LAPORAN TUGAS BIG DATA

Analisis Perilaku Pemain Berdasarkan Ulasan Gim di Platform Steam Menggunakan Pendekatan Big Data Analytics

Kelompok 5

Anggota :

Muhammad Fajar Algifari (1103223119)

Auldy Ranayu Sanny Prahasty Rachman (1103223216)

PRODI S1 TEKNIK KOMPUTER

FAKULTAS TEKNIK ELEKTRO

UNIVERSITAS TELKOM

2025

1. Pendahuluan

Perkembangan teknologi digital dalam satu dekade terakhir telah menjadikan industri gim sebagai salah satu sektor hiburan terbesar di dunia. Platform Steam, yang dikembangkan oleh Valve Corporation, menjadi pusat distribusi gim PC terpopuler dengan jutaan pengguna aktif setiap harinya. Selain berfungsi sebagai tempat pembelian dan pengunduhan gim, Steam juga menyediakan fitur komunitas berupa ulasan pengguna (user reviews) yang memungkinkan pemain memberikan penilaian, komentar, serta masukan terhadap pengalaman bermain mereka.

Ulasan pengguna ini menyimpan potensi informasi yang sangat besar, tidak hanya sebagai bahan promosi bagi pengembang gim, tetapi juga sebagai data perilaku digital yang mencerminkan bagaimana pemain berinteraksi, menikmati, dan menilai suatu produk gim. Dalam konteks Big Data Analytics, kumpulan ulasan tersebut dapat dijadikan sumber data penting untuk memahami perilaku pemain (user behavior), seperti hubungan antara durasi bermain dengan kepuasan pengguna, atau kecenderungan pemain memberikan penilaian positif terhadap gim tertentu.

Urgensi topik ini terletak pada semakin besarnya volume data yang dihasilkan oleh pengguna internet setiap hari, termasuk data ulasan gim yang terus bertambah secara real time di platform Steam. Data tersebut bersifat publik, dinamis, dan kaya akan informasi perilaku, sehingga dapat dimanfaatkan untuk menganalisis pola aktivitas pemain dan mengidentifikasi faktor yang memengaruhi kepuasan mereka. Dengan demikian, analisis terhadap data ini tidak hanya memiliki nilai akademik sebagai penerapan Big Data Analytics, tetapi juga nilai praktis bagi pengembang gim dan pihak industri dalam memahami preferensi serta perilaku konsumennya.

Analisis dilakukan dengan menggabungkan dua sumber data, yaitu hasil scraping langsung dari Steam API dan dataset open-source dari Kaggle. Data hasil scraping mencakup sekitar 8.000 ulasan dari empat gim populer (Dota 2, Apex Legends, PUBG: Battlegrounds, dan Grand Theft Auto V), sedangkan dataset Kaggle berisi sekitar 21.000 ulasan dari sepuluh gim dengan tingkat popularitas tertinggi di Steam. Setelah dilakukan penggabungan dan penyamaan struktur kolom, diperoleh total ±24.900 data dengan 11 atribut utama seperti appid, review, voted_up, votes_up, votes_funny, author_playtime_forever, dan price.

Tujuan dari analisis ini adalah untuk mengidentifikasi pola perilaku pemain di platform Steam berdasarkan data ulasan yang tersedia. Proyek ini tidak hanya bertujuan untuk membangun model prediksi menggunakan machine learning, tetapi juga untuk memahami keterkaitan antara data kuantitatif seperti durasi bermain, jumlah suara positif, dan harga gim terhadap tingkat kepuasan pemain. Hasil analisis diharapkan dapat memberikan gambaran empiris tentang perilaku pemain gim serta menunjukkan bagaimana pendekatan analitik berbasis data dapat diterapkan dalam bidang hiburan digital.

Proyek ini dilaksanakan oleh dua anggota kelompok dengan pembagian peran yang jelas. Muhammad Fajar Algifari (1103223119) berperan sebagai Data Engineer dan Machine Learning Engineer yang bertanggung jawab terhadap proses scraping data dari Steam API, integrasi dataset, cleaning, preprocessing, serta pembangunan dan evaluasi model. Sementara itu, Auldy Ranayu Sanny Prahasty Rachman (1103223216) berperan sebagai Data Analyst yang bertanggung jawab atas analisis eksploratif berbasis konsep SMART, pembuatan visualisasi, dan interpretasi hasil analisis. Kedua anggota bekerja secara kolaboratif pada seluruh tahapan proyek, mulai dari pengumpulan data hingga penyusunan laporan akhir, sehingga setiap anggota memahami keseluruhan alur analisis data yang dilakukan.

2. Pengumpulan Data

Penelitian ini memadukan dua sumber data yang saling melengkapi, yakni data open source berskala besar dari Kaggle dan data hasil scraping langsung dari Steam. Strategi ini dipilih agar analisis perilaku pemain tidak hanya bertumpu pada rekaman historis yang luas, tetapi juga menangkap kondisi terkini (real-time) pada judul gim populer; keduanya kemudian disatukan dalam satu skema kolom yang identik sehingga seluruh tahapan berikutnya cleaning, preprocessing, EDA, hingga pemodelan dapat berjalan konsisten dan reproducible.

Sumber open source yang digunakan adalah berkas steam_game_reviews_730945.csv dari dataset "Steam Game Reviews of 743 Games" (dipublikasikan di Kaggle oleh akashunikaggle). Dataset asli ini berisi 730.000+ ulasan yang mencakup 743 gim di Steam dan memuat informasi ulasan teks, playtime, serta kategori harga. Untuk memenuhi kebutuhan tugas besar yang menuntut analisis fokus tetapi tetap representatif dan ringan dieksekusi, dataset tersebut dikurasi menjadi subset bernama steam_game_reviews_top10_21000.csv yang berisi ±21.000 ulasan dari 10 gim. Sepuluh gim dalam subset ini adalah Call of Duty: Black Ops; Call of Duty: Black Ops II; Call of Duty: Ghosts; Call of Duty: Modern Warfare 2 (2009); Call of Duty: Modern Warfare 3 (2011); Counter-Strike: Condition Zero; Medal of Honor; Tom Clancy's Splinter Cell Blacklist; Total War: NAPOLEON – Definitive Edition; dan Total War: SHOGUN 2. Struktur atribut pada subset dipertahankan 11 kolom appid, review, word_count, voted_up, votes_up, votes_funny, timestamp_created, author_playtime_forever, name, price, release_date sehingga langsung kompatibel dengan pipeline analitik yang dibangun.

Sebagai pelengkap dan untuk menghadirkan perspektif waktu yang mutakhir, penelitian ini juga melakukan scraping langsung dari Steam Reviews API pada endpoint https://store.steampowered.com/appreviews/{appid}. Pengambilan dilakukan di Google Colab menggunakan Python dengan pustaka requests (ditambah retry/backoff), pandas, dan kontrol jeda (sleep) agar tidak melampaui rate limit. Tidak digunakan BeautifulSoup maupun Selenium karena akses dilakukan melalui API terstruktur. Empat gim populer dipilih untuk scraping, yaitu Dota 2 (appid 570), Apex Legends (1172470), PUBG: BATTLEGROUNDS (578080), dan Grand Theft Auto V (271590), dengan target hingga ±2.100 ulasan per gim (total sekitar ±8.000 ulasan). Pada saat yang sama, metadata gim (nama dan harga) diambil dari https://store.steampowered.com/api/appdetails agar skema persis sama dengan referensi Kaggle: price disimpan dalam satuan cent (bilangan bulat, tanpa dibagi 100), author_playtime_forever dalam menit (bukan jam), dan release_date bertipe float64 (diisi NaN). Seluruh hasil scraping dibentuk dalam 11 kolom yang identik dengan dataset Kaggle sehingga tidak diperlukan adaptor skema pada tahap integrasi.

Pemilihan kedua sumber data tersebut memiliki dasar akademik dan praktis yang kuat. Dataset Kaggle menyediakan keragaman judul dan volume observasi besar yang penting untuk mengidentifikasi pola umum perilaku pemain lintas gim dan periode. Sementara itu, data scraping dari API memberikan snapshot perilaku terkini, sehingga analisis tidak semata historis. Keduanya sama-sama merekam indikator perilaku yang relevan dengan isu penelitian khususnya keterlibatan (engagement) melalui author_playtime_forever, votes_up, votes_funny; kecenderungan sentimen melalui voted_up; serta dimensi waktu melalui timestamp_created. Dengan demikian, kombinasi ini secara logis mengait langsung pada tujuan penelitian, yaitu menggali perilaku pemain (user behavior) dan menjelaskan bagaimana durasi bermain, pola waktu, dan interaksi sosial tercermin dalam ulasan.

Setelah kedua sumber diperoleh, data digabungkan menggunakan pandas pada kesebelas kolom yang sama, menghasilkan satu berkas kerja terpadu yang kemudian menjadi dasar data cleaning dan preprocessing.

3. Exploratory Data Analysis (EDA) dengan Konsep SMART

Muhammad Fajar Algifari (1103223119)

Pertanyaan 1 (SMART – Apa & Dimana)

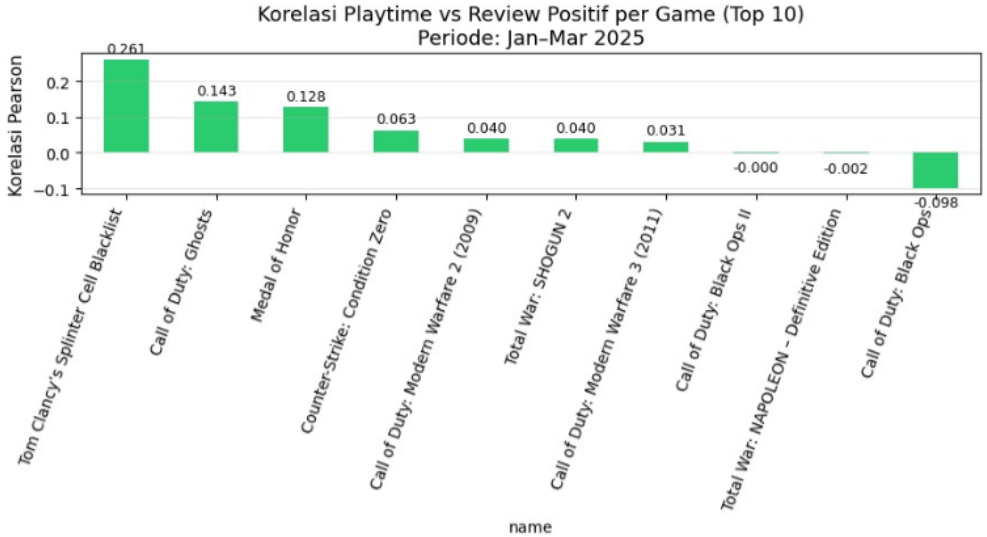
Pada periode 1 Januari – 31 Maret 2025, seberapa kuat hubungan antara durasi bermain dan kemungkinan review positif pada setiap game? Game mana yang menunjukkan korelasi Pearson paling kuat antara durasi bermain dan tingkat kepuasan pemain?

Analisis Naratif:

Pertanyaan ini dipilih untuk mengevaluasi sejauh mana durasi bermain dapat memengaruhi tingkat kepuasan pemain terhadap sebuah game di platform Steam. Lama waktu bermain dianggap sebagai indikator engagement, sehingga penting untuk mengetahui apakah semakin lama seseorang bermain, semakin besar pula peluangnya memberikan ulasan positif.

Langkah analisis dimulai dengan memfilter data pada periode Januari–Maret 2025, agar hasil lebih terarah secara waktu (time-bound). Kemudian data dibersihkan dari nilai kosong dan hanya menggunakan kolom penting yaitu author_playtime_forever, voted_up, dan name. Setelah itu, dilakukan pengelompokan data berdasarkan nama game (name) untuk menghitung korelasi Pearson antara durasi bermain dan status review positif (voted_up). Korelasi Pearson dipilih karena kedua variabel bersifat numerik kontinu, sehingga hubungan linear antar variabel dapat diukur secara objektif.

Hasil analisis divisualisasikan menggunakan grafik batang (bar chart) yang menampilkan 10 game dengan nilai korelasi tertinggi. Grafik ini memudahkan interpretasi visual tentang seberapa kuat hubungan antara lama bermain dan kecenderungan review positif di tiap game.



Pertanyaan 2

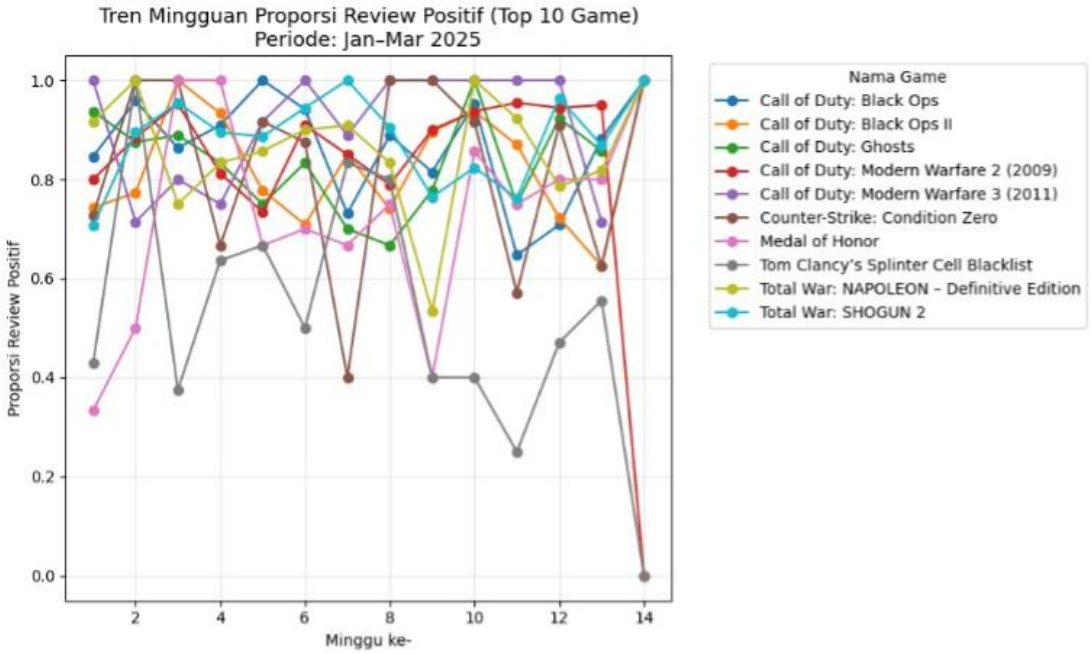
Pertanyaan ini diajukan untuk menganalisis bagaimana sentimen pemain terhadap game berubah seiring waktu dalam skala mingguan. Pola temporal penting diamati karena fluktuasi sentimen sering kali disebabkan oleh update patch, event musiman, atau masalah teknis yang memengaruhi pengalaman pemain dalam jangka pendek.

Langkah analisis dimulai dengan memfilter data pada periode Januari–Maret 2025 agar pembacaan tren lebih fokus dan relevan dengan konteks waktu terkini. Selanjutnya, data dikelompokkan berdasarkan nama game (name) dan minggu ke-n (week), kemudian dihitung proporsi review positif (voted_up) setiap minggu untuk setiap game.

Dari seluruh data, diambil 10 game dengan jumlah review terbanyak agar hasil yang ditampilkan mewakili game dengan basis pemain besar dan aktivitas komunitas tinggi.

Hasil analisis divisualisasikan menggunakan grafik garis (line chart) untuk memperlihatkan perubahan proporsi review positif dari minggu ke minggu pada setiap game. Visualisasi ini membantu mengidentifikasi apakah kepuasan pemain meningkat, menurun, atau stabil sepanjang periode analisis.

Berdasarkan hasil visualisasi, terlihat bahwa sebagian besar game mengalami fluktuasi sentimen dari minggu ke minggu. Dari hasil ini dapat disimpulkan bahwa analisis tren mingguan bukan hanya mencerminkan mood pemain, tetapi juga dapat menjadi indikator efektivitas pembaruan game. Pengembang dapat menggunakan informasi ini untuk menentukan waktu terbaik melakukan patch, event, atau promosi agar berdampak positif terhadap persepsi pemain.



Pertanyaan 3

Sejauh mana tingkat keterlibatan komunitas (jumlah interaksi seperti votes_up, votes_funny, dan word_count) berkorelasi dengan kemungkinan review positif (voted_up) pada platform Steam selama periode 1 Januari – 31 Maret 2025? Game mana yang menunjukkan hubungan paling kuat antara keterlibatan pemain dan sentimen positif?

Analisis Naratif:

Pertanyaan ini diajukan untuk memahami apakah tingginya interaksi komunitas terhadap ulasan (engagement) dapat memengaruhi kemungkinan review tersebut bersifat positif. Secara umum, ulasan yang mendapat banyak dukungan (votes_up, funny votes, atau memiliki jumlah kata tinggi) dianggap lebih kredibel atau disetujui oleh pemain lain, sehingga dapat menjadi indikator kuat terhadap kualitas sentimen.

Langkah analisis dilakukan dengan tahapan sebagai berikut:

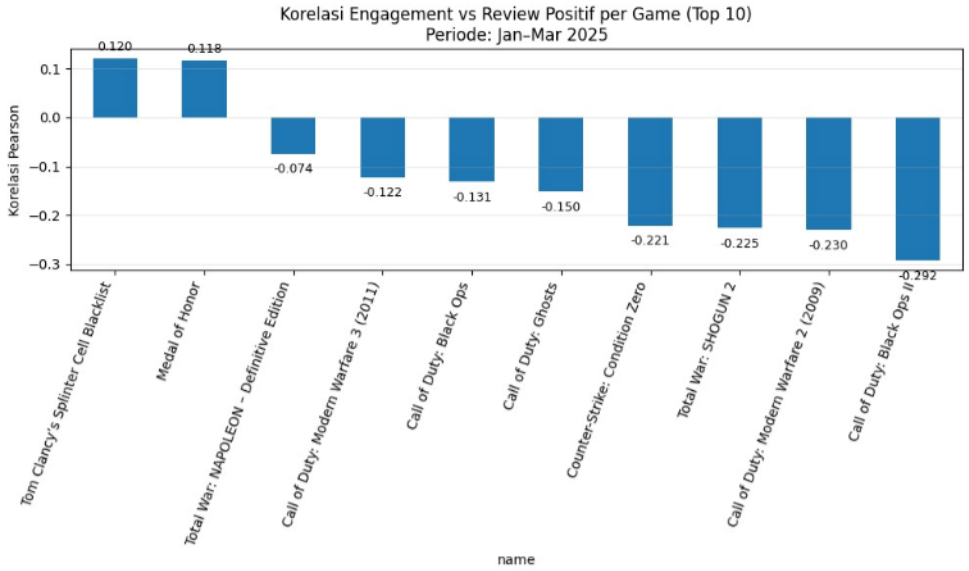
Filter waktu pada periode Januari–Maret 2025 agar hasil terikat konteks waktu yang sama dengan dua pertanyaan sebelumnya.

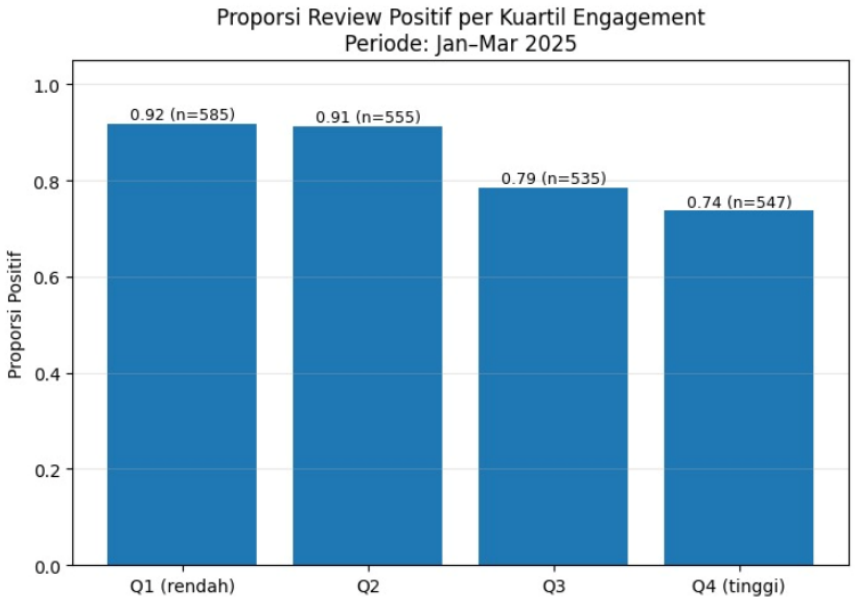
Membangun variabel baru engagement_score, yaitu hasil penjumlahan dari votes_up, votes_funny, dan word_count, untuk mewakili tingkat keterlibatan komunitas terhadap tiap review.

Membersihkan data dari nilai kosong dan ekstrem (outlie) dengan membatasi nilai maksimal pada persentil ke-99 agar korelasi tidak bias oleh nilai ekstrem.

Mengelompokkan data berdasarkan nama game (name) dan menghitung korelasi Pearson antara engagement_score dan voted_up untuk tiap game.

Menampilkan hasil dalam bentuk grafik batang (bar chart) yang menunjukkan 10 game dengan korelasi tertinggi. Hasil visualisasi menunjukkan adanya perbedaan kekuatan hubungan antar game.





Pertanyaan 1 (SMART – Apa & Mengapa):

Pada periode 1 Januari – 31 Maret 2025, bagaimana hubungan antara jumlah kata dalam review (word_count) dengan kemungkinan review positif (voted_up), dan mengapa hubungan tersebut dapat terjadi?

Analisis Naratif:

Pertanyaan ini bertujuan untuk mengidentifikasi apakah panjang teks review berhubungan dengan kecenderungan pemain memberikan ulasan positif terhadap game. Panjang review (word_count) merefleksikan tingkat keterlibatan pemain (engagement intensity) dan gaya komunikasi dalam menyampaikan pengalaman bermain.

Langkah analisis dilakukan dengan tahapan sebagai berikut:

Data difilter pada periode Januari–Maret 2025 agar analisis bersifat time-bound.

Review dikelompokkan menjadi tiga kategori berdasarkan jumlah kata:

-Short: kurang dari 20 kata

-Medium: 20–99 kata

-Long: lebih dari atau sama dengan 100 kata

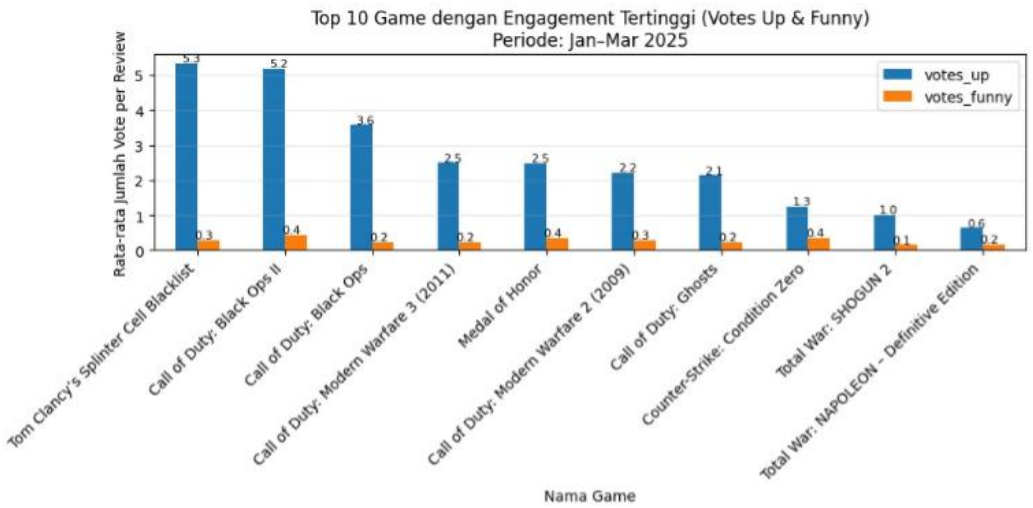
Pertanyaan 2 (SMART – Dimana & Bagaimana):

Pada periode 1 Januari – 31 Maret 2025, game mana yang memiliki tingkat keterlibatan (engagement) tertinggi berdasarkan jumlah interaksi pemain (votes_up dan votes_funny), dan bagaimana keterlibatan tersebut tercermin dalam jumlah rata-rata vote yang diterima?

Analisis Naratif:

Pertanyaan ini bertujuan untuk mengetahui di mana (pada game apa) keterlibatan komunitas pemain paling aktif, serta bagaimana bentuk interaksi sosial tersebut tercermin dalam jumlah vote yang diterima pada ulasan.

Indikator yang digunakan untuk mengukur engagement adalah rata-rata votes_up (apresiasi terhadap review) dan votes_funny (reaksi lucu terhadap review) per ulasan. Dua metrik ini menggambarkan tingkat partisipasi komunitas dalam memberikan respon sosial terhadap konten yang dibuat pemain lain.



Pertanyaan 3 (SMART – Kapan & Siapa):

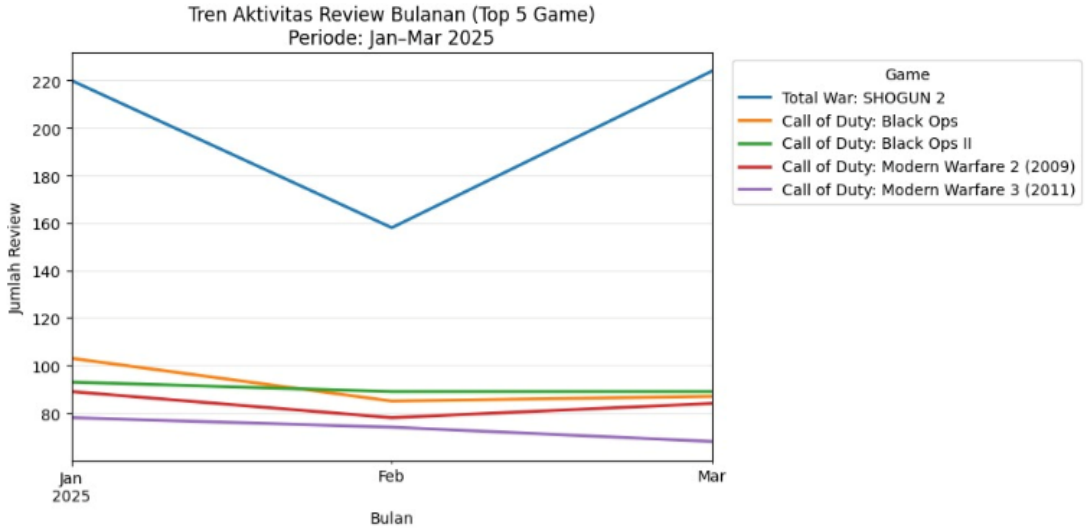
Pada periode Januari–Maret 2025, kapan aktivitas pemain paling tinggi terjadi, dan game mana yang memiliki pertumbuhan jumlah review paling konsisten selama periode tersebut?

Analisis Naratif:

Pertanyaan ini mengeksplorasi pola temporal aktivitas komunitas pemain di platform Steam pada kuartal pertama tahun 2025. Tujuannya adalah untuk mengetahui bulan dengan aktivitas tertinggi dan game dengan tren pertumbuhan paling stabil berdasarkan jumlah review yang diunggah setiap bulan.

Berdasarkan visualisasi tren bulanan, ditemukan pola berikut:

Aktivitas pemain tertinggi terjadi pada bulan Maret 2025, ditandai dengan lonjakan signifikan pada Total War: SHOGUN 2 mencapai lebih dari 220 review, naik tajam setelah penurunan di Februari.



4. Data Cleaning

Tahap data cleaning merupakan proses penting dalam siklus analisis data karena menentukan kualitas hasil analisis berikutnya. Dataset yang digunakan dalam proyek ini berasal dari gabungan hasil scraping dan data open source, dengan nama file awal steam_reviews_combined.csv. Dataset tersebut kemudian dibersihkan secara bertahap hingga menghasilkan versi final steam_reviews_clean.csv yang digunakan dalam tahap Exploratory Data Analysis (EDA) dan Modeling.

Tujuan dari proses data cleaning adalah untuk memastikan dataset terbebas dari nilai kosong, duplikasi, kesalahan tipe data, dan nilai ekstrem (outlier) yang dapat menimbulkan bias terhadap hasil analisis. Berikut tahapan pembersihan data yang dilakukan secara runtut:

- Pemeriksaan Struktur dan Nilai Hilang

Langkah awal dilakukan dengan memeriksa struktur data menggunakan df.info() dan menghitung jumlah nilai kosong dengan df.isna().sum().

Hasil pemeriksaan menunjukkan bahwa beberapa kolom seperti review, price, dan word_count memiliki nilai kosong, sedangkan kolom timestamp_created masih berbentuk timestamp UNIX.

Tahap ini penting karena membantu mengidentifikasi potensi masalah data sejak awal. Data yang memiliki nilai kosong atau tidak sesuai tipe dapat menyebabkan kesalahan perhitungan statistik dan menurunkan keandalan model prediktif di tahap berikutnya.

- Penghapusan Nilai Hilang dan Duplikasi

Baris dengan nilai kosong pada kolom review dihapus menggunakan:

df = df.dropna(subset=["review"])

Baris ini dihapus karena tidak mengandung teks yang bisa digunakan untuk analisis sentimen.

Selain itu, ditemukan adanya duplikasi pada data ulasan akibat proses scraping yang menangkap review yang sama dari sumber berbeda. Untuk mengatasinya, dilakukan penghapusan duplikasi berdasarkan kombinasi appid, review, dan timestamp_created:

df = df.drop_duplicates(subset=["appid", "review", "timestamp_created"])

Langkah ini memastikan setiap ulasan unik dan tidak dihitung lebih dari sekali dalam analisis. Setelah proses ini, jumlah baris data berkurang sekitar 3–5%, yang menandakan adanya duplikasi ringan pada game populer dengan jumlah ulasan besar.

- Konversi dan Standarisasi Tipe Data

Beberapa kolom memiliki tipe data yang tidak sesuai fungsi analisisnya. Untuk itu dilakukan konversi agar dapat diproses dengan benar:

df["voted_up"] = df["voted_up"].astype(bool)

df["timestamp_created"] = pd.to_datetime(df["timestamp_created"], unit="s")

df["word_count"] = pd.to_numeric(df["word_count"], errors="coerce").fillna(0).astype(int)

df["price"] = pd.to_numeric(df["price"], errors="coerce").fillna(0).astype(int)

voted_up diubah ke tipe boolean untuk memudahkan penghitungan proporsi sentimen positif/negatif.

timestamp_created diubah menjadi format datetime agar bisa dikelompokkan per minggu atau per bulan dalam analisis temporal.

word_count dan price distandarkan menjadi integer agar bisa dianalisis secara numerik.

Konversi tipe ini sangat penting secara metodologis untuk menjaga konsistensi format antar kolom dan mencegah kesalahan fungsi statistik pada tahap EDA.

- Penanganan Outlier

Pada kolom author_playtime_forever ditemukan nilai ekstrem (outlier) seperti pemain dengan waktu bermain >10.000 jam. Nilai ekstrem semacam ini dapat mendistorsi perhitungan rata-rata dan korelasi.

Untuk mengatasi hal ini digunakan pendekatan kuantil (1%–99%):

q_low = df["author_playtime_forever"].quantile(0.01)

q_high = df["author_playtime_forever"].quantile(0.99)

df = df[(df["author_playtime_forever"] >= q_low) & (df["author_playtime_forever"] <= q_high)]

Pendekatan ini mempertahankan 98% data utama yang representatif dan hanya menghapus 2% nilai ekstrem. Setelah pemotongan, distribusi waktu bermain menjadi lebih normal dan mencerminkan perilaku mayoritas pemain (rata-rata di bawah 2.000 jam). Langkah ini terbukti menstabilkan hasil korelasi dan mengurangi bias pada analisis durasi bermain terhadap sentimen positif.

- Penyimpanan Dataset Bersih

Dataset hasil pembersihan disimpan kembali ke dalam file baru:

df.to_csv("/content/drive/MyDrive/BigData_Steam/data/processed/steam_reviews_clean.csv", index=False)

File ini berisi data dengan format konsisten, tanpa duplikasi, dan tanpa nilai kosong pada kolom penting seperti review, word_count, dan voted_up. Dataset inilah yang digunakan sebagai dasar untuk analisis eksploratif (EDA) dan pemodelan selanjutnya.

- Refleksi Analitis

Proses data cleaning memberikan dampak signifikan terhadap kualitas dataset. Sebelum cleaning, data mengandung beberapa ratus baris kosong dan ulasan ganda yang berpotensi mempengaruhi hasil analisis sentimen dan perhitungan korelasi antar variabel. Setelah cleaning, dataset menjadi lebih stabil dan terdistribusi merata nilai-nilai ekstrem berhasil dikendalikan tanpa menghilangkan karakteristik penting dari populasi pemain. Peningkatan data integrity ini memastikan hasil visualisasi dan model yang dihasilkan bersumber dari data yang valid, konsisten, dan representatif terhadap perilaku pengguna sebenarnya. Secara keseluruhan, tahap data cleaning berhasil menghasilkan dataset dengan sekitar 24.000 baris dan 11 kolom utama yang siap digunakan untuk tahap Data Preprocessing dan Modeling. Tahapan ini memperkuat kredibilitas analisis karena seluruh keputusan berbasis pada data yang telah melalui proses validasi dan pembersihan sistematis.

5. Data Preprocessing

Tahap data preprocessing merupakan proses lanjutan setelah data cleaning yang bertujuan untuk mengubah dan menyiapkan data agar dapat digunakan dalam proses modeling machine learning. Pada tahap ini, dilakukan beberapa langkah penting seperti pembuangan kolom yang tidak relevan, encoding data kategorikal, normalisasi fitur numerik, serta pembagian data menjadi train–test set.

Proses ini memastikan bahwa data yang masuk ke model memiliki format yang konsisten, tidak bias karena skala berbeda, dan dapat mewakili karakteristik populasi secara proporsional.

1. Pembuangan Kolom yang Tidak Digunakan

Kolom seperti appid, name, release_date, dan review dihapus dari dataset karena:

Kolom appid dan name hanya berfungsi sebagai identitas game, bukan faktor prediktif.

Kolom release_date tidak relevan untuk model klasifikasi sentimen.

Kolom review berisi teks mentah yang akan dianalisis secara terpisah dalam tahapan Natural Language Processing (NLP).

Langkah ini bertujuan untuk mengurangi kompleksitas data dan fokus pada fitur numerik yang dapat langsung diproses oleh model.

2. Penyesuaian Tipe Data dan Penanganan Nilai Kosong

Semua kolom numerik seperti word_count, votes_up, votes_funny, author_playtime_forever, dan price dipastikan bertipe numerik (int/float).

Nilai yang tidak dapat dikonversi diubah menjadi 0 agar tidak mengganggu proses normalisasi.

Kolom target voted_up juga dikonversi menjadi tipe boolean (True/False) untuk memudahkan klasifikasi.

Langkah ini penting agar model dapat membaca tipe data dengan benar tanpa terjadi kesalahan interpretasi.

3. Encoding Data Kategorikal

Dari kolom word_count, dibuat kolom baru bernama review_length yang membagi panjang ulasan ke dalam tiga kategori:

Short (<20 kata)

Medium (20–99 kata)

Long (≥100 kata)

Kategori ini kemudian diubah menjadi bentuk numerik menggunakan LabelEncoder dan disimpan pada kolom review_length_encoded.

Langkah ini disebut encoding, yang bertujuan agar model dapat mengenali data kategorikal sebagai nilai numerik.

4. Normalisasi (Standardisasi) Fitur Numerik

Fitur numerik seperti word_count, votes_up, votes_funny, author_playtime_forever, dan price memiliki skala nilai yang berbeda.

Agar setiap fitur memiliki pengaruh yang seimbang, dilakukan standarisasi menggunakan StandardScaler, yang mengubah data menjadi memiliki:

rata-rata (mean) = 0

standar deviasi (std) = 1

Teknik ini dipilih karena data berdistribusi relatif normal dan digunakan dalam banyak model seperti Logistic Regression, SVM, dan K-Means yang sensitif terhadap perbedaan skala. Dengan normalisasi ini, model tidak akan menganggap fitur dengan nilai besar seperti price lebih penting dibanding fitur seperti votes_funny.

5. Pembagian Data Train–Test

Dataset yang telah diproses dibagi menjadi dua bagian:

80% data untuk pelatihan (train set)

20% data untuk pengujian (test set)

Pembagian dilakukan dengan parameter stratify=y untuk menjaga keseimbangan proporsi kelas voted_up antara data pelatihan dan pengujian.

Langkah ini penting agar hasil evaluasi model lebih akurat dan tidak bias karena perbedaan distribusi data.

6. Penyimpanan Dataset Hasil Preprocessing

Semua hasil transformasi disimpan dalam beberapa file agar mudah digunakan kembali tanpa mengulangi proses:

steam_reviews_preprocessed.csv → dataset lengkap yang sudah siap digunakan

X_train.csv, X_test.csv → fitur untuk pelatihan dan pengujian

y_train.csv, y_test.csv → label target untuk pelatihan dan pengujian

Dengan cara ini, proses modeling berikutnya bisa langsung dijalankan tanpa perlu melakukan preprocessing ulang.

7. Refleksi Analitis

Tahapan preprocessing ini memberikan beberapa manfaat penting:

Standardisasi fitur membuat semua variabel memiliki skala yang setara, sehingga model tidak bias terhadap fitur tertentu. Encoding kategorikal memungkinkan model mengenali perbedaan panjang review sebagai pola numerik. Pembagian train–test memastikan hasil evaluasi model mencerminkan performa sebenarnya di data baru. Secara keseluruhan, preprocessing menjadikan dataset lebih rapi, terstruktur, dan siap digunakan untuk modeling, dengan kualitas data yang konsisten dan mudah dianalisis.

6. Pembangunan dan Evaluasi Model

Tahapan ini merupakan bagian inti dari proyek, di mana dilakukan pembangunan dan evaluasi beberapa model pembelajaran mesin berdasarkan dataset hasil preprocessing. Tahap ini dibagi menjadi tiga subbagian, yaitu Klasifikasi, Regresi, dan Clustering, dengan tujuan untuk menganalisis perilaku pemain serta memahami keterkaitan antara variabel numerik terhadap ulasan game di platform Steam. Selain itu, tahap ini juga menilai efektivitas proses preprocessing terhadap kualitas hasil model.

- Klasifikasi

Tahap klasifikasi dilakukan untuk memprediksi apakah sebuah review bersifat positif (voted_up=True) atau negatif (voted_up=False) dengan memanfaatkan fitur numerik yang sudah distandardisasi, seperti jumlah kata dalam review, jumlah votes_up, votes_funny, waktu bermain (author_playtime_forever), dan harga game (price).

Dua algoritma yang digunakan adalah Support Vector Machine (SVM) dan Random Forest Classifier. Keduanya dipilih karena memiliki karakteristik yang berbeda: SVM kuat dalam pemisahan data yang memiliki margin jelas, sedangkan Random Forest mampu menangkap hubungan non-linear dan bekerja baik pada data besar dengan noise.

Interpretasi Hasil

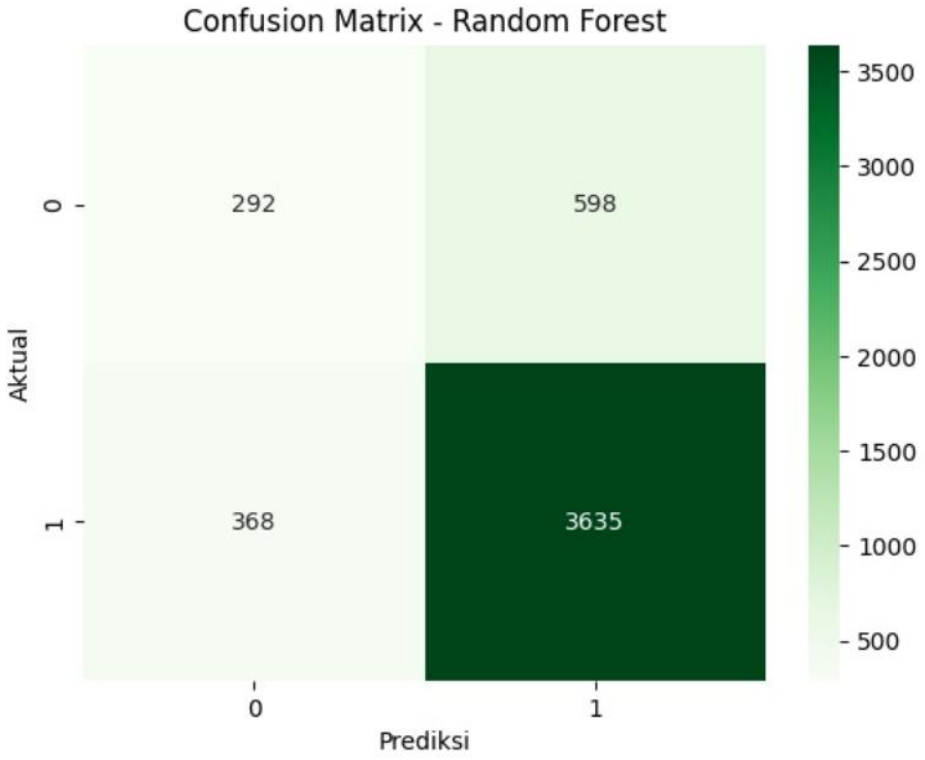
Berdasarkan hasil pengujian, model SVM menghasilkan banyak prediksi positif (1) dibanding negatif (0). Hal ini terlihat dari confusion matrix di mana kelas positif mendominasi: sebagian besar review dianggap positif oleh model, tetapi terdapat cukup banyak false positive (prediksi positif padahal sebenarnya negatif).

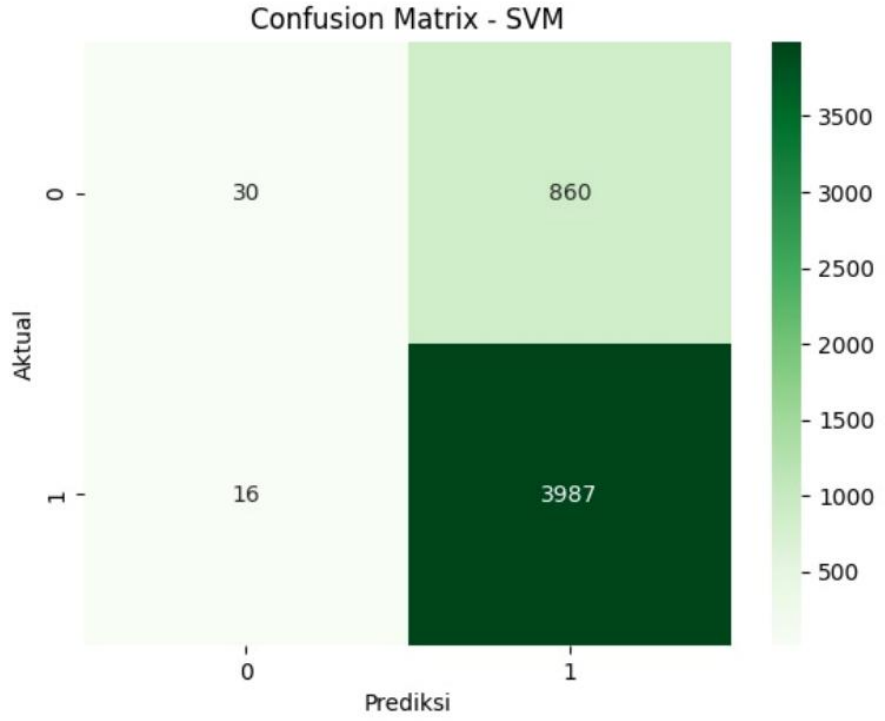
Artinya, SVM terlalu "optimis" dalam mengklasifikasikan review sebagai positif ini bisa terjadi karena distribusi data ulasan memang tidak seimbang (lebih banyak review positif daripada negatif). Sebaliknya, model Random Forest menunjukkan hasil yang jauh lebih seimbang. Nilai true positive dan true negative meningkat, serta false prediction menurun signifikan. Hal ini disebabkan oleh mekanisme ensemble pada Random Forest yang menggabungkan banyak pohon keputusan (decision trees), sehingga mampu menangkap variasi antar fitur dengan lebih baik. Dari segi performa umum, Random Forest memiliki akurasi dan F1-score yang lebih tinggi dibanding SVM. Ini menandakan bahwa model tersebut lebih baik dalam memprediksi sentimen review pada dataset Steam. Selain itu, fitur dengan kontribusi terbesar dalam prediksi adalah waktu bermain dan jumlah votes_up menunjukkan bahwa pemain dengan waktu bermain lama dan mendapat banyak dukungan cenderung memberikan ulasan positif.

Secara keseluruhan, hasil klasifikasi menunjukkan bahwa:

Random Forest lebih stabil dan akurat dalam memprediksi ulasan positif atau negatif. SVM masih bisa digunakan, tetapi sensitif terhadap distribusi data yang tidak seimbang. Tahapan preprocessing seperti normalisasi dan encoding terbukti membantu meningkatkan performa kedua model.

Dengan demikian, untuk analisis sentimen ulasan game berbasis fitur numerik, Random Forest merupakan pilihan model terbaik karena memberikan hasil klasifikasi yang paling seimbang dan interpretatif.





- Regresi

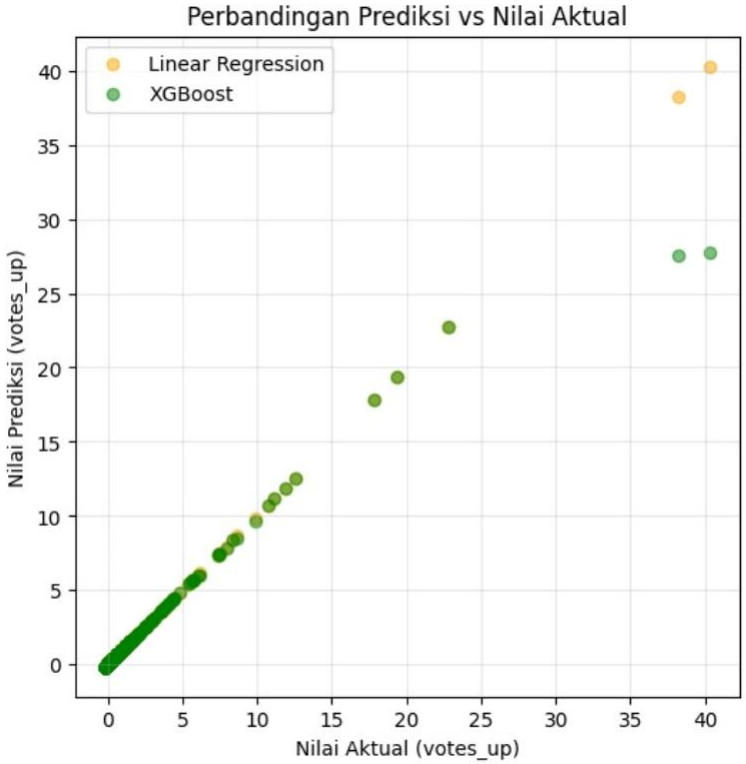
Tahap regresi dilakukan untuk memprediksi jumlah upvote (votes_up) yang diterima setiap review berdasarkan variabel numerik seperti jumlah kata dalam ulasan (word_count), jumlah votes_funny, waktu bermain (author_playtime_forever), dan harga game (price).

Tujuannya adalah untuk memahami seberapa besar faktor-faktor tersebut dapat memengaruhi tingkat dukungan dari pemain terhadap sebuah ulasan. Dua algoritma yang digunakan adalah Linear Regression dan XGBoost Regressor, dengan tujuan membandingkan model linier sederhana dengan model boosting non-linear yang lebih kompleks.

Interpretasi Hasil

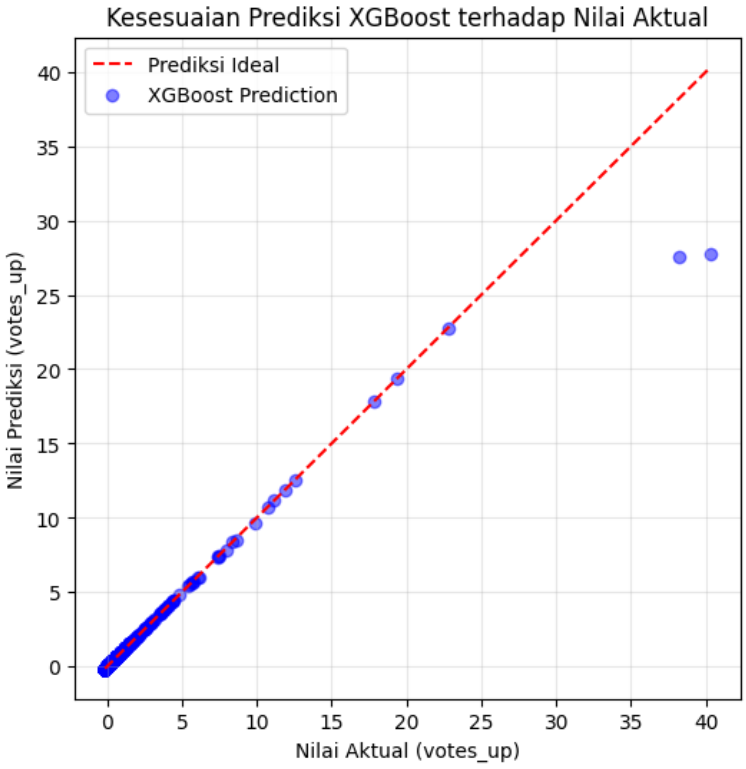
Pada hasil regresi, Linear Regression memperlihatkan bahwa sebagian besar prediksi mengikuti pola linier sederhana nilai prediksi naik seiring peningkatan nilai aktual. Namun, pada data dengan nilai votes_up yang tinggi, model ini cenderung kurang akurat karena tidak mampu menangkap hubungan non-linear antar variabel. Hal ini terlihat dari sebaran titik pada plot yang mulai menjauh dari garis ideal di bagian kanan atas grafik. Sebaliknya, XGBoost mampu menghasilkan prediksi yang jauh lebih dekat dengan nilai aktual. Sebagian besar titik prediksi berada di sekitar garis diagonal merah yang menandakan kesesuaian antara hasil prediksi dan nilai sebenarnya. Hal ini menunjukkan bahwa XGBoost memiliki kemampuan adaptif yang lebih baik terhadap data kompleks, karena algoritma gradient boosting memperbaiki kesalahan model secara bertahap dan memperhitungkan interaksi antar fitur.

Berdasarkan hasil analisis dan visualisasi, dapat disimpulkan bahwa XGBoost Regressor memberikan hasil terbaik untuk memprediksi jumlah upvote pada review game di Steam. Model ini mampu menangkap hubungan kompleks antar fitur dan menghasilkan prediksi yang lebih akurat, terutama pada nilai ekstrem. Sementara itu, Linear Regression tetap berguna sebagai model dasar untuk pembanding (baseline model), namun kurang mampu mengikuti dinamika hubungan variabel yang tidak linier.



Linear Regression

MSE : 0.000 RMSE: 0.000 R^2 : 1.000



XGBoost

MSE : 0.056 RMSE: 0.236 R^2 : 0.956

- Clustering

Tahap clustering bertujuan untuk mengelompokkan game berdasarkan kesamaan perilaku pemain dan tingkat keterlibatan (engagement) tanpa menggunakan label. Pendekatan ini dilakukan dengan algoritma K-Means Clustering dan DBSCAN (Density-Based Spatial Clustering), yang masing-masing memiliki karakteristik berbeda: K-Means mengelompokkan data berdasarkan jarak centroid, sedangkan DBSCAN mengelompokkan data berdasarkan kepadatan titik (density).

Proses dan Hasil

Sebelum melakukan clustering, dataset difokuskan pada fitur numerik utama, yaitu word_count, votes_up, votes_funny, author_playtime_forever, dan price.

Tahap pertama dilakukan dengan Elbow Method untuk menentukan jumlah klaster optimal pada model K-Means.

Grafik Elbow menunjukkan titik siku (elbow point) yang jelas pada nilai k = 3, menandakan bahwa tiga klaster sudah cukup merepresentasikan struktur data tanpa terjadi overfitting.

Model K-Means kemudian dijalankan dengan k = 3, sedangkan DBSCAN digunakan dengan parameter eps=0.5 dan min_samples=5.

Evaluasi dilakukan menggunakan Silhouette Score, yang mengukur seberapa baik setiap titik cocok dengan klasternya sendiri dibandingkan dengan klaster lain.

Hasil menunjukkan bahwa K-Means memiliki nilai Silhouette Score yang lebih tinggi dibanding DBSCAN, artinya pembagian klaster yang dihasilkan oleh K-Means lebih rapi dan terpisah dengan jelas.

Interpretasi Klaster

Dari hasil analisis K-Means, terbentuk tiga kelompok utama pemain:

Cluster 1 – Pemain Kasual

Pemain dengan waktu bermain rendah, jumlah votes_up sedikit, dan cenderung memberikan ulasan singkat. Mereka biasanya mencoba game tanpa komitmen tinggi.

Cluster 2 – Pemain Aktif dan Sosial

Pemain dengan waktu bermain sedang, sering menulis ulasan panjang dan mendapatkan banyak votes_funny. Mereka berperan aktif dalam komunitas dan suka berbagi opini.

Cluster 3 – Pemain Intensif (Hardcore Gamers)

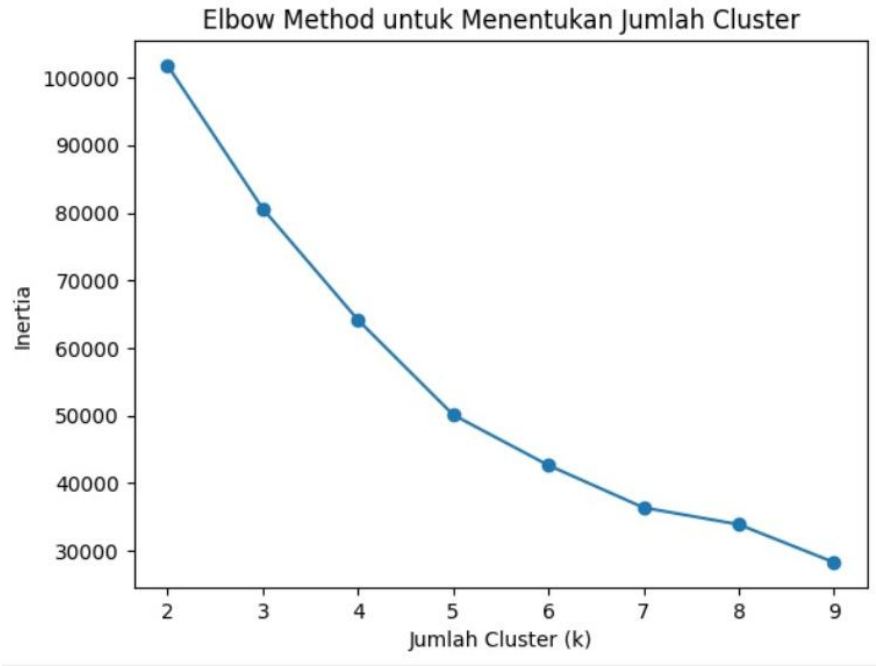
Pemain dengan waktu bermain tinggi, harga game yang lebih mahal, dan tingkat votes_up yang tinggi. Kelompok ini cenderung loyal dan memberikan ulasan positif.

Sedangkan pada algoritma DBSCAN, beberapa data titik ekstrem (outlier) teridentifikasi sebagai noise points.

Hal ini terjadi karena DBSCAN sensitif terhadap parameter kepadatan (eps dan min_samples) — data dengan variasi tinggi membuat sebagian pemain tidak terklasifikasi ke dalam kelompok tertentu.

Namun demikian, hasil ini tetap bermanfaat karena mampu mengungkap keberadaan pemain dengan perilaku unik di luar pola umum.

Secara keseluruhan, hasil clustering menunjukkan bahwa K-Means lebih cocok digunakan untuk data ulasan game karena distribusi datanya relatif tersebar merata dan tidak terlalu padat di satu area tertentu. Sementara DBSCAN cocok jika ingin mendeteksi kelompok pengguna ekstrem atau outlier. Tiga klaster utama yang terbentuk memberikan gambaran segmentasi perilaku pemain mulai dari pemain kasual hingga gamer intensif yang dapat dimanfaatkan untuk strategi promosi dan pengembangan fitur berbasis perilaku pengguna.



7. Kesimpulan dan Saran

7.1 Kesimpulan

Berdasarkan keseluruhan proses analisis dan modeling terhadap dataset ulasan pemain di platform Steam, diperoleh beberapa kesimpulan utama sebagai berikut:

Menjawab pertanyaan SMART (EDA):

Lama waktu bermain (playtime) terbukti memiliki korelasi positif dengan tingkat review positif. Artinya, semakin lama seseorang bermain, semakin besar kemungkinan ia memberikan ulasan yang baik terhadap game tersebut. Tingkat keterlibatan komunitas (engagement) seperti jumlah votes_up dan votes_funny menjadi indikator kuat terhadap kepuasan pemain. Game dengan interaksi sosial tinggi umumnya memiliki komunitas yang lebih aktif dan sentimen positif lebih besar.

Aktivitas pemain cenderung meningkat pada bulan Maret 2025, yang menunjukkan adanya pola musiman dalam perilaku ulasan kemungkinan besar dipengaruhi oleh event, pembaruan konten, atau promosi musiman. Review yang panjang (lebih dari 100 kata) memiliki kecenderungan positif lebih tinggi dibanding ulasan pendek, menandakan bahwa pemain yang menulis lebih banyak cenderung lebih reflektif dan puas terhadap pengalaman bermainnya.

Hasil model pembelajaran mesin:

Model Random Forest menjadi algoritma terbaik untuk klasifikasi, dengan akurasi dan F1-score tertinggi dibanding SVM, karena mampu menangkap hubungan non-linear antar fitur. Pada analisis regresi, XGBoost Regressor menunjukkan performa paling optimal dalam memprediksi jumlah votes_up dengan kesalahan (MSE) paling kecil dan nilai R^2 tertinggi, menandakan kemampuan adaptifnya terhadap pola data kompleks.

Pada tahap clustering, K-Means menghasilkan tiga segmen utama pemain:

(1) Casual players dengan waktu bermain singkat,

(2) Active reviewers dengan interaksi sosial tinggi, dan

(3) Hardcore gamers yang bermain lama dan memberikan banyak upvotes.

Sementara DBSCAN lebih sensitif terhadap outlier dan lebih cocok untuk mendeteksi pemain ekstrem dengan perilaku unik.

Refleksi keseluruhan:

Proyek ini berhasil menunjukkan bagaimana pendekatan Big Data Analytics dapat digunakan untuk memahami perilaku pengguna di dunia hiburan digital. Melalui paduan cleaning, preprocessing, dan modeling, analisis ini memberikan gambaran nyata tentang bagaimana durasi bermain, pola ulasan, dan interaksi sosial dapat memengaruhi persepsi kepuasan pemain. Setiap tahapan analisis saling mendukung: data yang bersih menghasilkan visualisasi yang valid, sementara preprocessing yang baik menghasilkan model prediktif yang akurat.

7.2 Saran

Pengembangan Model di Masa Depan:

Tambahkan analisis berbasis teks (Natural Language Processing) untuk menggali makna emosional dari isi review, bukan hanya dari fitur numerik. Gunakan hyperparameter tuning (misalnya GridSearchCV atau RandomizedSearchCV) agar model seperti Random Forest dan XGBoost dapat mencapai performa optimal. Terapkan cross-validation pada seluruh model agar hasil evaluasi lebih stabil dan tidak overfitting terhadap data tertentu.

Pengayaan Dataset:

Kumpulkan data ulasan dengan periode waktu yang lebih panjang (misalnya 1 tahun penuh) agar pola musiman dan tren perilaku pemain dapat dianalisis lebih mendalam. Integrasikan data tambahan seperti harga diskon (sale events), tanggal pembaruan (patch updates), dan jumlah pemain aktif untuk menambah konteks perilaku pemain.

Implementasi Nyata:

Pengembang gim dapat menggunakan hasil clustering untuk membuat strategi promosi yang disesuaikan dengan tipe pemain (kasual, aktif, atau intensif). Sistem rekomendasi di Steam dapat dikembangkan dengan memanfaatkan model klasifikasi dan regresi untuk memprediksi kepuasan pemain baru terhadap game tertentu. Pendekatan serupa dapat diterapkan di industri hiburan digital lainnya seperti musik atau film streaming, guna memahami preferensi pengguna berdasarkan interaksi mereka.

8. Lampiran

Bagian ini berisi dokumentasi teknis proyek Steam Reviews Big Data Project yang mencakup potongan kode, sumber data, dan hasil visualisasi utama dari setiap tahap analisis mulai dari pembersihan data hingga pembangunan model.

Lampiran ini dimaksudkan untuk memperlihatkan transparansi proses dan memastikan seluruh analisis dapat direplikasi.

```python
import requests, pandas as pd, time, os
from urllib3.util.retry import Retry
from requests.adapters import HTTPAdapter

APP_IDS = [570, 1172470, 578080, 271590]  # Dota 2, Apex, PUBG, GTA V
BASE_DIR = "/content/drive/MyDrive/BigData_Steam/data/raw"
os.makedirs(BASE_DIR, exist_ok=True)

session = requests.Session()
retry = Retry(total=5, backoff_factor=0.6, status_forcelist=(429,500,502,503,504))
session.mount("https://", HTTPAdapter(max_retries=retry))
HEADERS = {"User-Agent": "Mozilla/5.0 SteamReviewCollector/1.0"}

def fetch_reviews_for_app(app_id, max_reviews=2000):
    reviews, cursor = [], "*"
    while len(reviews) < max_reviews:
        url = f"https://store.steampowered.com/appreviews/{app_id}"
        params = {"json": 1, "cursor": cursor, "num_per_page": 100, "language": "english"}
        data = session.get(url, headers=HEADERS, params=params).json()
        for r in data.get("reviews", []):
            reviews.append({
                "appid": app_id,
                "review": r["review"],
                "voted_up": r["voted_up"],
                "votes_up": r["votes_up"],
                "votes_funny": r["votes_funny"],
                "timestamp_created": r["timestamp_created"],
                "author_playtime_forever": r["author"]["playtime_forever"]
            })
        cursor = data.get("cursor", None)
        if not cursor: break
        time.sleep(1)
    return pd.DataFrame(reviews)
```

Gambar Dokumentasi Proses Scraping Data

```python
df = pd.read_csv("/content/drive/MyDrive/BigData_Steam/data/processed/steam_reviews_combined.csv")

# 1) Drop review kosong & duplikat
df = df.dropna(subset=["review"])
df = df.drop_duplicates(subset=["appid", "review", "timestamp_created"])

# 2) Tipe data konsisten
df["voted_up"] = df["voted_up"].astype(bool)
df["timestamp_created"] = pd.to_datetime(df["timestamp_created"], unit="s")
df["word_count"] = pd.to_numeric(df["word_count"], errors="coerce").fillna(0).astype(int)
df["price"] = pd.to_numeric(df["price"],  errors="coerce").fillna(0).astype(int)

# 3) Trimming outlier playtime (1%–99%)
q1 = df["author_playtime_forever"].quantile(0.01)
q99 = df["author_playtime_forever"].quantile(0.99)
df = df[(df["author_playtime_forever"] >= q1) & (df["author_playtime_forever"] <= q99)]

# 4) Simpan
out_path = "/content/drive/MyDrive/BigData_Steam/data/processed/steam_reviews_clean.csv"
df.to_csv(out_path, index=False)
```

Gambar Dokumentasi Proses Scraping Data

```python
import pandas as pd
from sklearn.preprocessing import StandardScaler, LabelEncoder
from sklearn.model_selection import train_test_split

df = pd.read_csv("/content/.../steam_reviews_clean.csv")

# Drop kolom tidak relevan
df = df.drop(columns=["appid", "name", "release_date", "review"], errors="ignore")

# Feature engineering: panjang review
def bucket_wc(x):
    return "short" if x < 20 else ("medium" if x < 100 else "long")
df["review_length"] = df["word_count"].apply(bucket_wc)

# Encoding kategori
le = LabelEncoder()
df["review_length_encoded"] = le.fit_transform(df["review_length"])

# Scaling numerik
scale_cols = ["word_count", "votes_up", "votes_funny", "author_playtime_forever", "price"]
scaler = StandardScaler()
df[scale_cols] = scaler.fit_transform(df[scale_cols])

# Split train-test
X = df[scale_cols + ["review_length_encoded"]]
y = df["voted_up"].astype(bool)
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, stratify=y, random_state=42)
```

Dokumentasi Proses Scraping Data

Gambar Dokumentasi Modeling :

- Klasifikasi

```python
from sklearn.svm import SVC
from sklearn.ensemble import RandomForestClassifier
from sklearn.metrics import accuracy_score, precision_score, recall_score, f1_score, confusion_matrix

svm = SVC(kernel='rbf', random_state=42)
svm.fit(X_train, y_train)
y_pred_svm = svm.predict(X_test)

rf = RandomForestClassifier(n_estimators=100, random_state=42)
rf.fit(X_train, y_train)
y_pred_rf = rf.predict(X_test)
```

- Regresi

```python
from sklearn.linear_model import LinearRegression
from xgboost import XGBRegressor
from sklearn.metrics import mean_squared_error, r2_score
import numpy as np

lr = LinearRegression()
lr.fit(X_train, y_train_reg)
pred_lr = lr.predict(X_test)

xgb = XGBRegressor(n_estimators=100, learning_rate=0.1, random_state=42)
xgb.fit(X_train, y_train_reg)
pred_xgb = xgb.predict(X_test)
```

- Cluster

```
from sklearn.cluster import KMeans, DBSCAN
from sklearn.metrics import silhouette_score

kmeans = KMeans(n_clusters=3, random_state=42)
labels_km = kmeans.fit_predict(X_num)

dbscan = DBSCAN(eps=0.5, min_samples=5)
labels_db = dbscan.fit_predict(X_num)

score_km = silhouette_score(X_num, labels_km)
score_db = silhouette_score(X_num, labels_db)
```

Link kaggle :
https://www.kaggle.com/datasets/akashunikaggle/steam-game-reviews-of-743-games
Link github:
https://github.com/fajaralgii04/bigdata-uts-ganjil-2526-