Analisis Sentimen Ulasan Game Genshin Impact Menggunakan PCA dan SVM

Gemma Dwi Prasetya[1], Anita Qoiriah[2]

[1,2] Teknik Informatika, Universitas Negeri Surabaya
[1]gemma.19051@mhs.unesa.ac.id
[2]anitaqoiriah@unesa.ac.id

***Abstrak*— Genshin Impact adalah RPG yang sukses yang menarik banyak ulasan di Google Play Store, yang sentimennya menjadi indikator informatif bagi pengembang aplikasi. Penelitian ini mengeksplorasi klasifikasi sentimen pengguna dengan penerapan model *Support Vector Machine* (SVM) yang terintegrasi dengan ekonomisasi sumber daya dari *Principal Component Analysis* (PCA) untuk meningkatkan kecepatan reduksi dimensi algoritma. Ulasan untuk proyek ini bersumber dari Analisis Sentimen Ulasan Genshin Impact yang dihosting di Kaggle. Dataset berisi 944 ulasan, yang terdiri dari 510 ulasan yang diidentifikasi sebagai POSITIF dan 434 ulasan yang diidentifikasi sebagai NEGATIF, dengan sentimen yang ditetapkan selama pengunggahan dataset. Penelitian ini memanfaatkan model *Knowledge Discovery in Databases* (KDD) dan mencakup *Preprocessing*, TF-IDF, pemilihan fitur Chi-Square, PCA, penyeimbangan kelas (SMOTE), dan SVM dengan empat jenis kernel dan permutasi parameter untuk 1.440 eksperimen yang diharapkan dan dievaluasi pada tiga split dataset yang berbeda. Hasil eksperimen menunjukkan bahwa kernel RBF, setelah penerapan pemilihan fitur Chi-Square, PCA, dan SMOTE, mencapai hasil terbaik dengan akurasi 93,68% dan F1-Score 94,12% untuk split 90:10, dengan hasil yang lebih baik dibandingkan rasio split 80:20 dan 70:30. Dari hasil tersebut, SVM dengan PCA secara bersamaan mencapai klasifikasi sentimen permainan yang efektif dengan akurasi tinggi, dan hasil eksperimen menunjukkan bahwa pemilihan fitur Chi-Square meningkatkan hasil model. Penelitian selanjutnya dengan metodologi ini harus mencakup penggunaan embedding kata kontekstual dan meningkatkan ukuran dataset untuk memungkinkan peningkatan ketahanan dan generalisasi model.**

***Kata Kunci*— Analisis Sentimen, Genshin Impact, *Support Vector Machine*, *Principal Component Analysis*, Chi-Square, SMOTE**

I. PENDAHULUAN

Sejak dirilis pada tahun 2020 oleh HoYoverse, Genshin Impact, sebuah role-playing game (RPG) bertema dunia terbuka, telah diunduh ratusan juta kali dan terus menarik pemain baru hingga saat ini. Popularitas sebesar itu meninggalkan jejak berupa banyak sekali ulasan pemain di Google Play Store, baik yang memuji maupun mengeluhkan aspek tertentu dari permainan; jejak inilah yang sebenarnya bisa dimanfaatkan pengembang sebagai masukan apabila dianalisis secara terstruktur. Secara umum, analisis sentimen adalah teknik pengolahan teks untuk menangkap kecenderungan emosi suatu tulisan dan mengklasifikasikannya ke dalam kategori positif maupun negatif; teknik ini juga umum dipakai untuk memetakan seberapa sering kata-kata tertentu muncul dalam suatu kumpulan ulasan [1]. Karena jumlah ulasan terus bertambah setiap harinya, membacanya satu per satu menjadi tidak lagi realistis, sehingga diperlukan mekanisme klasifikasi otomatis berbasis pembelajaran mesin.

Analisis sentimen ulasan Genshin Impact berfokus pada berbagai aspek karena merupakan topik yang sudah banyak diteliti. Contohnya, studi [2] meneliti algoritma Naïve Bayes dan Support Vector Machine (SVM) dan menentukan bahwa SVM lebih cocok untuk tugas klasifikasi. Studi [3] menggunakan Naïve Bayes Classifier untuk menganalisis ulasan di Play Store dan memperoleh hasil yang memadai, sementara studi [4] memfokuskan pada platform X (Twitter) dan menganalisis sentimen ulasan Genshin Impact terkait kesehatan mental pemain dengan menggunakan kombinasi TF-IDF dan SVM. Studi [5] juga membandingkan Naïve Bayes dan SVM; perbedaannya, ulasan game yang digunakan dalam studi ini sama. Sayangnya, tidak satu pun dari studi tersebut melaporkan penggunaan teknik reduksi dimensi seperti Principal Component Analysis (PCA). Representasi fitur TF-IDF umumnya berkali-kali memiliki tingkat dimensionalitas yang tinggi, mencapai ribuan fitur, yang menyebabkan overfitting dan membuat model tidak efisien.

Di sisi lain, temuan penelitian [6] menunjukkan bahwa penambahan PCA pada klasifikasi sentimen berbasis SVM justru mampu mendongkrak performa dibandingkan tanpa reduksi dimensi sama sekali. Demikian juga, studi [7] menunjukkan bahwa untuk data berdimensi tinggi, Random Forest dan PCA memiliki hasil yang lebih baik daripada Random Forest tanpa PCA. Kedua hasil ini sama-sama mengarah pada satu kesimpulan: reduksi dimensi punya potensi mendongkrak performa klasifikasi pada data teks berdimensi tinggi. Namun, penggunaan SMOTE untuk penyeimbangan kelas, bersama PCA dan pemilihan fitur Chi-Square, serta data ulasan dari Genshin Impact, tidak umum dilakukan.

Mengisi kekosongan ini, studi ini bertujuan untuk mengeksplorasi sentimen dari ulasan game Genshin Impact melalui integrasi Analisis Komponen Utama (PCA), Support Vector Machine (SVM), pemilihan fitur Chi-Square, keseimbangan kelas dengan SMOTE, dan penyeimbangan dengan SMOTE. Studi ini merupakan upaya untuk meningkatkan pengurangan dimensi data secara optimal dengan pengklasifikasian sentimen positif dan negatif yang dicapai dengan ketelitian yang lebih besar. Hal ini dilakukan melalui pelaksanaan 1.440 kombinasi eksperimen yang berbeda.

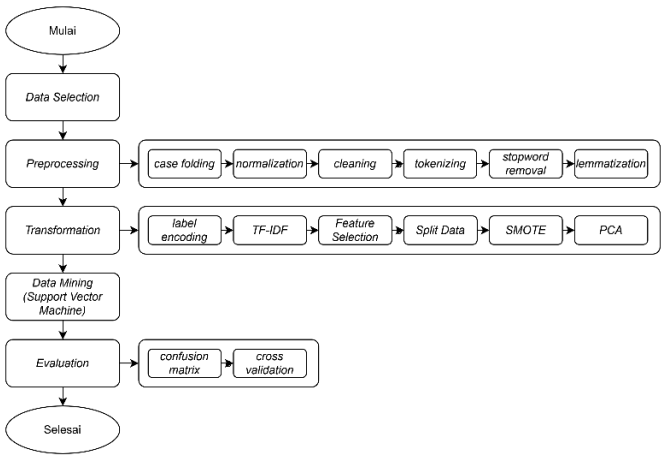
Kontribusi pengetahuan yang ditawarkan oleh studi ini adalah integrasi pemilihan fitur Chi-Square, PCA untuk

pengurangan dimensi, SMOTE untuk keseimbangan kelas, dan SVM untuk pengklasifikasian, yang digabungkan untuk pertama kalinya dalam kerangka penelitian tunggal untuk menganalisis ulasan game dari perspektif analisis sentimen. Penekanan diberikan pada pelaksanaan sistematis dari 1.440 kombinasi eksperimen, untuk mencapai konfigurasi terbaik, dan bukan hasil dari keberuntungan. Selain itu, fokus khusus diberikan pada akurasi dan efektivitas kerangka penelitian, terutama untuk implementasi skala besar di bidang analisis sentimen.

Riset seputar analisis sentimen ulasan Genshin Impact memang sudah mulai bertumbuh. Selain studi-studi [2], [3], [4], dan [5], penggunaan PCA dan SVM untuk klasifikasi telah ditunjukkan dalam penelitian lain, termasuk ulasan tentang situs web aplikasi kesehatan pemerintah [10] dan prediksi data berdimensi tinggi dengan Random Forest [7]. Keputusan dalam penelitian ini untuk menggunakan Google Play Store sebagai sumber data disebabkan oleh fakta bahwa, sebagai platform distribusi resmi aplikasi Android, ulasan di Google Play Store diposting melalui akun Google yang terverifikasi. Hal ini berbeda dengan sumber ulasan aplikasi lainnya, yang mungkin berisi ulasan anonim dan kurang otentik, sehingga kurang dapat diandalkan.

II. METODE PENELITIAN

Penelitian ini didasarkan pada model Knowledge Discovery in Databases (KDD), yang bertujuan untuk ekstraksi wawasan yang signifikan dan dapat ditindaklanjuti dari volume data besar. Kerangka ini terdiri atas lima tahap yang dijalankan berurutan, yakni data selection, preprocessing, transformation, data mining, dan evaluation [8], sebagaimana digambarkan pada Gambar. 1.



Gambar. 1 Alur Diagram KDD

Lima tahapan yang terlihat dalam Gambar 1 dijalankan secara berurutan, dengan tahap penggalian data mencakup 1.440 variasi (pemilihan fitur, pemartisian data, struktur PCA-SMOTE, konfigurasi kernel SVM, dll.) sebelum mencapai tahap evaluasi.

A. Data Selection

Dataset yang digunakan pada penelitian ini adalah Sentiment Analysis Genshin Impact Reviews yang bersumber dari platform Kaggle, mencakup 944 ulasan berbahasa Inggris (510 berlabel POSITIVE dan 434 berlabel NEGATIVE) yang dihimpun dalam rentang Januari–Maret 2024. Label sentimen pada dataset ini telah disematkan oleh pihak penyedia data. Untuk penelitian ini, hanya kolom berisi teks ulasan dan label sentimen yang digunakan, sementara rating ulasan, tanggal, dan kolom lainnya diabaikan karena studi ini bertujuan menganalisis hanya teks ulasan.

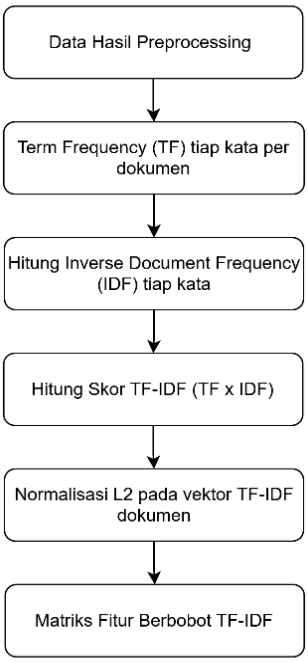
B. Preprocessing

Sebelum data siap masuk ke tahap ekstraksi fitur, teks ulasan lebih dulu dibersihkan dan dirapikan melalui enam langkah preprocessing berikut.

1) *Case Folding:* mengubah semua huruf pada teks ulasan menjadi huruf kecil agar konsisten.
2) *Normalization:* mengoreksi kata-kata tidak baku maupun bentuk singkatan ke bentuk yang lebih standar.
3) *Cleaning:* Langkah analisis sentimen ini menghilangkan data yang tidak relevan dan meliputi URL, tanda baca, angka, dan karakter arbitrer lainnya.
4) *Tokenizing:* membagi setiap kalimat ulasan menjadi satuan-satuan kata (token).
5) *Stopword Removal:* Langkah ini menggunakan pustaka NLTK untuk menghapus kata-kata umum yang tidak memiliki nilai sentimen.
6) *Lemmatization:* memanfaatkan WordNetLemmatizer agar tiap kata kembali ke bentuk dasarnya

C. Transformation

Label sentimen diubah menjadi bentuk numerik melalui Label Encoding, di mana 1 melambangkan kategori POSITIF dan 0 menunjukkan kategori NEGATIF. Kemudian, fitur diambil berdasarkan pendekatan Term Frequency-Inverse Document Frequency (TF-IDF) [9]. Sebagai referensi, pendekatan ini menghasilkan 3.424 fitur unik dalam studi ini. Pendekatan TF-IDF memberikan bobot berdasarkan frekuensi kata target (t) ditemukan dalam dokumen tertentu (d). Ini ditunjukkan dalam (1). Komponen lain dari persamaan TF-IDF melihat pada jumlah total dokumen (N) dan jumlah dokumen yang mengandung kata target (df(t)). Gambar 2 menunjukkan proses perhitungan TF-IDF secara lengkap.

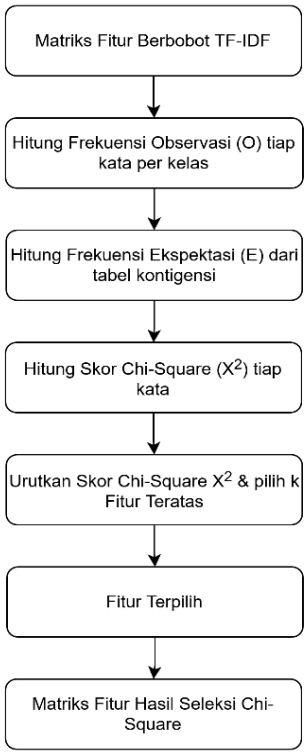


Gambar. 2 Diagram Tahapan Perhitungan TF-IDF

Proses ini seperti yang digambarkan dalam Gambar 2 dimulai dengan data yang telah diproses sebelumnya dan kemudian menghitung *Term Frequency* (TF) dan *Inverse Document Frequency* (IDF) untuk setiap kata. Hasil dari TF dan IDF adalah skor TF-IDF. Skor ini kemudian dinormalisasi (L2) per dokumen untuk membuat matriks fitur berbobot akhir.

$$\text{TF-IDF}(t,d) = \text{tf}(t,d) \times \log(N / \text{df}(t)) \quad (1)$$

Setelah bobot TF-IDF diperoleh, seleksi fitur dilanjutkan dengan Chi-Square [10], sebuah uji statistik yang mengukur seberapa independen suatu fitur terhadap kelas targetnya. Dari 3.424 fitur yang tersedia, penelitian ini menetapkan k=1.000 fitur dengan nilai χ^2 tertinggi. Skor χ^2 sendiri dihitung memakai persamaan (2), di mana Oi adalah frekuensi observasi dan Ei adalah frekuensi ekspektasi pada kategori ke-i; alur lengkapnya tampak pada Gambar. 3.

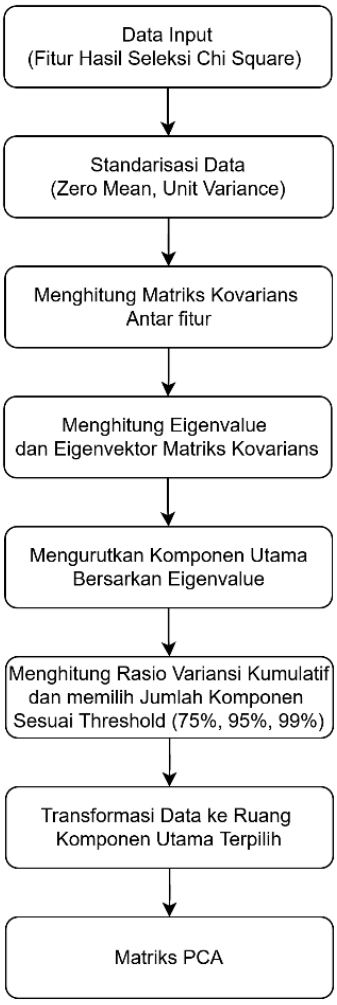


Gambar. 3 Diagram Tahapan Seleksi Fitur Chi-Square

Gambar 3 menggambarkan urutan operasi berikut: pembuatan awal matriks fitur berbobot TF-IDF, penentuan frekuensi yang diamati dan diharapkan dari kata berdasarkan kelas dan pembuatan tabel kontingensi, perhitungan skor χ^2, dan akhirnya, pengurutan serta pemilihan k fitur dengan skor tertinggi.

$$\chi^2 = \Sigma (Oi - Ei)^2 / Ei \quad (2)$$

Data yang sudah melewati seleksi fitur kemudian dibagi menjadi tiga skema rasio split (90:10, 80:20, 70:30). Setiap metode menggunakan Synthetic Minority Oversampling Method (SMOTE) [11] untuk membuat data sintetis dengan menginterpolasi tetangga terdekat dari kelas minoritas guna mengatasi ketidakseimbangan kelas. PCA [12] kemudian digunakan untuk mengurangi dimensi. PCA dilakukan pada tiga tingkat varians kumulatif yang berbeda (75%, 95%, 99%) baik sebelum maupun setelah SMOTE, menghasilkan empat arsitektur data berbeda untuk setiap pembagian data. Gambar 4 menyajikan skema reduksi dimensi tersebut.



Gambar. 4 Diagram Tahapan Principal Component Analysis (PCA)

Proses PCA yang tergambar pada Gambar. 4 diawali dari matriks fitur hasil Chi-Square, lalu data distandardisasi, matriks kovarians beserta eigenvalue dan eigenvector-nya dihitung, jumlah komponen utama ditentukan berdasarkan ambang variansi kumulatif yang dipilih, dan akhirnya data diproyeksikan ke ruang komponen utama yang baru.

D. Machine Learning

Klasifikasi pada tahap ini menggunakan Support Vector Machine (SVM) [13], algoritma yang bekerja dengan mencari hyperplane paling optimal untuk memisahkan dua kelas dalam ruang fitur. Fungsi keputusannya dirumuskan pada persamaan (3): w adalah vektor bobot, b adalah bias, sementara fungsi sign() menentukan apakah data diklasifikasikan sebagai POSITIVE (+1) atau NEGATIVE (-1).

$$f(x) = \text{sign}(w \cdot x + b) \quad (3)$$

Studi ini mempertimbangkan empat kernel (Linear, RBF, Polynomial, dan Sigmoid) dan parameter yang berbeda (C ∈ {0, 1; 1; 10}) dan (gamma ∈ {0.001; 0.1; 1}). Berbeda dengan pencarian grid otomatis, pengujian menyeluruh dilakukan untuk setiap kombinasi kernel-parameter (ekstensif), termasuk berbagai arsitektur (dengan/tanpa PCA dan SMOTE), segmentasi, dan pemilihan fitur (dengan/tanpa Chi-Square). Sebanyak 1.440 eksperimen dilakukan, dibagi merata (720) untuk setiap pengujian pemilihan fitur.

E. Evaluation

Confusion matrix [14] menjadi dasar tahap evaluasi ini, dari mana nilai Accuracy, Precision, Recall, dan F1-Score dihitung melalui persamaan (4) hingga (7). Keempat metrik tersebut disusun dari komponen TP (True Positive), TN (True Negative), FP (False Positive), dan FN (False Negative).

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

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

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

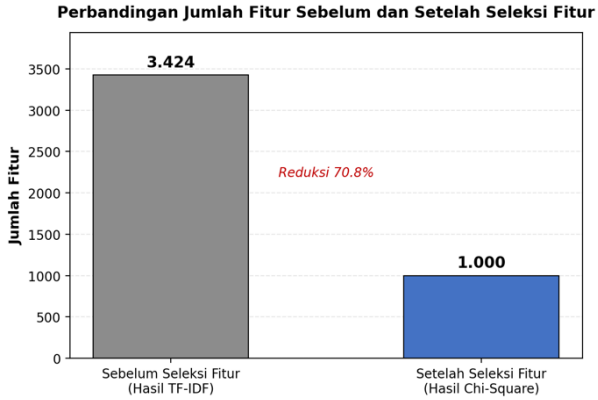
$$\text{F1-Score} = 2\times(\text{Precision}\times\text{Recall}) / (\text{Precision}+\text{Recall}) \quad (7)$$

III. Hasil dan Pembahasan

A. Hasil Preprocessing dan Transformation

Keenam langkah preprocessing, yaitu case folding, normalization, cleaning, tokenizing, stopword removal, dan lemmatization, diterapkan secara konsisten pada seluruh 944 dokumen ulasan. Hasilnya adalah korpus teks yang bersih dan seragam, siap dipakai sebagai dasar ekstraksi fitur pada tahap selanjutnya.

Ekstraksi fitur TF-IDF menghasilkan matriks berukuran 944×3.424. Dimensi ini kemudian dipangkas melalui seleksi Chi-Square menjadi 1.000 fitur dengan nilai χ^2 tertinggi. Perubahan jumlah fitur sebelum dan sesudah proses seleksi ini divisualisasikan pada Gambar. 5.



Gambar. 5 Perbandingan Jumlah Fitur Sebelum dan Setelah Seleksi Fitur

Seperti tampak pada Gambar. 5, jumlah fitur menyusut dari 3.424 menjadi 1.000, atau setara reduksi 70,8%. Fitur-fitur yang dipertahankan adalah yang keterkaitan statistiknya paling kuat dengan kelas sentimen, sementara fitur yang kurang informatif disingkirkan. Lima fitur dengan skor χ^2 tertinggi dapat dilihat pada Tabel I.

TABEL I
CONTOH HASIL *FEATURE SELECTION* CHI-SQUARE

Kata	Skor χ^2	p-value
amaze	6,85	0,0088
love	6,12	0,0134
graphic	5,97	0,0145
bug	5,44	0,0197
crash	5,21	0,0224

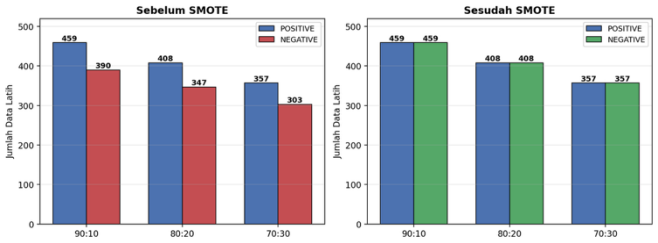
Tabel II merangkum hasil penerapan SMOTE pada ketiga skenario rasio split: kelas NEGATIVE di tiap rasio mendapat tambahan data sintetis hingga jumlahnya menyamai kelas POSITIVE. Perubahan jumlah data latih sebelum dan sesudah

SMOTE untuk ketiga rasio ini divisualisasikan pada Gambar. 6.

TABEL II
DISTRIBUSI KELAS SEBELUM DAN SETELAH SMOTE PADA SELURUH RASIO SPLIT DATA

Rasio	Sb+	Sb-	ΣSblm	Sintetis	Ss+	Ss-	ΣSsdh
90:10	459	390	849	+69	459	459	918
80:20	408	347	755	+61	408	408	816
70:30	357	303	660	+54	357	357	714

Terlihat dari Tabel II bahwa jumlah data sintetis yang dibangkitkan SMOTE untuk kelas NEGATIVE naik seiring bertambahnya proporsi data latih: 54 data pada rasio 70:30, 61 data pada rasio 80:20, hingga 69 data pada rasio 90:10. Ini wajar terjadi karena selisih jumlah sampel antara kelas POSITIVE dan NEGATIVE ikut melebar seiring data latih yang membesar.



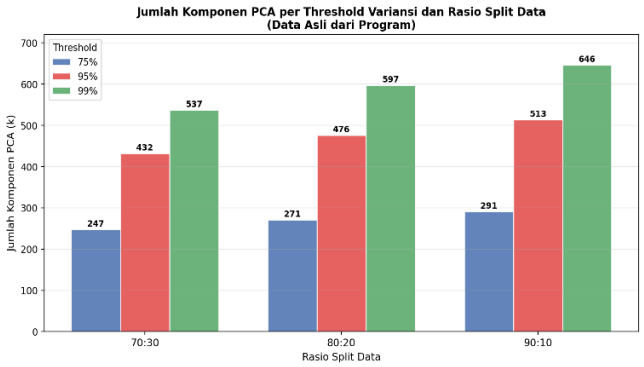
Gambar. 6 Perbandingan Jumlah Data Latih Sebelum dan Sesudah SMOTE per Rasio Split

Agar gambaran yang diperoleh tidak terpaku pada satu rasio saja, Tabel III merangkum hasil reduksi dimensi PCA di seluruh kombinasi tiga rasio split dan tiga ambang variansi yang diuji. Hasilnya divisualisasikan pada Gambar. 7 (perbandingan jumlah komponen antar ambang dan rasio) dan Gambar. 8 (kurva variansi kumulatif).

TABEL III
HASIL REDUKSI DIMENSI PCA PADA SELURUH RASIO SPLIT DATA DAN THRESHOLD VARIANSI

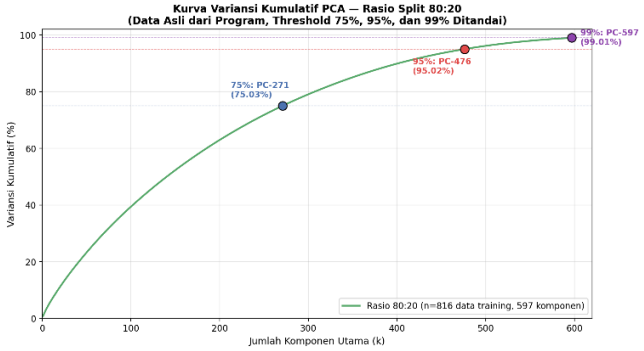
Rasio	Thr.	Train(SM)	Test	Komp.	Red(%)	VarK(%)
70:30	75%	714	284	247	75,30	75,05
70:30	95%	714	284	432	56,80	95,04
70:30	99%	714	284	537	46,30	99,00
80:20	75%	816	189	271	72,90	75,03
80:20	95%	816	189	476	52,40	95,02
80:20	99%	816	189	597	40,30	99,01
90:10	75%	918	95	291	70,90	75,11
90:10	95%	918	95	513	48,70	95,02
90:10	99%	918	95	646	35,40	99,00

Dari Tabel III terlihat kecenderungan jumlah komponen PCA pada ambang yang sama ikut bertambah seiring meningkatnya proporsi data latih. Hal ini wajar terjadi karena semakin banyak sampel yang tersedia untuk menangkap variansi data.



Gambar 7 Perbandingan Jumlah Komponen PCA per Threshold dan Rasio Split Data

Gambar 7 memperjelas pola tersebut: pada ambang yang sama, jumlah komponen PCA konsisten bertambah seiring naiknya rasio data latih di ketiga ambang (75%, 95%, 99%), sejalan dengan Tabel III. Pada rasio yang sama pun, semakin tinggi ambang variansi yang ditetapkan, semakin banyak komponen yang dibutuhkan untuk memenuhinya.



Gambar. 8 Kurva Variansi Kumulatif Rasio Split 80:20 (Threshold 75%, 95%, dan 99%)

Kurva variansi kumulatif pada Gambar 8 tampak melandai seiring bertambahnya jumlah komponen. Ini menunjukkan bahwa komponen-komponen di tahap awal menyumbang variansi jauh lebih besar dibandingkan komponen-komponen berikutnya, sehingga sebagian besar informasi data sebenarnya bisa diwakili dengan dimensi yang jauh lebih ringkas.

B. Hasil Machine Learning

Seluruh kombinasi antara empat kernel SVM (Linear, RBF, Polynomial, Sigmoid) dan tiga skenario rasio split data (90:10, 80:20, 70:30) diuji secara menyeluruh, menghasilkan dua belas kombinasi performa terbaik yang mewakili keseluruhan 1.440 kombinasi eksperimen. Tabel IV merangkum performa terbaik dari tiap kombinasi kernel dan rasio split tersebut.

TABEL IV
PERFORMA TERBAIK TIAP KERNEL PADA TIAP SKENARIO RASIO SPLIT DATA

Rasio	Kernel	PCA	C	γ	Acc(%)	F1(%)
90:10	Linear	75%	0,1	-	85,26	85,71
90:10	RBF	95%	1	0,001	93,68	94,12
90:10	Polynomial	99%	10	0,001	83,16	85,96
90:10	Sigmoid	95%	1	0,001	89,47	90,38
80:20	Linear	75%	0,1	-	85,71	86,70
80:20	RBF	75%	1	0,001	92,06	92,75

80:20	Polynomial	Tanpa PCA	10	1	80,42	83,11
80:20	Sigmoid	75%	1	0,001	92,06	92,75
70:30	Linear	75%	0,1	-	85,21	86,00
70:30	RBF	95%	1	0,001	88,73	89,81
70:30	Polynomial	Tanpa PCA	10	1	83,10	84,00
70:30	Sigmoid	75%	1	0,001	88,38	89,25

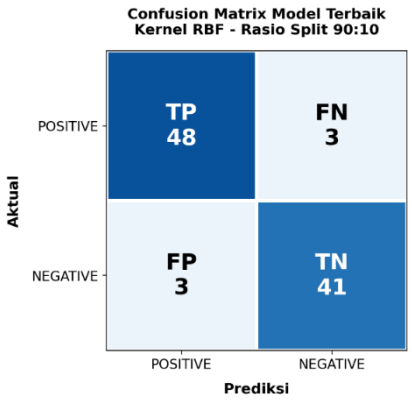
Tabel IV menunjukkan kernel RBF konsisten unggul dibandingkan ketiga kernel lainnya pada seluruh skenario rasio split, dengan F1-Score tertinggi pada rasio 90:10 (94,12%), lalu 80:20 (92,75%), dan 70:30 (89,81%). Kernel Sigmoid selalu menempati posisi kedua di ketiga rasio, sementara Polynomial konsisten menjadi yang terlemah. Karena rasio 90:10 unggul pada tiga dari empat kernel (RBF, Sigmoid, Linear), rasio ini dipilih sebagai skenario utama untuk pembahasan lebih lanjut, dengan performa terbaik tiap kernelnya dirangkum pada Tabel V.

TABEL V
PERFORMA TERBAIK TIAP KERNEL PADA RASIO 90:10

Kernel	Arsitektur	C	γ	Acc(%)	F1(%)
Linear	PCA 75% Sblm SMOTE	0,1	-	85,26	85,71
RBF	PCA 95% Stlh SMOTE	1	0,001	93,68	94,12
Polynomial	PCA 99% Stlh SMOTE	10	0,001	83,16	85,96
Sigmoid	PCA 95% Sblm SMOTE	1	0,001	89,47	90,38

C. Hasil Evaluasi

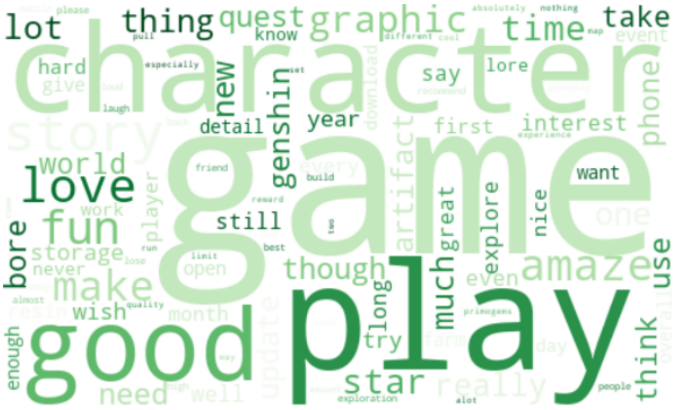
Konfigurasi dengan performa terbaik secara keseluruhan adalah kernel RBF yang dipadukan dengan seleksi fitur Chi-Square, arsitektur PCA yang diterapkan setelah SMOTE (ambang 95%, 513 komponen), rasio split 90:10, C=1, dan gamma=0,001. Confusion matrix dari konfigurasi ini ditampilkan pada Gambar. 9.



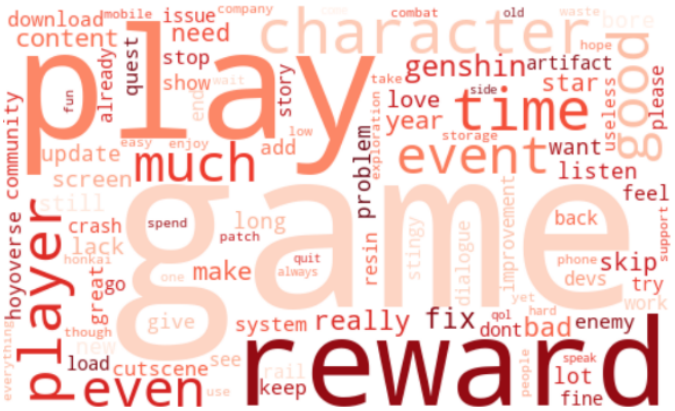
Gambar. 9 Confusion Matrix Model Terbaik

Gambar. 9 menunjukkan model mencapai Accuracy 93,68%, Precision 94,12%, Recall 94,12%, dan F1-Score 94,12%. Tingkat kesalahan klasifikasinya pun tergolong kecil: hanya 3 dari total 95 dokumen uji POSITIVE yang salah diklasifikasikan sebagai NEGATIVE, dan sebaliknya 3 dokumen NEGATIVE salah diklasifikasikan sebagai POSITIVE.

Untuk melengkapi gambaran secara kualitatif, kata-kata paling dominan pada masing-masing kelas hasil prediksi model terbaik divisualisasikan lewat word cloud, seperti terlihat pada Gambar. 10 untuk kelas POSITIVE dan Gambar. 11 untuk kelas NEGATIVE.



Gambar. 10 Word Cloud Kelas POSITIVE



Gambar. 11 Word Cloud Kelas NEGATIVE

Kata seperti "amaze", "love", dan "graphic" muncul menonjol pada kelas POSITIVE. Hal ini wajar karena mencerminkan apresiasi pemain terhadap kualitas visual dan pengalaman bermain secara umum. Sebaliknya, kata-kata pada kelas NEGATIVE lebih banyak terkait keluhan teknis, konsisten dengan fitur-fitur ber-skor χ^2 tertinggi yang sudah ditampilkan pada Tabel I.

D. Pembahasan

Sejauh mana masing-masing komponen metodologi memengaruhi rata-rata F1-Score dari keseluruhan kombinasi dapat dilihat pada Tabel VI.

TABEL VI
PENGARUH KOMPONEN METODOLOGI TERHADAP RATA-RATA F1-SCORE

Komponen	Dengan (%)	Tanpa (%)	Selisih
Chi-Square	76,60	72,27	+4,33
PCA	77,90	72,73	+5,17
SMOTE	76,50	76,71	−0,21

Tabel VI memperlihatkan bahwa seleksi fitur Chi-Square dan reduksi dimensi PCA sama-sama menyumbang peningkatan performa klasifikasi yang signifikan, sedangkan

kontribusi SMOTE relatif lebih kecil. Hal ini konsisten dengan tingkat ketidakseimbangan kelas pada dataset penelitian yang tergolong ringan (Imbalance Ratio 0,8510). Temuan ini sejalan dengan penelitian [15], yang juga mendapati peningkatan performa signifikan usai penerapan SMOTE pada kasus dengan ketidakseimbangan kelas yang lebih besar; artinya, manfaat SMOTE tampaknya memang berbanding lurus dengan seberapa timpang kelas pada dataset yang bersangkutan.

Penelitian [2] pada domain serupa (ulasan Genshin Impact) sebetulnya sudah mencapai performa tinggi tanpa melibatkan PCA maupun Chi-Square. Namun penelitian ini menunjukkan bahwa memadukan keduanya justru membawa dua manfaat sekaligus: performa klasifikasi meningkat, dan dimensi komputasi menyusut drastis, dari 3.424 fitur TF-IDF awal menjadi hanya 513 komponen pada konfigurasi terbaik (reduksi 85,0%). Hal ini pada akhirnya membuat waktu pelatihan model jadi lebih efisien tanpa mengorbankan akurasi, sejalan dengan apa yang dibuktikan penelitian [6] dan [7] soal manfaat reduksi dimensi PCA pada data berdimensi tinggi.

Pada tiga rasio pembagian yang ditunjukkan dalam Tabel IV, kernel RBF menempati peringkat pertama, dan kedua adalah Sigmoid, sementara di bagian bawah adalah Polynomial. Hasil ini sejalan dengan literatur, karena kernel RBF adalah yang paling fleksibel [13], dan merupakan karakteristik yang baik saat menangani data teks berdimensi tinggi setelah reduksi PCA. Setelah PCA, kernel RBF juga tampil paling baik, dan hasil yang konsisten diperoleh menggunakan varians 95% sebagai ambang batas, dibandingkan dengan 75% dan 99%, di mana 75% menyebabkan kehilangan informasi yang signifikan dan 99% dianggap tidak memadai, menggambarkan bahwa 95% mencapai keseimbangan terbaik. PCA setelah SMOTE juga secara konsisten mengungguli PCA sebelum SMOTE, karena arah varians setelah PCA mewakili kedua kelas karena distribusi kelas yang seimbang.

Dari sisi rasio split data, kombinasi terbaik selalu diraih pada rasio 90:10, disusul 80:20, dan terakhir 70:30. Hal ini menandakan bahwa proporsi data latih yang lebih besar memberi model SVM lebih banyak informasi untuk mempelajari batas keputusan optimal antara kelas POSITIVE dan NEGATIVE, meski konsekuensinya jumlah data uji untuk validasi jadi lebih sedikit. Di sisi lain, skenario membandingkan Dengan Chi-Square dan Tanpa Chi-Square menunjukkan bahwa manfaat F1-Score rata-rata dan pemilihan fitur tampaknya konsisten dan signifikan di semua rasio pembagian dan dalam semua kerangka PCA-SMOTE, menunjukkan bahwa manfaatnya konsisten dan tidak terbatas pada satu pengaturan eksperimen.

Dalam hal efisiensi komputasi, menggunakan 513 komponen utama dalam skenario terbaik dibandingkan 1.000 fitur Chi-Square mengurangi kompleksitas pelatihan SVM, karena kompleksitas komputasi pelatihan SVM hampir langsung sebanding dengan jumlah fitur input. Oleh karena itu, menggabungkan PCA dengan SVM tidak hanya memberikan keunggulan akurasi, tetapi juga dapat memberikan keunggulan penghematan komputasi. Ini adalah alasan (dan mengapa) memantau secara berkala sentimen ulasan, terutama ketika sebuah aplikasi target memiliki banyak ulasan. Potensinya, ini dapat membantu pengembang Genshin Impact untuk melihat ulasan pemain secara lebih sistematis. Metodologi yang diusulkan dalam makalah ini juga dapat diintegrasikan ke dalam alat pemantauan ulasan otomatis untuk meningkatkan pengembangan aplikasi berdasarkan umpan balik dari komunitas pengguna.

Meskipun temuan penelitian ini bisa bernilai, ada beberapa keterbatasan penting. Dataset dalam penelitian ini hanya 944 ulasan berbahasa Inggris selama 3 bulan, yang membatasi kemampuan penulis untuk menggeneralisasi temuan ke ulasan yang ditulis dalam bahasa lain, atau ke ulasan yang ditulis di luar periode waktu ini. Penulis juga membatasi penelitian ini pada klasifikasi biner ulasan (POSITIF dan NEGATIF), dimana ulasan dengan sentimen campuran atau netral dikeluarkan.

IV. KESIMPULAN

Melalui kerangka kerja KDD, penelitian ini berhasil mengklasifikasikan sentimen ulasan game Genshin Impact menggunakan SVM setelah dimensi datanya direduksi dengan PCA. Hasil terbaik, akurasi sebesar 93,68% dengan skor F1 sebesar 94,12%, dicapai pada rasio pembagian 90:10, menggunakan kernel RBF dengan SMOTE (varians 95%), dengan C=1 dan gamma=0.001. Pemilihan Fitur Chi-Square, SMOTE, dan PCA memiliki ukuran efek sebesar +4.33, +5.17, dan efek efek yang lebih rendah, masing-masing, karena ketidakseimbangan kelas yang sedang.

V. SARAN

Ke depannya, penelitian ini bisa dikembangkan dengan menjajaki metode ekstraksi fitur kontekstual seperti BERT atau IndoBERT, memperbesar ukuran dataset sekaligus rentang waktu pengambilan datanya, menambahkan kelas netral untuk skema klasifikasi multi-kelas, serta membandingkan performa dengan algoritma lain seperti Random Forest atau model deep learning pada karakteristik data serupa demi memperkuat generalisasi model.

UCAPAN TERIMA KASIH

Ucapan terima kasih penulis sampaikan kepada dosen pembimbing serta semua pihak yang telah membantu hingga penelitian ini dapat terselesaikan.

REFERENSI

[1] M. Wankhade, A. C. S. Rao, and C. Kulkarni, "A survey on sentiment analysis methods, applications, and challenges," Artif. Intell. Rev., vol. 55, pp. 5731–5780, 2022, doi: 10.1007/s10462-022-10144-1.

[2] M. Aldiansyah Pratama, F. N. Hasan, and M. A. Pratama, "Comparison of the Naïve Bayes Method and Support Vector Machine in Sentiment Analysis of Genshin Impact Game Reviews," MECOMARE, vol. 13, no. 2, pp. 76–88, 2024, doi: 10.35335/computational.v13i2.198.

[3] P. Arsi, P. Subarkah, and B. A. Kusuma, "Analisis sentimen game Genshin Impact pada Play Store menggunakan Naïve Bayes Classifier," J. Ilm. Tek. Mesin Elektro dan Komput., vol. 3, no. 1, pp. 161–170, 2023.

[4] S. I. A. Jaya and J. Zeniarja, "Sentiment analysis of Genshin Impact on X: Mental health implications using TF-IDF and Support Vector Machine," SinkrOn, vol. 8, no. 3, pp. 1589–1599, 2024, doi: 10.33395/sinkron.v8i3.13716.

[5] M. Safrudin, Martanto, and U. Hayati, "Perbandingan kinerja Naïve Bayes dan Support Vector Machine untuk klasifikasi sentimen ulasan game Genshin Impact," J. Mhs. Tek. Inform., vol. 8, no. 3, pp. 3182–3188, 2024, doi: 10.36040/jati.v8i3.8415.

[6] A. M. Fajria, A. Faqih, and G. Dwilestari, "The impact of Principal Component Analysis dimensionality reduction on sentiment classification performance using Support Vector Machine," J. Artif. Intell. Eng. Appl., vol. 4, no. 2, pp. 1–10, 2025, doi: 10.59934/jaiea.v4i2.744.

[7] F. Diba, M. S. Lydia, and P. Sihombing, "Analisis Random Forest menggunakan Principal Component Analysis pada data berdimensi tinggi," Indones. J. Comput. Sci., vol. 12, no. 4, 2023, doi: 10.33022/ijcs.v12i4.3329.
[8] K. N. Haridas, "Knowledge Discovery in Databases (KDD) Process in Data Mining," Int. J. Sci. Eng. Technol., vol. 13, no. 6, 2026, doi: 10.5281/zenodo.19885681.
[9] N. A. Semary, W. Ahmed, K. Amin, P. Pławiak, and M. Hammad, "Enhancing machine learning-based sentiment analysis through feature extraction techniques," PLOS ONE, vol. 19, no. 2, e0294968, 2024, doi: 10.1371/journal.pone.0294968.
[10] E. Hokijuliandy, H. Napitupulu, and Firdaniza, "Application of SVM and Chi-square feature selection for sentiment analysis of Indonesia's National Health Insurance Mobile Application," Mathematics, vol. 11, no. 17, 3765, 2023, doi: 10.3390/math11173765.
[11] I. M. Alkhawaldeh, I. Albalkhi, and A. J. Naswhan, "Challenges and limitations of synthetic minority oversampling techniques in machine learning," World J. Methodol., vol. 13, no. 5, pp. 373–378, 2023, doi: 10.5662/wjm.v13.i5.373.
[12] A. A. Wani, "Comprehensive review of dimensionality reduction algorithms: challenges, limitations, and innovative solutions," PeerJ Comput. Sci., vol. 11, e3025, 2025, doi: 10.7717/peerj-cs.3025.
[13] K.-L. Du, B. Jiang, J. Lu, J. Hua, and M. N. S. Swamy, "Exploring kernel machines and support vector machines: Principles, techniques, and future directions," Mathematics, vol. 12, no. 24, 3935, 2024, doi: 10.3390/math12243935.
[14] M. K. Suryadewiansyah and T. E. E. Tju, "Naïve Bayes dan Confusion Matrix untuk efisiensi analisa Intrusion Detection System Alert," J. Nas. Teknol. dan Sist. Inf., vol. 8, no. 2, pp. 81–88, 2022, doi: 10.25077/TEKNOSI.v8i2.2022.81-88.
[15] R. A. Nurdian, M. Ridwan, and A. Yusuf, "Komparasi metode SMOTE dan ADASYN dalam meningkatkan performa klasifikasi herregistrasi mahasiswa baru," J. Tek. Inform. dan Sist. Inf., vol. 8, no. 1, 2022, doi: 10.28932/jutisi.v8i1.4004.