klasifikasi, tetapi perbandingan keduanya untuk klasifikasi sentimen layanan OTT masih terbatas. Penelitian ini membandingkan kinerja keduanya untuk klasifikasi ulasan Aplikasi OTT berbahasa Indonesia. Selain itu, dilakukan validasi eksternal terhadap *rating* Aplikasi untuk mengidentifikasi metode dan rasio data optimal, serta menganalisis korelasi antara kinerja sentimen positif dengan *rating* tersebut. Tahapan perbandingan dilakukan melalui metode KDD meliputi pengumpulan data, *text preprocessing*, transformasi data, hingga klasifikasi dengan pengujian pembagian data *training* dan *testing* (70:30, 80:20, dan 90:10) untuk mendapatkan model yang optimal. Klasifikasi diuji menggunakan berbagai matrik evaluasi yaitu Akurasi, Presisi, *Recall*, dan *F1-Score*. Model optimal kemudian divalidasi menggunakan koefisien korelasi Pearson antara *F1-Score* sentimen positif dan *rating* Aplikasi. Hasil kesimpulan dari perbandingan kedua metode menunjukkan bahwa SVM secara konsisten mengungguli Naive Bayes di seluruh skenario pengujian. Model SVM dengan rasio 90:10 adalah model optimal dengan akurasi rata-rata 75%. Model ini kuat pada sentimen negatif (*F1-Score*=0,91) namun menunjukkan kegagalan kritis pada sentimen netral (*F1-Score*=0,00). Analisis korelasi Pearson menghasilkan nilai r = 0,538, mengindikasikan adanya korelasi positif sedang antara sentimen positif dengan popularitas aplikasi. Implikasi penelitian ini adalah model optimal ini dapat digunakan sebagai alat *Business Intelligence* meskipun perlu pengembangan lanjutan untuk mengatasi *imbalance* data pada kelas netral.

Kata Kunci: *Klasifikasi Sentimen; Korelasi Pearson; Over The Top; Naive Bayes; Support Vector Machine;*

A B S T R A C T
Intense competition among Over The Top (OTT) services in Indonesia demands an understanding of user satisfaction as a key to customer retention. Google Play Store reviews serve as a crucial data source for sentiment classification. Although Support Vector Machine (SVM) and Naive Bayes methods are popular for classification, comparisons between the two for OTT service sentiment classification remain limited. This research compares the performance of both methods for classifying Indonesian-language OTT application reviews. Furthermore, external validation against application ratings was conducted to identify the optimal method and data ratio, as well as to analyze the correlation between positive sentiment performance and those ratings. The comparison stages were carried out through the Knowledge Discovery in Databases (KDD) method, encompassing data collection, text preprocessing, data transformation, and classification with testing across training and testing data splits (70:30, 80:20, and 90:10) to obtain the optimal model. Classification was evaluated using various metrics, namely Accuracy, Precision, Recall, and F1-Score. The optimal model was then validated using the Pearson correlation coefficient between the positive sentiment F1-Score and application ratings. The conclusion from the comparison of the two methods indicates that SVM consistently outperforms Naive Bayes across all testing scenarios. The SVM model with a 90:10 ratio emerged as the optimal model, achieving an average accuracy of 75%. This model is robust in detecting negative sentiment (F1-Score up to 0.91) but exhibits a critical failure in the neutral sentiment category (F1-Score = 0.00). Pearson correlation analysis yielded an r-value of 0.538, indicating a moderate positive correlation between positive sentiment and application popularity. The implication of this research is that this optimal model can serve as a Business Intelligence tool, although further development is required to address data imbalance in the neutral class.

Keywords: *Over The Top (OTT); Naive Bayes; Sentiment Classification; Support Vector Machine (SVM); Pearson Correlation.*

1. PENDAHULUAN

Tren perilaku digital di Indonesia menunjukkan adanya konvergensi substansial pada perangkat *mobile*, yang terindikasi dari peningkatan signifikan dalam pemanfaatan layanan berbasis internet, khususnya pada layanan *Over The Top* (OTT) [1]. Layanan OTT menawarkan keunggulan berupa fleksibilitas, kemudahan, dan efisiensi biaya yang signifikan dibandingkan dengan metode distribusi konvensional [2]. Keunggulan Layanan OTT tersebut diimbangi oleh kompetisi intensif di sektor hiburan, sehingga Perusahaan diwajibkan selalu berinovasi untuk meningkatkan layanan demi meningkatkan kepuasan konsumen melalui perancangan *user experience* yang unggul dan penyediaan produk atau jasa yang tepat sasaran [3].

Pada masa keterbukaan informasi dan partisipasi digital ini, ulasan pengguna Aplikasi OTT berfungsi sebagai aset krusial untuk mengevaluasi efektivitas layanan secara *real time*

[4]. Pentingnya analisis ulasan pengguna terletak pada pemberian wawasan komprehensif tentang *user experience*, yang diakomodasi melalui penggunaan metode analisis sentimen untuk mengidentifikasi pola dan tren data secara efektif [5].

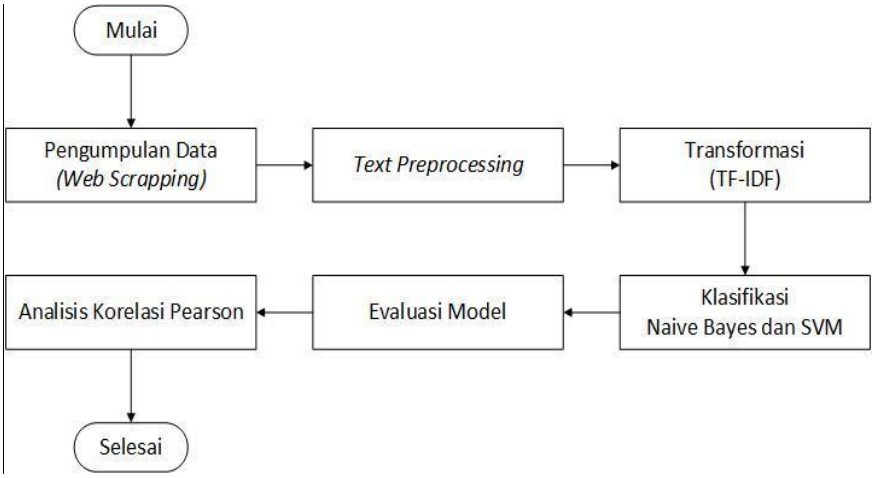
Dalam kerangka analisis sentimen, pendekatan *machine learning* memegang peranan esensial dalam pengembangan model klasifikasi, dengan memanfaatkan algoritma-algoritma kunci seperti Naive Bayes dan *Support Vector Machine* (SVM) [6].

Pada penelitian terdahulu yang dilakukan oleh Tanggraeni et al. [7] menggunakan algoritma Naive Bayes lalu membagi *data training* sebesar 90% dan *data testing* sebesar 10% pada 642 data ulasan. Pada penelitian ini menghasilkan akurasi sebesar 89%. Sedangkan penelitian yang dilakukan oleh Yolanda et al. [8] menggunakan algoritma SVM lalu membagi *data training* sebesar 80% dan *data testing* sebesar 20% untuk 4201 ulasan menghasilkan akurasi sebesar 89,29%. Selain itu terdapat juga dua penelitian terdahulu yang membandingkan kinerja algoritma Naive Bayes dan SVM. Kedua penelitian tersebut menunjukkan bahwa SVM memiliki akurasi lebih baik daripada Naive Bayes [9][10]. Meskipun demikian, terdapat dua keterbatasan kritis pada penelitian terdahulu. Pertama, penelitian hanya fokus pada satu objek aplikasi. Kedua, tidak adanya validasi yang mengukur secara kuantitatif seberapa jauh kinerja model sentimen positif berkorelasi dengan metrik performa bisnis yang nyata, yaitu *rating* aplikasi di *Google Play Store*. Untuk mengisi kesenjangan pada penelitian terdahulu tersebut pada penelitian ini mengusulkan pembangunan dua model klasifikasi sentimen menggunakan algoritma Naive Bayes dan SVM untuk Lima Aplikasi OTT. Penelitian ini bertujuan mengidentifikasi algoritma *Supervised Learning* dan rasio data optimal untuk klasifikasi sentimen ulasan Aplikasi OTT berbahasa Indonesia. Penelitian ini juga bertujuan untuk menganalisis korelasi linier antara kinerja sentimen positif dengan *rating* Aplikasi OTT di *Google Play Store*. Kontribusi dari penelitian ini terletak pada dua fokus utama. Pertama terkait evaluasi komprehensif kinerja kedua model menggunakan metrik Akurasi, Presisi, *Recall*, dan *F1-Score* dengan tiga skenario pembagian data (70:30, 80:20, dan 90:10). Pembagian skenario bertujuan untuk mengidentifikasi konfigurasi optimal bagi ulasan Aplikasi OTT. Kontribusi kedua adalah penelitian ini menggunakan koefisien korelasi Pearson untuk mengukur hubungan linier antara kinerja model sentimen positif yang diwakili oleh *F1-Score* sentimen positif dengan *rating* Aplikasi OTT di *Google Play Store*. Hasil penelitian ini memberikan rekomendasi algoritma yang andal bagi penyedia layanan OTT dan memberikan bukti empiris korelasi antara sentimen positif dengan metrik keberhasilan bisnis.

2. METODE PENELITIAN

Penelitian ini menggunakan pendekatan kuantitatif eksperimental untuk membangun model klasifikasi sentimen pada ulasan pengguna Aplikasi OTT di *Google Play Store*. Penelitian ini diimplementasikan menggunakan bahasa pemrograman *Python*. Bahasa pemrograman *Python* digunakan banyak oleh pengembang karena sintaksisnya yang jelas dan kemampuannya dalam menangani berbagai jenis tugas pemrograman [11]. Pada penelitian ini menggunakan dua algoritma *Supervised Learning* yaitu Naive Bayes dan SVM. Tahapan penelitian mengikuti metode *Knowledge Discovery in Databases (KDD), yaitu* mulai dari pengumpulan data, *text preprocessing*, transformasi TF-IDF, klasifikasi, evaluasi

performa klasifikasi, hingga analisis korelasi Pearson. KDD adalah mekanisme untuk menyelesaikan masalah dengan menganalisis data yang tersimpan dalam *database*, di mana data tersebut disimpan dalam bentuk elektronik dan pencarian dilakukan secara otomatis seperti pada komputer [12]. Alur penelitian disajikan pada Gambar 1.



Gambar 1. Alur Penelitian

Gambar 1 menunjukkan tahapan penelitian mulai dari pengumpulan data ulasan Lima Aplikasi OTT (Netflix, Vidio, Disney+, Prime Video, dan Catchplay+) di *Google Play Store* hingga analisis korelasi Pearson untuk memvalidasi hubungan kuantitatif antara kinerja model sentimen positif dengan *rating* Aplikasi OTT pada *Google Play Store.* Proses klasifikasi dapat dilihat pada *Pseudocode* Algoritma *Sentiment Classification and Business Validation* yang ditampilkan seperti Algoritma 1. Algoritma 1 menggambarkan proses data diolah secara berulang untuk setiap skenario rasio data, guna memastikan perbandingan yang objektif antara Naive Bayes dan SVM.

```
Algorithm: Sentiment Classification and Business Validation
Input: Preprocessed_Data (X), Labels (Y), App_Ratings (R)
Output: Evaluation Metrics, Optimal Model, Correlation (r)
Begin
X_tfidf = TF-IDF_Vectorization(X)
For each Ratio in [70:30, 80:20, 90:10]:
  Split (X_tfidf, Y) into Training_Set and Testing_Set
  For each Model in [Naive Bayes, SVM]:
    Train Model using Training_Set
    Predictions = Test Model on Testing_Set
    Calculate Accuracy, Precision, Recall, and F1-Score
  End For
End For
Identify Optimal_Model via highest average F1-Score
 r = Pearson_Correlation(F1_Positive_Optimal, R)
 Return All Metrics and r
 End
```

Gambar 2. Algoritma 1 *Sentiment Classification and Business Validation*

2.1. Pengumpulan Data Dengan Teknik *Web Scraping*

Salah satu fitur yang ada di *Google Play Store* adalah fitur ulasan. Dimana pengguna aplikasi dapat memberikan pendapat atau ulasan mereka tentang produk yang telah mereka gunakan [13]. Ulasan dari pengguna sangat penting karena bisa memberi gambaran mengenai sejauh mana pengguna merasa puas atau tidak puas dengan sebuah aplikasi [14]. Dalam penelitian ini, data ulasan yang terbaru dan berbahasa Indonesia adalah data yang dipilih. Data ulasan yang dipilih adalah ulasan aplikasi Netflix, Vidio, Disney+, Prime Video, dan Catchplay+ dari *Google Play Store*. Pada penelitian ini teknik *web scraping* digunakan dalam pengumpulan data ulasan. *Web scraping* adalah mekanisme yang digunakan untuk mengambil dan mengumpulkan informasi atau data dari situs *web* tertentu [15]. Pada penelitian ini teknik *web scraping* menggunakan *library google-play-scraper* dari *Python*. *Library* ini berfungsi dengan menargetkan *package name* dari Aplikasi yang dituju di *Google Play Store,* sehingga memungkinkan pengambilan data secara langsung tanpa terpengaruh oleh perubahan tata letak antar muka pada halaman *Google Play Store*. Target pengumpulan data ulasan dalam penelitian ini adalah Lima Aplikasi OTT. Masing-masing Aplikasi OTT diidentifikasi berdasarkan *package name,* yaitu Netflix (*com.netflix.mediaclient*), Vidio (*com.vidio.android*), Disney+ (*in.startv.hotstar.dplus*), Prime Video (*com.amazon.avod.thirdpartyclient*), dan Catchplay+ (*com.catchplay.asiaplay*).

2.2. *Text Preprocessing*

Tujuan dari *Text Preprocessing* adalah mengubah dokumen yang berupa teks tidak teratur menjadi bentuk yang lebih rapi dan mudah diproses [16]. Tahapan *Text Preprocessing* meliputi seleksi kolom, pelabelan data, *cleaning*, *case folding*, *stopword removal*, tokenisasi (*tokenizing*), dan *stemming.* Proses seleksi kolom bertujuan untuk menetapkan kolom yang relevan dengan klasifikasi. Kolom yang digunakan pada penelitian ini adalah *Content, Score, Label, Text Clean, Text Stopword, Text Tokens*, dan *Text Stemming*. Proses pelabelan data bertujuan mengkonversi skor bintang (*rating*) dari tiap ulasan Aplikasi OTT pada *Google Play Store* menjadi kategori sentimen yang diskrit seperti positif, netral, dan negatif. Proses pembersihan (*cleaning*) menghilangkan komponen yang tidak berhubungan (kolom, tanda baca, spasi kosong), diikuti *case folding* untuk memastikan konsistensi kapitalisasi. Proses *stopword removal* menghilangkan kata tidak relevan ("yang", "atau"). Tokenisasi mengubah kalimat menjadi daftar kata-kata, yang merupakan format *input* yang diperlukan untuk pelatihan model. Terakhir, *stemming* mengubah kata-kata berimbuhan ("menonton", "ditonton") menjadi kata dasar "tonton" untuk mengurangi redundansi fitur dan memastikan variasi kata yang memiliki arti serupa diperlakukan sebagai entitas tunggal oleh model.

2.3. Transformasi

Transformasi bertujuan untuk memberikan bobot lebih besar pada kata-kata penting dalam ulasan tertentu dan bobot lebih kecil pada kata-kata umum. Pada tahap transformasi data, metode TF-IDF dengan menggunakan kelas *TfidVectorizer* pada *Python* digunakan untuk mengubah teks menjadi vektor angka sesuai dengan frekuensi kata. Metode TF-IDF adalah cara untuk mengetahui seberapa penting suatu kata dalam sebuah dokumen

dibandingkan dengan dokumen-dokumen lain dalam kumpulan dokumen tersebut [17]. Rumus TF-IDF dapat dilihat pada persamaan (1).

$$w_{t,d} = w_{tf} \times idf_t \quad (1)$$

dimana

$w_{t,d}$: Pembobotan TF-IDF
w_{tf} : Bobot kata
idf_t : Bobot Inverse

2.4. Klasifikasi

Tahap klasifikasi bertujuan untuk menguji kinerja algoritma *machine learning* dalam memprediksi sentimen ulasan berdasarkan fitur yang telah diekstrak. Proses klasifikasi dilakukan menggunakan algoritma Naive Bayes dan SVM, yang masing-masing memisahkan data menjadi segmen *data training* (melatih model) dan *data testing* (evaluasi hasil). Dalam penelitian ini, tiga skenario pengujian dilaksanakan pada setiap algoritma dengan memvariasikan persentase pembagian *data training* dan *data testing* seperti ditunjukkan pada Tabel 1.

Tabel 1. Pembagian Data Klasifikasi

Skenario	*Data Training*	*Data Testing*
1	70%	30%
2	80%	20%
3	90%	10%

2.4.1.Naive Bayes

Algoritma Naive Bayes didasarkan pada teorema Bayes, yang dikembangkan oleh Thomas Bayes pada abad ke 18 [18]. Algoritma Naive Bayes merupakan suatu metode klasifikasi yang dapat dipakai untuk memperkirakan probabilitas keanggotaan suatu kelas tertentu, dengan dasar pada teorema Bayes [19]. Algoritma ini termasuk dalam *Supervised Learning* karena membutuhkan *data training* sebelum proses klasifikasi. Secara umum rumus dasar persamaan teorema *Naive Bayes* disajikan pada persamaan (2).

$$P(H|X) = \frac{P(H)x\ P(H)}{P(X)} \quad (2)$$

dimana:

X : Data dengan kelas yang belum diketahui
H : Hipotesis X merupakan suatu kelas spesifik
P(H|X) : Probabilitas H berdasarkan kondisi X
P(H) : Probabilitas pada hipótesis H (prior)
P(X|H) : Probabilitas X berdasarkan kondisi H
P(X) : Probabilitas X (data sampel yang diamati)

2.4.2. *Support Vector Machines* (SVM)

Algoritma SVM digunakan untuk menemukan *hyperplane* yang terbaik dengan memaksimalkan jarak antara setiap kelas [20]. *Hyperplane* ini dipilih sedemikian rupa sehingga jarak (margin) antara *hyperplane* dengan titik data terdekat dari setiap kelas (S*upport Vectors*) adalah maksimal [21]. Untuk menjamin akurasi dan *reproducibility*, dilakukan optimasi *hyperparameter* melalui metode *Grid Search* dengan 5 *Fold Cross Validation*. Detail konfigurasi yang diterapkan meliputi variasi parameter regularisasi C (0.1, 1, 10) untuk mengontrol kesalahan klasifikasi, serta penggunaan kernel Linear dan *Radial Basis Function* (RBF) untuk menangani kompleksitas distribusi data. Selain itu, parameter Gamma diatur dengan opsi *scale* dan *auto* untuk mengatur jangkauan pengaruh titik data dalam ruang fitur. Mekanisme SMOTE digunakan sebelum pelatihan guna memastikan model tidak bias terhadap kelas mayoritas. Dalam SVM, rumus untuk memprediksi kelas dari suatu sampel x menggunakan persamaan (3).

$$f(x) = sign(w * x + b) \quad (3)$$

dimana:

f (x)	:	Fungsi keputusan yang mengklasifikasikan sampel x
w	:	Vektor bobot
x	:	Vektor fitur input
b	:	Bias
sign(-)	:	Fungsi tanda yang menghasilkan *Label* kelas

2.5. Evaluasi Model

Pada tahap evaluasi digunakan matrik evaluasi untuk mengevaluasi kinerja modeling dari sistem. Matrik pengukuran yang digunakan adalah akurasi, presisi*, recall, dan F1-Score*. Akurasi dihitung berdasarkan jumlah prediksi yang benar dibagi dengan total seluruh data yang dievaluasi. Akurasi menunjukkan proporsi total klasifikasi yang benar dari semua kasus. Presisi berfokus pada seberapa sering model benar ketika memprediksi kelas positif. *Recall* mengukur kemampuan model untuk menemukan semua instansi kelas positif yang sebenarnya ada. Terakhir, *F1-Score* adalah rata-rata harmonik dari Presisi dan *Recall*. *F1-Score* memberikan satu metrik gabungan yang seimbang [22]. Setiap matrik evaluasi ini memiliki penilaian tersendiri yang digunakan untuk mengukur berbagai hal tentang bagaimana sebaiknya model tersebut [23]. Rumus perhitungan Akurasi, Presisi, *Recall*, dan *F1-Score* ditunjukkan pada persamaan (4) sampai persamaan (7).

$$Accuracy = \frac{TP+TN}{TP+FN+FP+TN} \quad (4)$$

$$Precision = \frac{TP}{TP+FP} \quad (5)$$

$$Recall = \frac{TP}{TP+FN} \quad (6)$$

$$F1 - Score = \frac{2TP}{2TP+FN+FP} \quad (7)$$

2.6. Korelasi Pearson

Penelitian ini menerapkan Uji Korelasi Pearson *Product Moment* (r) untuk mengukur tingkat keeratan hubungan antar variabel, yang hasilnya diwakili oleh koefisien korelasi (r) [24]. Analisis korelasi Pearson mempelajari informasi tentang hubungan penggunaan media visual, dan untuk mengetahui apakah variabel X dan variabel Y memiliki keterkaitan yang signifikan [25]. Persamaan (8) menggambarkan rumus korelasi Pearson.

$$r = \frac{N.\sum XY - (\sum X).(\sum Y)}{\sqrt{[N.\sum X^2 - (\sum X)^2][N\sum Y^2 - (\sum Y)^2]}} \quad (8)$$

dimana:

r	:	Angka indeks korelasi antara variabel X dengan variabel Y
N	:	Jumlah Sampel
$\sum X^2$	:	Jumlah kuadrat variabel X
$\sum Y^2$	:	Jumlah kuadrat variabel Y
$\sum XY$	:	Jumlah hasil perkalian antara skor X dan skor Y
$\sum X$	:	Jumlah variabel X
$\sum Y$	:	Jumlah variabel Y

3. HASIL DAN PEMBAHASAN

3.1. Pengumpulan Data

Data ulasan pengguna Aplikasi OTT diakuisisi dari *Google Play Store* menggunakan teknik *web scraping* dengan menargetkan 1000 ulasan terbaru berbahasa Indonesia untuk setiap Aplikasi OTT. Tujuan utama dari pengumpulan data ini adalah untuk mendapatkan variabel teks ulasan yang diklasifikasikan ke dalam tiga jenis sentimen, yaitu sentimen positif, negatif, dan netral. Klasifikasi ini dilakukan guna mengevaluasi persepsi pengguna terhadap layanan OTT secara spesifik, yang kemudian menjadi parameter dalam mengukur tingkat kepuasan pelanggan pada masing-masing platform. Contoh hasil *web scraping* seperti pada Tabel 2. Pada Tabel 2 terlihat contoh data yang dikumpulkan diantaranya meliputi *ReviewId* yang merupakan *Id* unik ulasan*, UserName* yang berisi nama pengguna yang memberikan ulasan*, Content* yang berisi ulasan pengguna, *Score* yang merepresentasikan tingkat kepuasan. Kemudian *ThumbsUpCount* adalah total ulasan disukai, dan *ReviewCreatedVersion* yang merupakan versi aplikasi saat ulasan dibuat.

Tabel 2. Contoh Hasil *Text Preprocessing*

reviewId	*userName*	*userImage*	*content*	*score*	*thumbsUpCount*	*reviewCreatedVersion*	*at*	*replyContent*	*repliedAt*	*appVersion*
abb9...	Fachri	https://pla...	Saya tidak bisa menonton.	1	635	3.0	2025	None	None	3.0

3.2. Proses *Text Preprocessing*

Tahapan *text preprocessing* meliputi seleksi kolom, pelabelan, *cleaning*, *case folding*, *stopword removal*, *tokenizing*, dan *stemming*. Contoh transformasi ulasan mentah menjadi fitur siap analisis disajikan pada Tabel 3. Kategori Label ditetapkan berdasarkan *Score* sebagai berikut: Negatif (S*core* 1 dan 2), Netral (*Score* 3), dan Positif (*Score* 4 dan 5). Selanjutnya proses *case folding* dan *cleaning* pada kolom *Text Clean*, proses *stopword* pada kolom *Text Stopword*, proses *tokenizing* pada kolom *Text Tokens*, dan terakhir proses *stemming pada kolom Text Stemming*.

Tabel 3. Contoh Hasil *Text Preprocessing*

Content	*Score*	*Label*	*Text Clean*	*Text Stopword*	*Text Tokens*	*Text Stemming*
Saya tidak bisa menonton siaran Sepak Bola,	1	Negatif	saya tidak bisa menonton siaran sepak bola	menonton siaran sepak bola	['menonton', 'siaran', 'sepak', 'bola',]	tonton siar sepak bola

3.3. Transformasi

Data ulasan yang telah melalui *Text Preprocessing* diolah menjadi vektor numerik menggunakan metode TF-IDF. Pembobotan ini memberikan nilai yang lebih tinggi pada kata-kata spesifik yang diskriminatif terhadap sentimen. Sementara kata-kata umum diberi bobot rendah.

3.4. Klasifikasi

Klasifikasi dilakukan untuk menentukan algoritma dan skenario terbaik berdasarkan rata-rata Akurasi, Presisi, *Recall*, dan *F1-Score* dari seluruh data aplikasi OTT seperti pada Tabel 4.

Tabel 4. Perbandingan Kinerja Algoritma

Algoritma	Skenario	Akurasi (rata-rata)	Presisi (rata-rata)	*Recall* (rata-rata)	*F1-Score* (rata-rata)
Naive Bayes	1	71%	0,66	0,71	0,62
	2	71%	0,66	0,71	0,62
	3	73%	0,67	0,73	0,65
SVM	1	71%	0,71	0,71	0,71
	2	73%	0,71	0,74	0,71
	3	75%	0,72	0,75	0,73

Pada Tabel 4, metode SVM skenario 3 menghasilkan nilai akurasi rata-rata mencapai 75%. Akurasi ini adalah akurasi tertinggi dalam seluruh pengujian, menunjukkan bahwa 75% dari total prediksi yang dilakukan oleh model adalah benar. Pada metode SVM skenario 3, menghasilkan Presisi rata-rata mencapai 0,72. Nilai Presisi ini menunjukkan kemampuan

model untuk menghindari *False Positives*. Selain itu pada skenario ini, *Recall* rata-rata mencapai 0,75. Nilai *Recall* ini merupakan yang tertinggi dalam pengujian, menunjukkan kemampuan model untuk mengidentifikasi sebagian besar kasus positif yang sebenarnya (menghindari *False Negatives*). Nilai *F1-Score* rata-rata pada SVM skenario 3 mencapai 0,73. Nilai tertinggi ini menunjukkan keseimbangan terbaik antara Presisi dan *Recall* yang dicapai oleh model SVM pada skenario 3.

3.5. Evaluasi

Kinerja model diuji menggunakan metrik Akurasi, Presisi, *Recall*, dan *F1-Score* pada setiap kelas sentimen seperti pada Tabel 5. Pada Tabel 5 menunjukkan Prime Video meraih akurasi tertinggi (78%), dengan kinerja superior pada sentimen negatif (F1-Score 0,91). Sebaliknya, Disney+ dan Catchplay+ mencatat Akurasi terendah yaitu 73%. Secara umum, nilai *F1-Score* di atas 0,78 pada kelas negatif menunjukkan model sangat unggul dalam mendeteksi sentimen negatif. Namun model gagal mengklasifikasikan kelas netral di seluruh aplikasi (F1-Score 0,00 pada Vidio dan Catchplay+). Terkait sentimen positif, Catchplay+ mencatat *Recall* tertinggi (0,80) yang menunjukkan model sangat efektif mengenali ulasan positif, sedangkan Vidio mencatat *Recall* terendah (0,36) yang mengindikasikan banyaknya ulasan positif yang tidak terdeteksi.

Tabel 5. Kinerja Model Optimal

Aplikasi OTT	Kelas Sentimen	Presisi	*Recall*	*F1-Score*	Akurasi
Netflix	Positif	0,44	0,40	0,42	77%
	Netral	0,17	0,11	0,13	
	Negatif	0,85	0,89	0,87	
Vidio	Positif	0,73	0,36	0,48	75%
	Netral	0,00	0,00	0,00	
	Negatif	0,75	0,97	0,85	
Disney+	Positif	0,63	0,55	0,59	73%
	Netral	0,22	0,15	0,18	
	Negatif	0,82	0,91	0,86	
Prime Video	Positif	0,50	0,33	0,40	78%
	Netral	0,07	0,20	0,11	
	Negatif	0,94	0,88	0,91	
Catchplay+	Positif	0,69	0,80	0,74	73%
	Netral	0,00	0,00	0,00	
	Negatif	0,77	0,78	0,78	

3.6. Analisis Korelasi Pearson

Analisis Korelasi Pearson dilakukan untuk memvalidasi hubungan kuantitatif antara kinerja model sentimen yang diwakili oleh X (*F1-Score* positif) dan Y (Metrik *rating* pada *Google Play Store)* seperti pada Tabel 6.

Tabel 6. Analisa Korelasi Pearson

Aplikasi	X (*F1-Score* Positif)	Y (*Rating*)
Netflix	0,42	3,6
Vidio	0,48	4,4
Disney+	0,59	3,2
Prime Video	0,40	3,4
Catchplay+	0,74	4,7

Hasil perhitungan koefisien korelasi Pearson pada kelima Aplikasi OTT menunjukkan nilai r = 0,538. Nilai ini mengindikasikan korelasi positif sedang antara kemampuan model dalam mengklasifikasikan sentimen positif dengan tingkat kepuasan pengguna yang terwujud dalam *rating* Aplikasi.

4. KESIMPULAN

Penelitian ini berhasil mengidentifikasi algoritma optimal untuk klasifikasi sentimen ulasan Aplikasi OTT serta memvalidasi hubungannya sentimen positif dengan *rating* aplikasi di *Google Play Store*. SVM dengan skenario 3 ditetapkan sebagai model optimal, karena mencapai Akurasi rata-rata 75% dan kinerja superior dibandingkan Naive Bayes. Secara kuantitatif, analisis korelasi Pearson menunjukkan adanya korelasi positif sedang r = 0,538 antara kinerja sentimen positif dengan *rating* aplikasi. Hal menegaskan hubungan searah dengan persepsi kepuasan pengguna. Namun, model ini menunjukkan kelemahan signifikan dalam mengidentifikasi sentimen netral dengan *F1-Score* 0,00 pada aplikasi Vidio dan Catchplay+, serta berkisar antara 0,00 hingga 0,18 pada aplikasi lainnya. Oleh karena itu, penelitian lanjutan sangat diperlukan untuk berfokus pada mitigasi masalah data *imbalance* dan ambiguitas leksikal, melalui implementasi teknik sampling lanjutan atau modifikasi arsitektur fitur, guna menangani kelas minoritas netral secara efektif.

DAFTAR PUSTAKA

[1] D. Muhammad, T. S. Ramli, and R. R. Permata, “Tanggung Jawab Hukum *Over The Top* News Aggregator Terhadap Pelanggaran Hak Ekonomi Produk Jurnalistik,” *Media Huk. Indones.*, vol. 2, no. 6, pp. 19–30, 2025, doi: 10.5281/zenodo.15375277.

[2] L. Z. Valentine, “Analisis Perspektif Regulasi *Over The Top* di Indonesia dengan Pendekatan Regulatory Impact Analysis,” *J. Telekomun. dan Komput.*, vol. 8, no. 3, p. 222, 2018, doi: 10.22441/incomtech.v8i3.5675.

[3] H. A. Ramadhan, B. M. Purwaamijaya, and R. G. Guntara, “Pengaruh User Experience terhadap Customer Satisfaction pada Aplikasi Seluler Streaming Vidio,” *JTIM J. Teknol. Inf. dan Multimed.*, vol. 5, no. 2, pp. 122–133, 2023, doi: 10.35746/jtim.v5i2.367.

[4] D. Shafira, A. Rizki, M. S. Khabib, N. Rahmayuna, and G. Utomo, “Klasifikasi Sentimen Ulasan Pengguna Aplikasi Layanan Publik *Google Play Store* Menggunakan NLP dan ML,” vol. 20, no. 1, pp. 51–64, doi: 10.33365/jtk.v20i1.586.

[5] M. E. Suprapto, A. Q., Irwansyah, I., & Irfandianto, “Analisis Dinamika Ulasan

Penggunaan Aplikasi Vidio Dan Viu," *J. Ilmu Komun. UHO J. Penelit. Kaji. Ilmu Komun. dan Inf.*, vol. 10, no. 1, pp. 67–78, 2024, doi: 10.52423/jikuho.v10i1.462.

[6] R. R. S. Dwi Amelia, Pratomo Setiaji, "Perbandingan Algoritma SVM dan Naive Bayes dalam Klasifikasi Sentimen pada Ulasan Aplikasi Traveloka dan Agoda," vol. 11, no. 2, pp. 263–270, 2025, doi: 10.26418/jp.v11i2.96869.

[7] A. I. Tanggraeni and M. N. N. Sitokdana, "Analisis Sentimen Aplikasi E-Government pada Google Play Menggunakan Algoritma Naïve Bayes," *JATISI (Jurnal Tek. Inform. dan Sist. Informasi)*, vol. 9, no. 2, pp. 785–795, 2022, doi: 10.35957/jatisi.v9i2.1835.

[8] D. Triully Prasetyo and Atiqah Meutia Hilda, "Metode Support Vector Machine Untuk Analisis Sentimen Aplikasi Threads di *Google Play Store*," *Indones. J. Comput. Sci.*, vol. 13, no. 5, pp. 76–83, 2024, doi: 10.33022/ijcs.v13i5.4446.

[9] O. M. Wulandari, I. Maulana, F. Syamsudin, and R. Waluyo, "Perbandingan Algoritma Naive Bayes dan SVM dalam Analisis Sentimen Twitter terhadap Isu Ijazah Jokowi Palsu," *J. Manaj. Inform. Sist. Inf. dan Teknol. Komput.*, vol. 4, no. 1, pp. 392–400, 2025, doi: 10.70247/jumistik.v4i1.145.

[10] N. Khoirunnisaa, K. Nabila Nastiti Kesuma, S. Setiawan, and A. Yunizar Pratama Yusuf, "Klasifikasi Teks Ulasan Aplikasi Netflix Pada *Google Play Store* Menggunakan Algoritma Naïve Bayes Dan SVM," *SKANIKA Sist. Komput. dan Tek. Inform.*, vol. 7, no. 1, pp. 64–73, Jan. 2024, doi: 10.36080/skanika.v7i1.3138.

[11] U. M. Riau, "Pengenalan Dasar Pemrograman Python Dengan Google Colaboratory Basic," no. 1, 2024, doi: 10.55606/jppmi.v3i1.1087.

[12] S. Suliman, "Implementasi Data Mining Terhadap Prestasi Belajar Mahasiswa Berdasarkan Pergaulan dan Sosial Ekonomi Dengan Algoritma K-Means Clustering," *Simkom*, vol. 6, no. 1, pp. 1–11, 2021, doi: 10.51717/simkom.v6i1.48.

[13] A. Nurian, "Analisis Sentimen Ulasan Pengguna Aplikasi Google Play Menggunakan Naïve Bayes," *J. Inform. dan Tek. Elektro Terap.*, vol. 11, no. 3s1, pp. 829–835, Sep. 2023, doi: 10.23960/jitet.v11i3s1.3348.

[14] E. Eskiyaturrofikoh and R. R. Suryono, "Analisis Sentimen Aplikasi X Pada *Google Play Store* Menggunakan Algoritma Naïve Bayes Dan Support Vector Machine (SVM)," *JIPI (Jurnal Ilm. Penelit. dan Pembelajaran Inform.*, vol. 9, no. 3, pp. 1408–1419, Aug. 2024, doi: 10.29100/jipi.v9i3.5392.

[15] S. Kusumo, "Penerapan Web Scraping Deskripsi Produk Menggunakan Selenium Python Dan Framework Laravel," *JATISI (Jurnal Tek. Inform. dan Sist. Informasi)*, vol. 9, no. 4, pp. 3426–3435, Dec. 2022, doi: 10.35957/jatisi.v9i4.2727.

[16] I. T. Julianto, "Analisis Sentimen Terhadap Sistem Informasi Akademik Institut Teknologi Garut," *J. Algoritm.*, vol. 19, no. 1, pp. 449–456, 2022, doi: 10.33364/algoritma/v.19-1.1112.

[17] R. R. Salam, M. F. Jamil, Y. Ibrahim, R. Rahmaddeni, S. Soni, and H. Herianto, "Analisis Sentimen Terhadap Bantuan Langsung Tunai (BLT) Bahan Bakar Minyak (BBM) Menggunakan Support Vector Machine," *MALCOM Indones. J. Mach. Learn. Comput. Sci.*, vol. 3, no. 1, pp. 27–35, 2023, doi: 10.57152/malcom.v3i1.590.

[18] B. A. Maulana, M. J. Fahmi, A. M. Imran, and N. Hidayati, "Analisis Sentimen Terhadap Aplikasi Pluang Menggunakan Algoritma Naive Bayes dan Support Vector Machine

(SVM)," *MALCOM Indones. J. Mach. Learn. Comput. Sci.*, vol. 4, no. 2, pp. 375–384, Feb. 2024, doi: 10.57152/malcom.v4i2.1206.

[19] A. Jalu *et al.*, "Microsoft Word - 20. Atmaja Jalu Narendra Kisma, Chyntia Raras Ajeng Widiawati, Suliswaningsih 174-," *J. Tek. Inform. dan Sist. Inf.*, vol. 10, no. 2, pp. 174–184, 2023, doi: 10.35957/jatisi.v10i2.4784.

[20] M. Iqbal, M. Afdal, and R. Novita, "Implementasi Algoritma Support Vector Machine Untuk Analisa Sentimen Data Ulasan Aplikasi Pinjaman Online di *Google Play Store*," *MALCOM Indones. J. Mach. Learn. Comput. Sci.*, vol. 4, no. 4, pp. 1244–1252, Jul. 2024, doi: 10.57152/malcom.v4i4.1435.

[21] D. Fajar Nawulansih, N. Ceisa Santi, and I. Aristia Sa'ida, "Analisis Sentimen Ulasan Aplikasi DANA di *Google Play Store*: Penerapan Support Vector Machine dan Synthetic Minority Over-sampling Technique," *J. Pendidik. dan Teknol. Indones.*, vol. 5, no. 9, pp. 2660–2671, Sep. 2025, doi: 10.52436/1.jpti.1053.

[22] M. Fadli and R. A. Saputra, "Klasifikasi Dan Evaluasi Performa Model Random Forest Untuk Prediksi Stroke," *JT J. Tek.*, vol. 12, no. 02, pp. 72–80, 2023, doi: 10.31000/jt.v12i2.9099.

[23] Z. A. W. Sugandi and S. Sarmini, "Implementasi Metode Support Vector Machine (SVM) Pada Sistem Rekomendasi Produk Perawatan Wajah Berbasis Web," *J. Sist. dan Teknol. Inf.*, vol. 12, no. 3, p. 388, 2024, doi: 10.26418/justin.v12i3.74905.

[24] S. F. Sihotang, F. S. W. Harahap, and M. R. Mazaly, "Pendekatan Korelasi Pearson Dan Spearman Dalam Menganalisis Hubungan Penggunaan Bahan Ajar Berbasis Daring Dan Prestasi Akademik Mahasiswa," *MES J. Math. Educ. Sci.*, vol. 10, no. 1, pp. 270–276, Oct. 2024, doi: 10.30743/mes.v10i1.10204.

[25] P. Jambi, "Multi Proximity : Jurnal Statistika Universitas Jambi Analisis Korelasi Pearson Jumlah Penduduk dengan Jumlah Kendaraan Bermotor di Kepadatan penduduk dan jumlah kendaraan bermotor merupakan dua faktor yang saling variabel tersebut . Salah satu metode sta," vol. 2, no. 1, pp. 39–44, 2023, doi: 10.22437/multiproximity.v2i1.25568.