

**ANALISIS KINERJA ALGORITMA LOGISTIC REGRESSION
DAN SVM (SUPPORT VECTOR MACHINE) PADA
SENTIMEN ULASAN PERMAINAN MOBILE E-FOOTBALL**

DISUSUN OLEH

ATHARY ZIKRY

2109010139



**PROGRAM STUDI SISTEM INFORMASI
FAKULTAS ILMU KOMPUTER DAN TEKNOLOGI INFORMASI
UNIVERSITAS MUHAMMADIYAH SUMATERA UTARA
MEDAN
2026**

**ANALISIS KINERJA ALGORITMA LOGISTIC REGRESSION
DAN SVM (SUPPORT VECTOR MACHINE) PADA
SENTIMEN ULASAN PERMAINAN MOBILE E-FOOTBALL**

SKRIPSI

Diajukan sebagai salah satu syarat untuk memperoleh gelar Sarjana Komputer (S.Kom)
dalam Program Studi Sistem Informasi pada Fakultas Ilmu Komputer dan Teknologi
Informasi, Universitas Muhammadiyah Sumatera Utara

DISUSUN OLEH

Athary

2109010139

**PROGRAM STUDI SISTEM INFORMASI
FAKULTAS ILMU KOMPUTER DAN TEKNOLOGI INFORMASI
UNIVERSITAS MUHAMMADIYAH SUMATERA UTARA**

MEDAN

2026

LEMBAR PENGESAHAN

Judul Skripsi : Analisis Kinerja Algoritma Logistic Regression dan SVM
(Support Vector Machine) Pada Sentimen Ulasan
Permainan Mobile E-Football

Nama Mahasiswa : Athary Zikry

NPM : 2109010139

Program Studi : Sistem Informasi

Menyetujui
Komisi Pembimbing



(Farid Akbar Siregar, M.Kom)
NIDN. 0103039401

Ketua Program Studi



(Mahardika Abdi Prawira Tanjung,
S.Kom., M.Kom)
NIDN. 0117088902

Dekan



(Dr. Al-Khowarizmi, S.Kom., M.Kom.)
NIDN. 0127099201

PERNYATAAN ORISINALITAS

**ANALISIS KINERJA ALGORITMA LOGISTIC REGRESSION
DAN SVM (SUPPORT VECTOR MACHINE) PADA
SENTIMEN ULASAN PERMAINAN MOBILE E-FOOTBALL**

SKRIPSI

Saya menyatakan bahwa karya tulis ini adalah hasil karya sendiri, kecuali beberapa kutipan dan ringkasan yang masing-masing disebutkan sumbernya.

Medan, Januari 2026

Yang membuat pernyataan



ATHARY ZIKRY

2109010139

**PERNYATAAN PERSETUJUAN PUBLIKASI
KARYA ILMIAH UNTUK KEPENTINGAN
AKADEMIS**

Sebagai sivitas akademika Universitas Muhammadiyah Sumatera Utara, saya bertanda tangan dibawah ini:

Nama : Athary Zikry
NPM : 2109010139
Program Studi : Sistem Informasi
Karya Ilmiah : Skripsi

Demi pengembangan ilmu pengetahuan, menyetujui untuk memberikan kepada Universitas Muhammadiyah Sumatera Utara Hak Bedas Royalti Non-Eksekutif (*Non-Exclusive Royalty free Right*) atas penelitian skripsi saya yang berjudul:

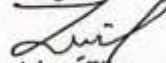
**ANALISIS KINERJA ALGORITMA LOGISTIC REGRESSION DAN SVM
(SUPPORT VECTOR MACHINE) PADA SENTIMEN ULASAN
PERMAINAN MOBILE E-FOOTBALL**

Beserta perangkat yang ada (jika diperlukan). Dengan Hak Bebas Royalti Non-Eksekutif ini, Universitas Muhammadiyah Sumatera Utara berhak menyimpan, mengalih media, memformat, mengelola dalam bentuk database, merawat dan mempublikasikan Skripsi saya ini tanpa meminta izin dari saya selama tetap mencantumkan nama saya sebagai penulis dan sebagai pemegang dan atau sebagai pemilik hak cipta.

Demikian pernyataan ini dibuat dengan sebenarnya.

Medan, Januari 2026

Yang membuat pernyataan


Athary Zikry

2109010139

RIWAYAT HIDUP

DATA PRIBADI

Nama Lengkap : Athary Zikry
Tempat dan Tanggal Lahir : Medan, 30 September 2003
Alamat Rumah : Komplek Griya Marelan Blok F-3
Telepon/Faks/HP : 089636366748
E-mail : atharyzikry3009@gmail.com
Instansi Tempat Kerja : Belum Bekerja
Alamat Kantor : -

DATA PENDIDIKAN

SD : SD WAHIDIN SUDIRO HUSODO TAMAT: 2015
SMP : SMP PERTIWI MEDAN TAMAT: 2018
SMA : SMK TRITECH INFORMATIKA TAMAT: 2021

KATA PENGANTAR



Puji dan syukur penulis panjatkankehadirat Allah SWT atas limpah rahmat, taufik, dan hidayah-Nya sehingga penulis dapat menyelesaikan skripsi yang berjudul **“Analisis Kinerja Algoritma Logistic Regression dan Support Vector Machine (SVM) Pada Sentimen Ulasan Permainan Mobile E-Football”**

Penulis tentunya berterima kasih kepada berbagai pihak dalam dukungan serta doa dalam penyelesaian skripsi. Penulis juga mengucapkan terima kasih kepada:

1. Bapak Prof. Dr. Akrim M.Pd, Rektor Universitas Muhammadiyah Sumatera Utara (UMSU)
2. Bapak Dr. Al-Khowarizmi, S.Kom., M.Kom. Dekan Fakultas Ilmu Komputer dan Teknologi Informasi (FIKTI) UMSU.
3. Ibu Dr. Firahti Rizky, S.Kom., M.Kom Wakil Dekan I Fakultas Ilmu Komputer dan Teknologi Informasi (FIKTI) UMSU.
4. Bapak Mhd Basri, S.Si., M. Kom. Wakil Dekan III Fakultas Ilmu Komputer dan Teknologi Informasi (FIKTI) UMSU.
5. Bapak Mahardika Abdi Prawira Tanjung, S.Kom., M.Kom. Ketua Program Studi Sistem Informasi Fakultas Ilmu Komputer dan Teknologi Informasi (FIKTI) UMSU.
6. Bapak Mulkan Azhari, S.Kom., M.Kom. Sekretaris Program Studi Sistem Informasi Fakultas Ilmu Komputer dan Teknologi Informasi (FIKTI) UMSU.
7. Bapak Farid Akbar Siregar M, Kom. selaku pembimbing saya yang telah memberikan bimbingan, arahan, dan ilmu yang berharga selama proses penyusunan skripsi ini.
8. Panutan terbesar dalam hidup penulis, Bapak Syamsul Bahri, SH.MH, Terima Kasih atas setiap doa yang tak pernah putus, pengorbanan, serta perjuangan demi masa depan penulis. Beliau adalah guru kehidupan

terbaik dengan keteladanan, motivasi, dan dukungan tanpa henti mampu mengantarkan penulis menyelesaikan studi hingga meraih gelar sarjana.

9. Pintu Surgaku, ibunda Henny S,ag, Kasih sayang dan doa beliau menjadi sumber kekuatan yang tak pernah terpisahkan dalam perjalanan penulis hingga mampu menyelesaikan pendidikan dan memperoleh gelar sarjana.
10. Terima kasih yang sebesar-besarnya kepada kakak dan abang, Dina Nasuha, SH dan Attirmidzi Ariq, SH yang telah memberikan dukungan penuh selama proses penyusunan skripsi ini. Bantuan yang diberikan baik dalam bentuk motivasi, perhatian, saran, maupun bantuan moril dan materi, sangat berarti bagi penulis dalam menghadapi berbagai kendala dan tantangan selama penelitian berlangsung.
11. Sahabat penulis yaitu Nazri, Arya, Fiqri, Tio, Zaini yang sudah selalu menemani, membantu, memberikan semangat dan mendengarkan cerita keluh kesah penulis selama proses pengerjaan skripsi ini.
12. Teruntuk Clarisa Insyira, terima kasih telah menjadi support system penulis, terimakasih atas dukungan, semangat, motivasi, kebaikan, perhatian dan kebijaksanaannya yang telah diberikan selama proses penyelesaian skripsi ini. Dan terimakasih telah menjadi bagian dari perjalanan hidup penulis.
13. Semua pihak yang terlibat langsung ataupun tidak langsung yang tidak dapat penulis ucapkan satu-persatu yang telah membantu penyelesaian skripsi ini.
14. Terakhir penulis ingin mengucapkan terimakasih kepada seseorang yang mungkin sering terlupakan, ya penulis ini ingin mengucapkan terimakasih kepada diri sendiri karena tetap bertahan sampai pada tahap ini. Terimakasih karena tidak menyerah dan berani melawan rasa takut, serta keraguan terbesar dalam diri, terimakasih untuk tidak menyerah tetapi memilih melangkah dan terus berjuang walau kadang tak tau arah dan sering merasa terlambat dari yang lain. Sampai ditahap ini bukanlah hal yang mudah bagi anak kecil yang perjuangannya dipenuhi dengan benturan besar, perjalanan mu sangat tidak mudah tapi kau sanggup

melewati badai itu. Saya bangga padamu, saya tau perjalanan ini belum selesai, masih banyak ketidakpastian dan luka yang mungkin datang tetapi semoga kamu mampu melewatinya dan ingat kamu pantas bahagia, kamu berhak bermimpi, dan kamu layak untuk sampai di tujuan itu. Teruslah hidup dengan niat yang baik. Terimakasih Athary Zikry, kamu hebat.

**ANALISIS KINERJA ALGORITMA LOGISTIC REGRESSION DAN SVM
(SUPPORT VECTOR MACHINE) PADA SENTIMEN ULASAN
PERMAINAN MOBILE E-FOOTBALL**

ABSTRAK

Perkembangan industri permainan mobile mendorong meningkatnya partisipasi pengguna dalam memberikan ulasan pada platform digital, salah satunya pada permainan mobile e-Football di Google Play Store. Ulasan tersebut mengandung berbagai opini yang mencerminkan tingkat kepuasan dan ketidakpuasan pengguna terhadap kualitas permainan. Penelitian ini bertujuan untuk menganalisis sentimen ulasan pengguna terhadap permainan mobile e-Football menggunakan algoritma Logistic Regression dan Support Vector Machine (SVM). Data yang digunakan dalam penelitian ini diperoleh dari ulasan pengguna Google Play Store yang dikumpulkan melalui teknik *web scraping*. Data kemudian diproses melalui tahapan *preprocessing* yang meliputi *cleaning*, *case folding*, normalisasi, tokenisasi, *stopword removal*, dan *stemming*. Selanjutnya, data teks direpresentasikan ke dalam bentuk numerik menggunakan metode Term Frequency–Inverse Document Frequency (TF-IDF). Evaluasi model dilakukan menggunakan metrik akurasi, presisi, recall, dan F1-score. Hasil penelitian menunjukkan bahwa algoritma Support Vector Machine (SVM) memiliki kinerja yang sedikit lebih baik dibandingkan Logistic Regression, dengan nilai akurasi sebesar 81,13%, sedangkan Logistic Regression memperoleh nilai akurasi sebesar 80,97%. Berdasarkan hasil tersebut, algoritma SVM dinilai lebih optimal dalam mengklasifikasikan sentimen ulasan pengguna terhadap permainan mobile e-Football.

Kata Kunci: Analisis Sentimen, E-Football, Support Vector Machine, Logistic Regression, TF-IDF

**ANALISIS KINERJA ALGORITMA LOGISTIC REGRESSION DAN SVM
(SUPPORT VECTOR MACHINE) PADA SENTIMEN ULASAN
PERMAINAN MOBILE E-FOOTBALL**

ABSTRACT

The rapid growth of the mobile gaming industry has increased user participation in providing reviews on digital platforms, including the mobile game e-Football on Google Play Store. These reviews contain various opinions that reflect user satisfaction and dissatisfaction with the game quality. This study aims to analyze user sentiment toward the mobile e-Football game using Logistic Regression and Support Vector Machine (SVM) algorithms. The data used in this study were collected from Google Play Store user reviews through a web scraping technique. The data were processed through several preprocessing stages, including cleaning, case folding, normalization, tokenization, stopword removal, and stemming. Furthermore, the textual data were transformed into numerical features using the Term Frequency–Inverse Document Frequency (TF-IDF) method. Model performance was evaluated using accuracy, precision, recall, and F1-score metrics. The results indicate that the Support Vector Machine (SVM) algorithm achieved slightly better performance than Logistic Regression, with an accuracy of 81.13%, while Logistic Regression achieved an accuracy of 80.97%. Therefore, SVM can be considered a more optimal method for sentiment classification of mobile e-Football user reviews.

Keywords: *Sentiment Analysis; e-Football; Support Vector Machine; Logistic Regression; TF-IDF.*

DAFTAR ISI

LEMBAR PENGESAHAN	ii
PERNYATAAN ORISINALITAS.....	iii
PERNYATAAN PERSETUJUAN PUBLIKASI.....	iv
KARYA ILMIAH UNTUK KEPENTINGAN AKADEMIS	iv
RIWAYAT HIDUP	v
KATA PENGANTAR.....	vi
ABSTRAK	viii
ABSTRACT	ix
DAFTAR ISI.....	x
DAFTAR TABEL.....	xii
DAFTAR GAMBAR.....	xiii
BAB I PENDAHULUAN	1
1.1 Latar Belakang Masalah	1
1.2 Rumusan Masalah.....	2
1.3 Batasan Masalah	2
1.4 Tujuan Penelitian	3
1.5 Manfaat Penelitian	3
BAB II LANDASAN TEORI	4
2.1 Data dan Informasi.....	4
2.2 Text Mining	5
2.3 Analisis Sentimen	6
2.4 Preprocessing Teks	7
2.5 Logistic Regression.....	9
2.6 Support Vector Machine (SVM).....	11
2.7 Confusion Matrix.....	14
2.8 Unified Modeling Language (UML)	14
2.8.1 Jenis Diagram UML yang Digunakan	15
2.9 Google Colab	16
2.10 Penelitian Terdahulu	17
BAB III METODOLOGI PENELITIAN	19
3.1 Jenis dan Pendekatan Penelitian	19
3.2 Alur Penelitian	19
3.3 Lokasi dan Objek Penelitian	21
3.4 Sumber Data	21
3.5 Teknik Pengumpulan Data.....	22
3.6 Preprocessing Data	22
3.6.1 Cleaning.....	23
3.6.2 Case Folding	23
3.6.3 Normalisasi Kata	24
3.6.4 Tokenizing.....	24
3.6.5 Stopwords	25
3.6.6 Stemming.....	25
3.7 Ekstraksi Fitur.....	26
3.8 Pembagian Data (Data Splitting)	27
3.9 Metode Klasifikasi.....	27

3.9.1 Logistic Regression	27
3.9.2 Support Vector Machine (SVM)	28
3.10 Evaluasi Model	28
3.11 Flowchart Penelitian	29
3.12 Use Case Diagram	30
3.13 Activity Diagram	31
BAB IV HASIL DAN PEMBAHASAN.....	32
4.1 Scrapping Data.....	32
4.2 Hasil Preprocessing Data	33
4.3 Hasil Pembobotan TF-IDF.....	33
4.4 Split Data	38
4.5 Hasil Klasifikasi Dengan Logistic Regression	38
4.6 Hasil Klasifikasi Dengan SVM (Support Vector Machine)	40
4.7 Hasil Evaluasi Logistic Regression	41
4.8 Hasil Evaluasi SVM (Support Vector Machine)	44
4.9 Hasil Visualisasi	46
4.9.1 Hasil Visualisasi Logistic Regression	47
4.9.2 Hasil Visualisasi SVM (Support Vector Machine)	48
4.10 Analisis Kinerja Algoritma Logistic Regression Dan SVM (Support Vector Machine)	49
BAB V KESIMPULAN DAN SARAN	51
5.1 Kesimpulan	51
5.2 Saran	52
DAFTAR PUSTAKA	53
LAMPIRAN	

DAFTAR TABEL

Tabel 2.1 Penelitian Terdahulu	17
Tabel 3.1 Cleaning	23
Tabel 3.2 Case Folding	12
Tabel 3.3 Normalisasi Kata.....	24
Tabel 3.4 Tokenizing	25
Tabel 3.5 Stopwords.....	25
Tabel 3.6 Stemming	26
Tabel 4.1 Confusion Matrix Tanpa Nilai	42
Tabel 4.2 Confusion Matrix Keseluruhan.....	42
Tabel 4.3 Confusion Matrix Keseluruhan.....	42
Tabel 4.4 Confusion Matrix Keseluruhan.....	43
Tabel 4.5 Confusion Matrix Tanpa Nilai	44
Tabel 4.6 Confusion Matrix Keseluruhan.....	44
Tabel 4.7 Confusion Matrix Keseluruhan.....	45
Tabel 4.8 Confusion Matrix Keseluruhan.....	46

DAFTAR GAMBAR

Gambar 3.1 Flowchart Proses Analisis Sentimen	29
Gambar 3.2 Use Case Diagram	30
Gambar 3.3 Activity Diagram	31
Gambar 4.1 Hasil Scrapping Data	33
Gambar 4.2 Data Hasil Scrapping (CSV)	33
Gambar 4.3 Hasil Cleaning	34
Gambar 4.4 Hasil Case Folding	34
Gambar 4.5 Hasil Normalisasi Kata	35
Gambar 4.6 Hasil Tokenize	35
Gambar 4.7 Hasil Stopword Removal	36
Gambar 4.8 Hasil Stemming Data	36
Gambar 4.9 Diagram Frekuensi Data.....	37
Gambar 4.10 Hasil WorldCloud Stemming.....	37
Gambar 4.11 Hasil TF-IDF.....	38
Gambar 4.12 Hasil Split Data	38
Gambar 4.13 Hasil Klasifikasi Logistic Regression	39
Gambar 4.14 Confusion Matrix	39
Gambar 4.15 Hasil Klasifikasi SVM	40
Gambar 4.16 Confusion Matrix	40
Gambar 4.17 WorldCloud Sentimen Positif	46
Gambar 4.18 WorldCloud Sentimen Negatif.....	47
Gambar 4.19 WorldCloud Sentimen Positif	47
Gambar 4.20 WorldCloud Sentimen Negatif.....	48
Gambar 4.21 Perbandingan Akurasi dan F1-Score.....	49

BAB I

PENDAHULUAN

1.1 Latar Belakang Masalah

Industri mobile gaming di Indonesia telah bertransformasi dari sekadar sarana hiburan menjadi sektor ekonomi kreatif yang sangat masif. Berdasarkan data perkembangan digital, genre simulasi olahraga menempati posisi yang sangat strategis dalam ekosistem e-sport nasional. Permainan e-Football besutan Konami merupakan salah satu judul yang memiliki basis penggemar terbesar di Indonesia. Namun, transisi dari model bisnis konvensional ke model free-to-play serta perubahan mesin permainan (game engine) sering memicu polarisasi opini di kalangan pengguna. Fluktuasi kepuasan pemain E-Football yang tercermin dalam ulasan di Google Play Store merupakan indikator krusial yang menggambarkan bagaimana kualitas pembaruan perangkat lunak diterima oleh pasar secara luas

Ulasan-ulasan yang diberikan oleh pengguna di platform digital adalah bentuk data tekstual tidak terstruktur yang menyimpan informasi berharga mengenai pengalaman pengguna (user experience). Tantangan utama dalam mengolah data ini adalah bahasa yang digunakan pemain Indonesia cenderung tidak baku, menggunakan banyak singkatan, serta istilah-istilah slang yang spesifik dalam dunia sepak bola. Ulasan pada gim mobile memiliki kompleksitas linguistik yang tinggi, sehingga pemrosesan secara manual tidak lagi efektif untuk skala data yang mencapai puluhan ribu entri (Alengka, 2025). Oleh karena itu, penerapan teknologi Natural Language Processing (NLP) melalui analisis sentimen menjadi kebutuhan mendesak untuk mengubah data mentah tersebut menjadi pengetahuan yang dapat ditindaklanjuti oleh pengembang.

Dalam disiplin ilmu Data Science, efektivitas analisis sentimen sangat bergantung pada pemilihan model klasifikasi. Dua algoritma yang sering diperdebatkan dalam literatur akademik adalah Logistic Regression dan Support Vector Machine (SVM). Logistic Regression sering kali menjadi pilihan utama dalam klasifikasi biner karena efisiensi komputasinya yang tinggi dan

kemampuannya untuk memberikan interpretasi probabilitas yang jelas terhadap sebuah kelas sentimen. Dalam beberapa kasus klasifikasi teks berskala besar, kesederhanaan Logistic Regression mampu menghasilkan performa yang stabil dengan risiko overfitting yang lebih rendah dibandingkan model yang lebih kompleks (Prasetyo, 2024).

Di sisi lain, Support Vector Machine (SVM) memiliki keunggulan dalam menangani data dengan dimensi tinggi, seperti data tekstual yang telah melalui proses vectorization. SVM bekerja dengan cara mencari hyperplane paling optimal untuk memisahkan data antar kelas. SVM sering kali melampaui algoritma lain dalam hal akurasi dan F1-Score ketika dihadapkan pada dataset yang memiliki sebaran data tidak seimbang (imbalanced data), yang sering ditemukan pada ulasan aplikasi di mana sentimen negatif atau positif bisa sangat mendominasi salah satu sisi (Putra & Zuliarso, 2025).

Meskipun penelitian mengenai analisis sentimen telah banyak dilakukan pada berbagai aplikasi e-commerce atau layanan publik, penelitian yang secara spesifik membandingkan kinerja Logistic Regression dan SVM pada domain game e-Football masih sangat terbatas. Sebagian besar penelitian sebelumnya lebih banyak berfokus pada algoritma Naive Bayes atau Deep Learning yang membutuhkan sumber daya perangkat keras yang besar. Terdapat kebutuhan untuk mengevaluasi algoritma mana yang lebih efisien namun tetap akurat jika diterapkan pada karakteristik ulasan pemain e-Football di Indonesia yang unik.

1.2 Rumusan Masalah

Rumusan masalah dalam penelitian ini adalah bagaimana penerapan algoritma Logistic Regression dan Support Vector Machine (SVM) dalam mengklasifikasikan sentimen ulasan pengguna terhadap game mobile e-Football, serta bagaimana hasil klasifikasi sentimen yang dihasilkan dari masing-masing algoritma berdasarkan data ulasan yang diperoleh.

1.3 Batasan Masalah

1. Data yang digunakan adalah ulasan pengguna aplikasi e-Football dari Google Play Store pada periode tertentu.

2. Kategori sentimen yang dianalisis hanya terdiri dari dua kelas: positif dan negatif.
3. Algoritma yang digunakan hanya Logistic Regression dan Support Vector Machine (SVM).
4. Data ulasan diproses melalui tahap preprocessing seperti tokenisasi, penghapusan stopword, dan stemming.

Penelitian hanya menampilkan hasil klasifikasi masing-masing algoritma tanpa melakukan perbandingan kinerja.

1.4 Tujuan Penelitian

1. Untuk menerapkan metode Logistic Regression dalam mengklasifikasikan sentimen ulasan pengguna terhadap game mobile e-Football.
2. Untuk menerapkan metode Support Vector Machine (SVM) dalam mengklasifikasikan sentimen pengguna terhadap game yang sama.
3. Untuk mengetahui hasil klasifikasi sentimen positif dan negatif dari ulasan pengguna game e-Football berdasarkan masing-masing metode.
4. Untuk melihat sejauh mana proses text preprocessing dan teknik scraping data ulasan dapat diterapkan secara efektif dalam proses analisis sentimen berbasis machine learning.

1.5 Manfaat Penelitian

1. Penelitian ini memberikan penulis kesempatan untuk lebih memahami penerapan algoritma machine learning dalam studi kasus nyata, khususnya dalam analisis sentimen berbasis teks.
2. Penulis juga mendapatkan pengalaman langsung dalam proses pengambilan data dari internet (scraping), pengolahan data teks, hingga evaluasi hasil klasifikasi menggunakan metrik tertentu.
3. Melalui penelitian ini, penulis dapat melatih kemampuan berpikir kritis dan terstruktur dalam menyelesaikan permasalahan yang berkaitan dengan data dan pemodelan algoritma.

BAB II

LANDASAN TEORI

2.1 Data dan Informasi

Data didefinisikan sebagai sekumpulan fakta objektif, simbol, atau catatan mentah mengenai suatu peristiwa yang belum memiliki konteks dan makna bagi penerimanya. Dalam disiplin ilmu komputer, data merupakan entitas dasar yang dihasilkan dari berbagai aktivitas digital, seperti transaksi, pengamatan, maupun input ulasan dari pengguna. Sebagai entitas mentah, data tidak dapat digunakan secara langsung untuk mendukung proses penalaran sebelum melalui tahapan pengolahan. Menurut Irawaty (2024), data berfungsi sebagai representasi dari kenyataan yang membutuhkan pemrosesan sistematis agar dapat diubah menjadi entitas yang lebih berharga, di mana kualitas data mentah menjadi penentu utama dalam validitas hasil akhir yang diperoleh dari sebuah sistem informasi.

Informasi merupakan hasil dari transformasi data yang telah diklasifikasikan, diorganisasikan, dan dianalisis sehingga memiliki arti, konteks, dan kegunaan bagi penggunanya. Perubahan data menjadi informasi bertujuan untuk mengurangi ketidakpastian serta memberikan wawasan yang dapat digunakan sebagai dasar pengambilan keputusan. Informasi yang efektif harus memiliki karakteristik akurasi yang tinggi, relevansi dengan kebutuhan pengguna, serta dapat diakses pada waktu yang tepat. Proses pengolahan ini dapat dilakukan melalui metode statistik maupun algoritma pembelajaran mesin untuk menemukan pola tersembunyi yang tidak terlihat pada data mentah secara kasat mata.

Dalam konteks penelitian ini, data primer yang diolah adalah kumpulan ulasan (*reviews*) pengguna permainan *mobile* e-Football yang diperoleh secara digital dari Google Play Store. Ulasan tersebut dikategorikan sebagai data tekstual tidak terstruktur (*unstructured data*) yang memiliki tingkat kompleksitas tinggi karena keragaman gaya bahasa manusia, penggunaan slang, dan singkatan. Tanpa adanya pemrosesan sistematis, ribuan ulasan tersebut hanya akan menjadi tumpukan teks yang tidak memberikan gambaran jelas mengenai kondisi permainan maupun tingkat kepuasan pemain.

Melalui penerapan analisis sentimen, data ulasan yang tidak teratur tersebut dikonversi menjadi informasi strategis yang menggambarkan sentimen kolektif pengguna secara objektif. Informasi ini mencakup pemetaan persepsi terhadap mekanisme permainan, stabilitas server, hingga respons terhadap pembaruan fitur. Hal ini selaras dengan penelitian Hanafi (2024) yang menyatakan bahwa pengolahan data mentah dari ulasan aplikasi menjadi informasi sentimen sangat krusial bagi pengembang untuk memahami pengalaman pengguna secara mendalam dan melakukan perbaikan layanan berdasarkan fakta yang akurat. Dengan demikian, proses pengolahan data menjadi informasi melalui analisis sentiment merupakan fondasi yang mendasari penggunaan algoritma klasifikasi dalam penelitian ini.

2.2 Text Mining

Text mining merupakan salah satu cabang dari *data mining* yang berfokus pada ekstraksi pola atau informasi berkualitas tinggi dari kumpulan data berbentuk teks yang tidak terstruktur. Menurut Haris dkk. (2024), *text mining* adalah proses penemuan pola dan informasi penting dari data teks dalam jumlah besar, seperti ulasan pengguna pada platform digital, dengan bantuan teknik *Natural Language Processing* (NLP) untuk memahami konteks bahasa manusia.

Tahapan utama dalam implementasi *text mining* meliputi:

- Pengumpulan Data: Tahap akuisisi data mentah dari sumber digital. Dalam riset ini, ulasan diambil dari Google Play Store yang sering kali mencerminkan pengalaman dan persepsi pengguna terhadap kualitas permainan.
- Praproses Data (*Preprocessing*): Langkah pembersihan teks dari elemen yang tidak relevan. Tahapan ini meliputi *case folding* (penyeragaman huruf kecil), *tokenizing* (pemecahan kalimat), *stopword removal* (penghapusan kata umum), dan *stemming* (mengubah kata ke bentuk dasar).
- Ekstraksi Fitur: Transformasi kata-kata ke dalam representasi numerik agar dapat diolah oleh mesin, di mana metode yang umum digunakan

- adalah *Term Frequency-Inverse Document Frequency* (TF-IDF) atau representasi *embedding*.
- Klasifikasi: Proses pengelompokan teks ke dalam kategori sentimen (positif, negatif, atau netral) menggunakan algoritma pembelajaran mesin.

Dalam penelitian ini, *text mining* digunakan sebagai instrumen utama untuk mengolah ulasan pengguna permainan e-Football. Hal ini diperkuat oleh penelitian Hanafi dan R (2024) yang menyatakan bahwa pengolahan data ulasan pada platform digital menjadi informasi sentimen sangat krusial bagi pengembang untuk memahami pengalaman pengguna secara mendalam dan melakukan perbaikan layanan berdasarkan data yang akurat.

Berdasarkan konsep text mining tersebut, salah satu penerapannya yang banyak digunakan pada data ulasan digital adalah analisis sentimen. Analisis sentimen memanfaatkan teknik text mining untuk mengidentifikasi dan mengelompokkan opini pengguna berdasarkan kecenderungan sikap yang terkandung dalam teks.

2.3 Analisis Sentimen

Analisis Sentimen adalah salah satu cabang *text mining* yang berfokus pada identifikasi, ekstraksi, dan klasifikasi opini dalam teks. Tujuan utama dari sentimen analisis adalah mengetahui sikap atau perasaan seseorang terhadap objek tertentu. Menurut Ramadhan dkk. (2024), sentimen analisis memungkinkan sistem komputer untuk memahami apakah sebuah teks mengandung opini yang bersifat positif, negatif, atau netral secara otomatis menggunakan pemrosesan bahasa alami.

Dalam ranah permainan *mobile*, ulasan pengguna sering kali berisi ekspresi emosional terkait pengalaman bermain, misalnya “*game* ini seru sekali” (sentimen positif), “sering *lag* dan *server* buruk” (sentimen negatif), atau “fitur lumayan, tapi masih perlu ditingkatkan” (sentimen netral). Dengan sentimen analisis, opini tersebut dapat dikategorikan dan dihitung proporsinya secara sistematis.

Penerapan sentimen analisis pada e-Football dapat memberikan manfaat bagi pengembang, antara lain:

- Mengetahui tingkat kepuasan pemain terhadap kualitas grafis, *gameplay*, dan kestabilan *server*.
- Mengidentifikasi masalah teknis yang sering dikeluhkan secara massal oleh pengguna.
- Memahami ekspektasi pengguna terhadap fitur-fitur baru yang dikembangkan.

Dengan demikian, sentimen analisis tidak hanya membantu memahami persepsi pemain, tetapi juga berfungsi sebagai alat evaluasi kualitas layanan. Hal ini diperkuat oleh penelitian Wati dkk. (2022) yang menyatakan bahwa pengolahan ulasan pengguna menjadi informasi sentimen melalui teknik klasifikasi sangat efektif untuk memantau performa aplikasi dan kepuasan pengguna secara *real-time*. Untuk dapat menghasilkan klasifikasi sentimen yang akurat, data teks yang digunakan perlu dipersiapkan secara sistematis. Oleh karena itu, sebelum dilakukan proses ekstraksi fitur dan klasifikasi, diperlukan tahapan *preprocessing* teks yang bertujuan untuk membersihkan dan menstandarkan data ulasan.

2.4 Preprocessing Teks

Preprocessing teks adalah tahapan awal dalam text mining dan analisis sentimen yang bertujuan untuk membersihkan serta menyiapkan data teks yang bersifat tidak terstruktur agar dapat diproses secara optimal oleh algoritma pembelajaran mesin. Data teks mentah, khususnya ulasan pengguna di platform digital, sering mengandung variasi penulisan, kata tidak baku, serta karakter yang tidak relevan sehingga dapat mengurangi kualitas analisis jika tidak dibersihkan. Proses ini melibatkan serangkaian teknik untuk menstandarkan teks dan menghilangkan noise, dengan tujuan meningkatkan kualitas representasi data serta mendukung efektivitas model klasifikasi dalam mengenali pola sentimen secara akurat. Putusan teks yang telah melalui tahap *preprocessing* umumnya memiliki

kontribusi lebih besar dalam meningkatkan performa dan stabilitas model analisis sentimen dibandingkan teks yang belum diproses (Khomsah & Aribowo, 2020).

Berikut adalah tahapan-tahapan preprocessing teks:

1. Cleaning

Cleaning merupakan proses pembersihan teks dari karakter atau elemen yang tidak memiliki kontribusi terhadap makna sentimen. Tahapan ini meliputi penghapusan tanda baca, angka, simbol khusus, URL, emotikon, serta karakter non-alfabet lainnya. Tujuan dari cleaning adalah memastikan bahwa teks yang diproses hanya mengandung kata-kata yang relevan sehingga tidak mengganggu proses analisis selanjutnya.

2. Case Folding

Case folding adalah proses penyeragaman huruf dalam teks, biasanya dengan mengubah seluruh karakter menjadi huruf kecil. Tahapan ini bertujuan untuk menghindari perbedaan makna yang disebabkan oleh variasi penggunaan huruf besar dan huruf kecil. Dengan case folding, kata yang sama namun ditulis dalam bentuk berbeda akan diperlakukan sebagai satu entitas yang sama oleh sistem.

3. Normalisasi

Normalisasi merupakan proses penyelarasan kata-kata tidak baku atau singkatan ke dalam bentuk kata baku yang sesuai dengan kaidah bahasa. Dalam ulasan pengguna, sering ditemukan penggunaan kata tidak formal, slang, atau singkatan yang dapat menyulitkan proses analisis. Melalui normalisasi, variasi kata tersebut diubah ke bentuk yang seragam sehingga makna kata dapat dikenali dengan lebih baik oleh sistem.

4. Tokenizing

Tokenizing adalah proses pemecahan teks menjadi unit-unit yang lebih kecil yang disebut token, biasanya berupa kata atau frasa. Tahapan ini memungkinkan sistem untuk menganalisis setiap kata secara terpisah. Tokenizing menjadi dasar bagi proses selanjutnya, seperti pembobotan kata dan klasifikasi sentimen, karena algoritma pembelajaran mesin bekerja pada representasi token tersebut.

5. Stopword Removal

Stopword removal merupakan proses penghapusan kata-kata umum yang sering muncul dalam teks tetapi tidak memiliki kontribusi signifikan terhadap makna sentimen, seperti kata penghubung dan kata depan. Penghapusan stopwords bertujuan untuk mengurangi jumlah fitur yang tidak relevan sehingga fokus analisis dapat diarahkan pada kata-kata yang lebih bermakna.

5. Stemming

Stemming adalah proses mengubah kata ke bentuk dasarnya dengan menghilangkan imbuhan atau afiks yang melekat pada kata tersebut. Tahapan ini bertujuan untuk menyatukan berbagai variasi kata yang memiliki akar kata yang sama agar dianggap sebagai satu fitur. Dengan stemming, dimensi data dapat dikurangi dan proses klasifikasi sentimen menjadi lebih efisien.

Setelah data teks melalui tahap preprocessing dan berada dalam bentuk yang lebih terstruktur, langkah selanjutnya adalah merepresentasikan teks tersebut ke dalam bentuk numerik. Representasi ini dilakukan melalui proses pembobotan kata, salah satunya menggunakan metode Term Frequency–Inverse Document Frequency (TF-IDF).

2.5 Logistic Regression

Logistic Regression merupakan algoritma statistik dan pembelajaran mesin yang digunakan untuk memodelkan probabilitas variabel dependen kategori biner. Meskipun namanya mengandung kata "regresi", metode ini sebenarnya digunakan untuk tugas klasifikasi. Prinsip kerja algoritma ini adalah menggunakan fungsi

logistik atau fungsi sigmoid untuk memetakan input data tekstual (yang telah diubah menjadi vektor numerik) ke dalam rentang nilai antara 0 dan 1. Jika probabilitas yang dihasilkan lebih besar dari ambang batas tertentu (biasanya 0,5), maka data akan diklasifikasikan ke dalam satu kelas (misalnya positif), dan sebaliknya. Keunggulan Logistic Regression terletak pada kesederhanaannya yang memungkinkan proses pelatihan model berjalan sangat cepat, memiliki interpretasi yang mudah dipahami, serta cenderung stabil saat menangani dataset yang memiliki hubungan linier antar fiturnya (Prasetyo, 2024).

Secara matematis, proses klasifikasi dimulai dengan menghitung kombinasi linier dari fitur input menggunakan persamaan:

$$z = \beta_0 + \beta_1x_1 + \beta_2x_2 + \dots + \beta_nx_n$$

Nilai z kemudian dimasukkan ke dalam fungsi Sigmoid untuk menghasilkan probabilitas P :

$$P = \sigma(z) = \frac{1}{1 + e^{-z}}$$

Keterangan:

- P : Probabilitas keluaran (antara 0 dan 1).
- e : Bilangan Euler atau konstanta eksponensial ($\approx 2,718$).
- z : Hasil dari fungsi linier (logit).
- x : Fitur input (misalnya nilai pembobotan kata TF-IDF).
- β : Koefisien atau bobot yang dipelajari oleh model.

Penerapan *Logistic Regression* dalam analisis sentimen memiliki beberapa keuntungan teknis, antara lain:

1. Efisiensi Komputasi: Proses pelatihan model relatif sangat cepat dan tidak membutuhkan sumber daya perangkat keras (GPU/RAM) yang besar dibandingkan algoritma *Deep Learning*.

2. Output Probabilistik: Tidak hanya memberikan label kelas, algoritma ini juga memberikan nilai kepercayaan (probabilitas) seberapa kuat sebuah ulasan dikategorikan sebagai sentimen tertentu.
3. Rendah Risiko Overfitting: Pada dataset yang memiliki dimensi fitur yang jelas dan terstruktur, model ini cenderung memberikan performa yang stabil dan generalisasi yang baik pada data baru

Meskipun efisien, terdapat beberapa batasan dalam penggunaan algoritma ini:

1. Asumsi Linieritas: Algoritma ini mengasumsikan adanya hubungan linier antara variabel independen dan log-odds dari variabel dependen. Jika data memiliki pola non-linier yang kompleks (seperti pada bahasa sarkasme yang mendalam), akurasi model mungkin menurun.
2. Ketergantungan pada Kualitas Fitur: *Logistic Regression* sangat sensitif terhadap *noise* dan membutuhkan *preprocessing* yang sangat bersih agar fitur-fitur yang dihasilkan benar-benar merepresentasikan sentimen.
3. Keterbatasan pada Hubungan Kompleks: Tidak mampu menangani masalah yang membutuhkan pemahaman konteks urutan kata yang sangat kompleks secara mendalam sebagaimana yang bisa dilakukan oleh model berbasis kernel atau saraf tiruan (Prasetyo, 2024).

Selain Logistic Regression, terdapat algoritma klasifikasi lain yang juga banyak digunakan dalam analisis sentimen, yaitu Support Vector Machine (SVM). Algoritma ini digunakan sebagai pembanding untuk mengetahui performa klasifikasi sentimen pada data ulasan yang sama.

2.6 Support Vector Machine (SVM)

Support Vector Machine (SVM) adalah algoritma klasifikasi yang bekerja dengan mencari *hyperplane* terbaik untuk memisahkan data ke dalam kelas tertentu. Dalam analisis sentimen, SVM berfungsi untuk memisahkan ulasan positif dan negatif berdasarkan vektor fitur teks. Menurut Kulsum dkk. (2022), SVM merupakan algoritma yang sangat efektif dalam melakukan klasifikasi

ulasan aplikasi karena kemampuannya menangani data tekstual dengan margin pemisahan kelas yang optimal.

Secara matematis, fungsi SVM dapat dinyatakan :

$$f(x) = \text{sign} \left(\sum_{i=1}^n a_i y_i K(x_i, x) + b \right)$$

Keterangan:

- x_i : data latih
- y_i : label kelas (positif = +1, negatif = -1)
- α_i : bobot hasil perhitungan optimasi
- $K(x_i, x)$: fungsi kernel untuk memetakan data ke dimensi lebih tinggi
- b : bias
- $f(x)$: hasil klasifikasi (positif atau negatif)

Keunggulan SVM adalah kemampuannya menghasilkan klasifikasi yang lebih akurat pada data berdimensi tinggi, seperti teks dengan ribuan fitur kata. Hal ini membuat SVM sangat relevan untuk analisis sentimen pada ulasan permainan *mobile* e-Football, yang biasanya terdiri dari data teks panjang dengan variasi kata yang luas. Implementasi ini sejalan dengan penelitian Sihotang dkk. (2023) yang membuktikan bahwa penggunaan metode SVM sangat membantu dalam proses klasifikasi opini untuk menentukan label sentimen yang akurat pada data ulasan digital.

Secara keseluruhan, SVM merupakan algoritma yang andal dalam analisis sentimen berbasis teks, terutama ketika dihadapkan pada data dengan kompleksitas fitur yang tinggi. Oleh karena itu, penggunaan SVM dalam penelitian ini diharapkan dapat memberikan pembandingan kinerja yang objektif terhadap algoritma Logistic Regression dalam mengklasifikasikan sentimen ulasan pengguna.

Penerapan Support Vector Machine (SVM) dalam analisis sentimen memiliki beberapa keuntungan teknis, antara lain:

1. Kemampuan Menangani Data Berdimensi Tinggi:

SVM sangat efektif digunakan pada data teks yang direpresentasikan dalam bentuk vektor berdimensi tinggi, seperti TF-IDF. Algoritma ini tetap mampu menghasilkan performa klasifikasi yang baik meskipun jumlah fitur kata sangat besar.

2. Margin Pemisahan Kelas yang Optimal:

SVM bekerja dengan mencari hyperplane yang memiliki margin pemisah maksimum antar kelas, sehingga menghasilkan model dengan kemampuan generalisasi yang baik terhadap data baru dan relatif tahan terhadap overfitting.

3. Fleksibilitas Melalui Fungsi Kernel:

Dengan penggunaan fungsi kernel, SVM mampu menangani pola data yang tidak dapat dipisahkan secara linier. Hal ini sangat relevan dalam analisis sentimen, di mana hubungan antar kata dan makna sering bersifat kompleks dan non-linier.

Meskipun memiliki keunggulan, terdapat beberapa batasan dalam penggunaan algoritma SVM, antara lain:

1. Kebutuhan Komputasi yang Lebih Tinggi:

Proses pelatihan SVM umumnya membutuhkan waktu komputasi yang lebih lama dan sumber daya perangkat keras yang lebih besar dibandingkan algoritma klasifikasi linear seperti Logistic Regression, terutama pada dataset berukuran besar.

2. Sensitivitas terhadap Parameter:

Performa SVM sangat dipengaruhi oleh pemilihan parameter, seperti nilai regularisasi dan jenis kernel yang digunakan. Penentuan parameter yang kurang tepat dapat menyebabkan penurunan akurasi model.

3. Tingkat Interpretabilitas yang Rendah:

Berbeda dengan Logistic Regression, hasil klasifikasi SVM relatif sulit

untuk diinterpretasikan karena tidak secara langsung menunjukkan kontribusi setiap fitur terhadap keputusan model, sehingga kurang sesuai untuk penelitian yang membutuhkan penjelasan model secara mendalam.

2.7 Confusion Matrix

Confusion Matrix merupakan metode evaluasi performa klasifikasi yang banyak digunakan dalam penelitian analisis sentimen karena mampu memberikan gambaran detail mengenai distribusi hasil prediksi model terhadap data aktual. Melalui matriks ini, peneliti dapat mengidentifikasi jumlah prediksi benar maupun salah pada setiap kelas, sehingga dapat dihitung metrik evaluasi seperti akurasi, presisi, recall, dan F1-score. Hal ini menjadikan confusion matrix lebih informatif dibandingkan sekadar mengandalkan akurasi, terutama ketika data bersifat tidak seimbang. Seperti dikemukakan oleh (Rinardi 2024), penggunaan confusion matrix memungkinkan peneliti memahami pola kesalahan klasifikasi secara lebih spesifik, yang penting dalam meningkatkan kualitas model.

Sejalan dengan itu, (Yuspriyadi 2023) juga menegaskan bahwa confusion matrix sangat efektif untuk membandingkan performa berbagai algoritma klasifikasi, termasuk Naive Bayes dan Support Vector Machine, karena metrik yang dihasilkan mampu menunjukkan kelemahan dan keunggulan setiap algoritma dalam konteks data yang kompleks. Selanjutnya, (Susanto 2022) menambahkan bahwa evaluasi berbasis confusion matrix dapat menjadi dasar dalam menentukan model terbaik untuk diterapkan pada studi analisis sentimen berbasis teks, termasuk dalam domain permainan digital seperti e-Football.

2.8 Unified Modeling Language (UML)

Unified Modeling Language (UML) merupakan bahasa pemodelan standar yang digunakan untuk memvisualisasikan, merancang, dan mendokumentasikan sistem perangkat lunak secara grafis. UML berfungsi sebagai alat bantu komunikasi antara analis sistem dan pengembang agar rancangan sistem dapat dipahami secara konsisten sebelum tahap implementasi dilakukan. Menurut Hapsari (2020), UML menyediakan notasi visual yang terstandarisasi untuk

menggambarkan struktur, perilaku, serta interaksi antar komponen dalam suatu sistem informasi.

Dalam konteks penelitian analisis sentimen, UML digunakan sebagai pendekatan konseptual untuk memodelkan alur kerja sistem secara umum, mulai dari interaksi pengguna dengan sistem hingga proses pengolahan data teks. Penggunaan UML membantu menyederhanakan sistem yang kompleks ke dalam bentuk diagram visual sehingga memudahkan pemahaman terhadap logika sistem secara menyeluruh. Hal ini sejalan dengan pendapat Taufiq dkk. (2021) yang menyatakan bahwa UML berperan penting dalam mengorganisasi logika sistem, khususnya pada sistem yang melibatkan pengolahan data teks dan pemrosesan bahasa alami.

Selain itu, UML juga berfungsi sebagai dokumentasi sistem yang bersifat jangka panjang. Dengan adanya pemodelan yang jelas, proses pengembangan, pemeliharaan, dan pengembangan lanjutan sistem dapat dilakukan secara lebih terstruktur. Oleh karena itu, penerapan UML dalam penelitian ini digunakan sebagai dasar konseptual dalam merancang sistem analisis sentimen agar selaras antara kebutuhan sistem dan proses implementasinya, sebagaimana ditegaskan oleh Taufiq dkk. (2021).

2.8.1 Jenis Diagram UML yang Digunakan

1. Use Case Diagram

Use case diagram digunakan untuk menggambarkan hubungan interaksi antara aktor dan sistem secara umum. Diagram ini menunjukkan fungsi-fungsi utama yang dapat dilakukan oleh pengguna terhadap sistem analisis sentimen sehingga batasan dan ruang lingkup sistem dapat didefinisikan dengan jelas (Hapsari, 2020).

2. Activity Diagram

Activity diagram digunakan untuk menggambarkan alur aktivitas sistem secara konseptual dari awal hingga akhir. Diagram ini memberikan gambaran umum

mengenai tahapan proses dalam sistem analisis sentimen, mulai dari penerimaan data hingga dihasilkannya informasi sentimen (Taufiq dkk., 2021).

3. Class Diagram

Class diagram digunakan untuk memodelkan struktur sistem secara konseptual dalam bentuk kelas-kelas utama beserta relasinya. Diagram ini membantu dalam memahami pembagian komponen sistem secara logis tanpa masuk ke detail implementasi teknis (Hapsari, 2020).

4. Sequence Diagram

Sequence diagram digunakan untuk menggambarkan urutan interaksi antar objek dalam sistem berdasarkan alur waktu. Diagram ini memberikan gambaran bagaimana komunikasi antar komponen sistem berlangsung secara berurutan dalam proses analisis sentimen (Taufiq dkk., 2021).

2.9 Google Colab

Google Colaboratory atau Google Colab adalah platform *notebook* berbasis *cloud* yang dikembangkan oleh Google untuk memungkinkan penulisan dan pengeksekusian kode Python secara interaktif melalui peramban web. Platform ini menjadi solusi bagi peneliti yang membutuhkan lingkungan komputasi tinggi tanpa harus melakukan instalasi perangkat lunak atau konfigurasi perangkat keras secara manual di komputer lokal. Menurut Kurniawan dkk. (2024), Google Colab sangat efektif digunakan dalam penelitian yang melibatkan pengolahan data besar karena menyediakan akses gratis ke sumber daya komputasi seperti GPU (*Graphics Processing Unit*) dan TPU (*Tensor Processing Unit*) yang dapat mempercepat proses pelatihan model pembelajaran mesin.

Selain keunggulan pada aspek perangkat keras, Google Colab juga mendukung kolaborasi riset secara *real-time*. Seluruh berkas *notebook* disimpan secara otomatis di Google Drive, sehingga memudahkan proses berbagi kode dan dokumentasi antar peneliti. Sebagaimana dijelaskan oleh Andriyanto dkk. (2024),

integrasi ini memastikan bahwa setiap tahapan eksperimen, mulai dari pembersihan data hingga evaluasi model, dapat dilakukan secara transparan dan mudah diproduksi kembali (*reproducible*). Hal ini sangat mendukung efisiensi waktu dalam penelitian analisis sentimen yang memerlukan banyak iterasi pengujian.

Kemudahan akses terhadap pustaka-pustaka populer seperti Pandas, Scikit-Learn, dan NLTK menjadikan Google Colab sebagai lingkungan pengembangan utama dalam bidang sains data. Platform ini memungkinkan peneliti untuk fokus pada logika algoritma tanpa terhambat oleh keterbatasan spesifikasi teknis perangkat keras lokal. Penggunaan Google Colab dalam skripsi ini bertujuan untuk memproses data ulasan *e-Football* secara daring dengan memanfaatkan stabilitas server Google untuk menjamin kelancaran eksekusi algoritma klasifikasi.

2.10 Penelitian Terdahulu

Berikut ringkasan beberapa penelitian terdahulu yang relevan dengan topik analisis sentimen menggunakan algoritma Logistic Regression dan SVM dalam lima tahun terakhir:

Tabel 2. 1 Penelitian Terdahulu

No	Peneliti & Tahun	Judul Penelitian	Sumber Data	Algoritma	Hasil Utama
1	Iskandar dkk. (2024)	Analisis Sentimen Ulasan Aplikasi E-Wallet menggunakan SVM dan Logistic Regression	Google Play Store	SVM & LR	SVM unggul dengan akurasi 88%, sementara LR mencapai 86%.
2	Sari & Rosiani (2024)	Komparasi Klasifikasi Sentimen Review Pengguna Aplikasi Mobile	Google Play Store	SVM & LR	SVM mencapai akurasi 87%. LR unggul dalam kecepatan proses

No	Peneliti & Tahun	Judul Penelitian	Sumber Data	Algoritma	Hasil Utama
		dengan SVM dan LR			data.
3	Fahmi dkk. (2022)	Sentiment Analysis of Digital Payment Reviews using Machine Learning	Google Play Store	SVM, LR, & NB	SVM memberikan performa konsisten dengan akurasi di atas 85%.
4	Sibarani dkk. (2023)	Analisis Sentimen Masyarakat Terhadap Kenaikan Harga BBM Menggunakan SVM dan LR	Twitter (X)	SVM & LR	Logistic Regression mengungguli SVM pada dataset teks pendek dengan akurasi 84%.
5	Putra dkk. (2023)	Perbandingan Algoritma SVM dan Logistic Regression untuk Analisis Sentimen Marketplace	Google Play Store	SVM & LR	LR memperoleh akurasi 84.5%, sedikit di atas SVM yang memperoleh 83.2%.

BAB III

METODOLOGI PENELITIAN

3.1 Jenis dan Pendekatan Penelitian

Penelitian ini menggunakan pendekatan kuantitatif dengan metode eksperimen. Pendekatan kuantitatif dipilih karena penelitian berfokus pada pengolahan data ulasan pengguna yang telah diubah ke dalam bentuk numerik untuk dianalisis secara komputasional. Data numerik tersebut dihasilkan melalui proses pengolahan teks dan selanjutnya digunakan untuk mengukur kinerja algoritma klasifikasi sentimen secara objektif.

Metode eksperimen diterapkan dengan cara menerapkan algoritma Logistic Regression dan Support Vector Machine (SVM) pada data ulasan permainan mobile *e-Football* yang telah melalui tahap preprocessing. Melalui metode ini, penelitian bertujuan untuk melihat bagaimana model klasifikasi bekerja dalam mengelompokkan sentimen ulasan pengguna berdasarkan parameter evaluasi yang digunakan. Pendekatan ini memungkinkan hasil penelitian dapat diukur, dianalisis, serta diuji kembali secara sistematis berdasarkan metrik evaluasi yang telah ditetapkan.

3.2 Alur Penelitian

Alur penelitian ini menjelaskan tahapan-tahapan yang dilakukan dalam proses analisis sentimen terhadap ulasan pengguna aplikasi permainan mobile *e-Football* yang diperoleh dari Google Play Store. Data yang digunakan dalam penelitian ini berupa teks ulasan pengguna dalam Bahasa Indonesia yang mencerminkan pengalaman, penilaian, serta tanggapan pengguna terhadap permainan.

Penelitian diawali dengan pengumpulan data ulasan pengguna aplikasi *e-Football* yang diambil dari Google Play Store menggunakan teknik *web scraping*.

Data yang diperoleh berupa kumpulan teks ulasan mentah yang selanjutnya digunakan sebagai dataset penelitian. Setelah data terkumpul, dilakukan preprocessing awal untuk membersihkan data dari unsur-unsur yang tidak relevan. Tahapan preprocessing awal meliputi *cleaning* untuk menghapus karakter yang tidak diperlukan, *case folding* untuk mengubah seluruh huruf menjadi huruf kecil, serta normalisasi kata guna menyelaraskan kata tidak baku ke bentuk baku.

Dataset yang telah melalui preprocessing awal kemudian dilanjutkan ke tahap preprocessing lanjutan untuk kebutuhan klasifikasi berbasis pembelajaran mesin. Pada tahap ini dilakukan *tokenizing* untuk memecah teks menjadi kata, *stopword removal* untuk menghilangkan kata-kata umum yang tidak memiliki pengaruh terhadap sentimen, serta *stemming* untuk mengubah kata berimbuhan menjadi bentuk kata dasar. Tahapan ini bertujuan untuk menghasilkan data teks yang lebih terstruktur dan siap digunakan dalam proses ekstraksi fitur.

Setelah data teks siap, dilakukan ekstraksi fitur menggunakan metode TF-IDF (*Term Frequency–Inverse Document Frequency*) untuk mengubah data teks menjadi representasi numerik. Representasi numerik ini selanjutnya digunakan sebagai input pada proses klasifikasi. Dataset kemudian dibagi menjadi data latih dan data uji guna mendukung proses pelatihan dan pengujian model.

Proses klasifikasi sentimen dilakukan dengan menerapkan algoritma Logistic Regression dan Support Vector Machine (SVM). Penggunaan kedua algoritma ini bertujuan untuk memanfaatkan kemampuan model klasifikasi linear dalam mengenali pola sentimen positif dan negatif pada data ulasan pengguna. A'yun dkk. (2024) menekankan bahwa eksperimen sistematis pada algoritma SVM penting dilakukan untuk memahami kemampuan model dalam menangani fitur teks berdimensi tinggi, sehingga menghasilkan performa klasifikasi yang optimal.

Hasil klasifikasi dari model Logistic Regression dan SVM selanjutnya dievaluasi menggunakan metrik akurasi, presisi, recall, dan F1-score. Evaluasi ini dilakukan untuk memperoleh gambaran kinerja masing-masing algoritma dalam mengklasifikasikan sentimen ulasan pengguna. Al Amin dkk. (2024) menyatakan bahwa Logistic Regression memiliki keunggulan dalam kemudahan interpretasi model serta efisiensi waktu komputasi, sehingga sesuai digunakan dalam klasifikasi sentimen berbasis teks. Seluruh hasil evaluasi tersebut digunakan sebagai dasar dalam menganalisis karakteristik sentimen pengguna terhadap permainan mobile *e-Football*.

3.3 Lokasi dan Objek Penelitian

Lokasi penelitian dilakukan secara daring (online) dengan memanfaatkan sumber data dari platform Google Play Store, khususnya pada kolom ulasan pengguna aplikasi mobile e-football. Objek penelitian ini adalah ulasan teks dari pengguna aplikasi yang dipublikasikan secara terbuka, karena ulasan tersebut merepresentasikan opini dan sentimen pengguna secara langsung terhadap pengalaman bermain. Haq, Rosyadi, Nasir, dan Khambali (2024) menjelaskan bahwa ulasan pengguna pada Google Play Store mencerminkan tingkat kepuasan dan ketidakpuasan pengguna secara autentik, sehingga dapat dimanfaatkan sebagai sumber data yang valid dan relevan dalam penelitian analisis sentimen aplikasi mobile.

3.4 Sumber Data

Data yang digunakan dalam penelitian ini diperoleh dari ulasan pengguna aplikasi permainan mobile e-Football yang terdapat pada Google Play Store. Ulasan tersebut dipilih karena berisi pendapat dan pengalaman pengguna setelah menggunakan aplikasi, sehingga dapat menggambarkan respon pengguna terhadap permainan secara langsung. Data yang diambil berupa teks ulasan tanpa melibatkan informasi identitas pengguna. Seluruh ulasan yang terkumpul digunakan sebagai bahan utama dalam penelitian dan diolah pada tahap selanjutnya, mulai dari pembersihan teks hingga proses klasifikasi sentimen.

Setiap ulasan diperlakukan sebagai satu kesatuan data yang dianalisis untuk mengetahui kecenderungan sentimen pengguna terhadap permainan e-Football. Dengan memanfaatkan ulasan pengguna sebagai sumber data, penelitian ini berfokus pada analisis sentimen berdasarkan pengalaman nyata yang disampaikan secara bebas oleh pengguna.

3.5 Teknik Pengumpulan Data

Tahapan pengumpulan data merupakan langkah awal yang fundamental dalam penelitian ini untuk mendapatkan basis data ulasan pengguna aplikasi *e-Football* secara objektif. Mengingat volume ulasan pada Google Play Store sangat besar dan bersifat dinamis, maka proses pengambilan data dilakukan dengan menerapkan metode *Web Scraping* secara otomatis. Teknik ini memungkinkan peneliti untuk mengekstraksi informasi secara terstruktur dari halaman web yang pada dasarnya tidak terstruktur menjadi format data yang dapat diolah lebih lanjut.

Dalam implementasinya, proses ini memanfaatkan *library* khusus berbasis bahasa pemrograman Python, yaitu *google-play-scraper*. Alat ini digunakan untuk melakukan *crawling* pada server Google Play Store guna mengambil metadata ulasan yang mencakup beberapa parameter penting, antara lain:

- Identitas Pengguna: Nama pengguna (*userName*) untuk memastikan validitas ulasan tunggal.
- Skor Penilaian (*Score*): Nilai bintang (1-5) yang diberikan pengguna sebagai indikator awal sentimen.
- Konten Ulasan (*Content*): Teks ulasan mentah yang merupakan objek utama penelitian.
- Waktu Ulasan (*At*): Tanggal pemberian ulasan untuk memastikan relevansi data dengan versi aplikasi saat ini.

3.6 Preprocessing Data

Preprocessing data merupakan tahap penting dalam penelitian ini yang bertujuan untuk menyiapkan data ulasan pengguna agar layak digunakan pada

proses analisis sentimen. Data ulasan yang diperoleh dari Google Play Store masih berupa teks mentah dan mengandung berbagai elemen yang tidak relevan, sehingga perlu dilakukan pembersihan dan penyeragaman. Melalui tahap preprocessing, data teks diharapkan menjadi lebih terstruktur dan mampu merepresentasikan sentimen pengguna secara lebih akurat.

3.6.1 Cleaning

cleaning data adalah tahap krusial dalam analisis data di mana data mentah (raw data) disiapkan untuk diproses lebih lanjut. Tujuannya adalah untuk mendeteksi, memperbaiki, atau menghapus kesalahan, duplikasi, dan inkonsistensi yang terdapat dalam data. Proses ini memastikan bahwa data yang digunakan untuk analisis akurat, konsisten, dan dapat diandalkan. Tanpa *cleaning* data yang tepat, hasil analisis bisa menjadi tidak valid dan menyesatkan.

Tabel 3. 1 Cleaning

Sebelum	Sesudah
walaupun grafik sudah standar tapi masih banyak lag. berasa berat banget	walaupun grafik sudah standar tapi masih banyak lag berasa berat banget

Proses data *cleaning* ini bertujuan untuk menyempurnakan dan menstandarisasi teks ulasan (kolom "content") menjadi versi yang lebih rapi dan siap dianalisis (kolom "cleaning"). Tahapan yang dilakukan mencakup beberapa aspek krusial dalam pemrosesan bahasa alami (natural language processing).

3.6.2 Case Folding

Case Folding merupakan proses untuk penyeragaman bentuk huruf atau kata ke dalam bentuk huruf kecil. Selain itu case folding juga bisa untuk menghilangkan tanda baca dan menghapus karakter spasi yang berlebihan. Pada proses ini berguna agar huruf besar dan huruf kecil tidak terdeteksi memiliki perbedaan arti. Adapun sebagai contoh kalimat ulasan

Tabel 3. 2 Case Folding

Sebelum	Sesudah
Tolong PES Efootballagak di gampangin lah dapat kartu epicnya biar player lebih semangat lagijagan pelit ya	tolong pes efootballagak di gampangin lah dapat kartu epicnya biar player lebih semangat lagijagan pelit ya

3.6.3 Normalisasi Kata

Pada bagian spelling normalization berguna untuk melakukan perbaikan kata-kata yang disingkat maupun salah ejaan dengan bentuk tertentu dengan maksud yang sama, sebagai contoh pada kata “tidak” memiliki banyak bentuk penulisan yaitu tak, enggak, tdak, tdk, dan sebagainya. Adapun contoh dalam kalimat ulasan.

Tabel 3. 3 Normalisasi Kata

Sebelum	Sesudah
walau pun grafik udah standar tapi masih banyak lag berasa berat banget	walau pun grafik udah standar tapi masih banyak lag berasa berat sekali

3.6.4 Tokenizing

Tokenizing atau tokenisasi merupakan proses memisahkan teks dalam dokumen menjadi potongan kata yang tidak saling berpengaruh atau independen yang disebut dengan token. Tokenizing digunakan untuk mendapatkan potongan kata atau token yang akan menjadi entitas yang memiliki nilai dalam penyusunan matriks dokumen pada proses selanjutnya. Pada proses ini memudahkan proses perhitungan keberadaan kata tersebut dalam dokumen maupun dalam menghitung

frekuensi kemunculan kata tersebut dalam corpus. Adapun sebagai contoh kalimat ulasan

Tabel 3. 4 Tokenizing

Sebelum	Sesudah
permainan nya udah bagus apa lagi kalo di tambah lisensi liga indonesia pasti makin bagus lagi	Permainan, nya, udah, bagus, apa, lagi, kalo, di, tambah, lisensi, liga, Indonesia, pasti, makin, bagus, lagi

3.6.5 Stopwords

Tahap filtering merupakan tahap dilakukannya penghilangan kata di dalam dokumen atau pengurangan dimensi kata di dalam corpus yang disebut stopwords. Stopwords digunakan untuk menghilangkan kata-kata yang tidak ada berpengaruh atau tidak mengurangi informasi didalam dokumen tersebut namun keberadaannya sering muncul. Contoh isi dalam bahasa indonesia berdasarkan data penelitian yaitu “semenjak”, “update”, “sering”, “lag”, “dan”, “banyak”, “bug” dan lain sebagainya

Tabel 3. 5 Stopwords

Sebelum	Sesudah
semenjak,update,sering,lag,dan,banyak,bug	Semenjak, update, lag, bug

3.6.6 Stemming

Stemming adalah proses dalam Natural Language Processing (NLP) untuk mengubah kata berimbuhan menjadi bentuk dasarnya (kata dasar atau stem). Imbuhan yg akan dihapus seperti awalan (me-, di-, ke-, se-), akhiran (-kan, -an, -i), atau sisipan

Tabel 3. 6 Stemming

Sebelum	Sesudah
efootball, plis, sih, pas, kalo, main, ama, orang, tuh, permainan, nya, jelek, mulu, pemain, nya, tuh, kaya, gabisa, atur, sesuai, reting, pemain, nya, tolong, cape, pemain, nya, lol, plis, perbaiki, cepat	efotball plis sih pas kalo main ama orang tuh main nya jelek mulu main nya tuh kaya gabisa atur sesuai reting main nya tolong cape main nya lol plis baik cepat

3.7 Ekstraksi Fitur

Tahapan ekstraksi fitur merupakan proses krusial dalam klasifikasi teks yang menjembatani data mentah dengan algoritma pembelajaran mesin. Mengingat model Support Vector Machine (SVM) dan Logistic Regression bekerja berdasarkan perhitungan matematis dan geometris, maka data tekstual yang telah melalui proses preprocessing (seperti cleaning, case folding, hingga stemming) tidak dapat diinput secara langsung. Oleh karena itu, penelitian ini menerapkan metode TF-IDF untuk mengonversi dokumen teks menjadi representasi numerik berbentuk matriks sparse.

Implementasi TF-IDF dalam penelitian ini tidak hanya sekadar menghitung kemunculan kata, tetapi juga mengevaluasi signifikansi sebuah kata terhadap ulasan game e-Football secara keseluruhan. Proses pembobotan ini melibatkan dua mekanisme kalkulasi utama:

1. Term Frequency (TF) Mekanisme ini memberikan penilaian terhadap intensitas kemunculan sebuah istilah (token) di dalam satu ulasan individu. Asumsi dasarnya adalah semakin sering suatu kata muncul (misalnya kata "lag" atau "grafik"), maka kata tersebut dianggap memiliki relevansi yang kuat terhadap makna ulasan tersebut.
2. Inverse Document Frequency (IDF) Mekanisme ini berfungsi sebagai penyeimbang dengan cara meninjau sebaran kata di seluruh dataset. IDF

3. akan memberikan bobot yang rendah (penalti) terhadap kata-kata yang terlalu generik dan muncul di hampir semua ulasan (seperti kata "ini", "ada", atau "game"), karena kata-kata tersebut tidak memiliki daya pembeda yang kuat. Sebaliknya, kata-kata yang bersifat spesifik dan hanya muncul pada kategori sentimen tertentu akan diberikan bobot yang lebih tinggi.

3.8 Pembagian Data (Data Splitting)

Setelah seluruh data melalui tahapan ekstraksi fitur menggunakan TF-IDF, langkah selanjutnya yang sangat fundamental dalam pembangunan model klasifikasi adalah pembagian dataset (*data splitting*). Pada penelitian ini, keseluruhan dataset ulasan *e-Football* didistribusikan ke dalam dua subset yang berbeda, yaitu Data Latih (*Training Data*) dan Data Uji (*Testing Data*). Pemisahan ini dilakukan secara acak menggunakan metode *random sampling* guna memastikan tidak adanya bias dalam pemilihan data.

Proporsi pembagian data yang diterapkan dalam penelitian ini adalah 80:20, di mana 80% dari total dataset dialokasikan sebagai data latih dan 20% sisanya digunakan sebagai data uji. Rasio ini dipilih berdasarkan standar umum dalam literatur pembelajaran mesin yang dianggap mampu memberikan keseimbangan optimal antara kemampuan model dalam mempelajari pola (*learning*) dan kemampuan model dalam menggeneralisasi data baru (*generalization*).

3.9 Metode Klasifikasi

3.9.1 Logistic Regression

Cara Kerja: Algoritma ini menggunakan fungsi statistik Sigmoid untuk menghitung probabilitas sebuah ulasan. Hasil perhitungan yang mendekati nilai 1 diklasifikasikan sebagai sentimen positif, sedangkan yang mendekati 0 diklasifikasikan sebagai negatif.

- Kelebihan: Proses training sangat cepat, ringan secara komputasi, dan hasilnya mudah dipahami melalui nilai probabilitas.

- Kekurangan: Performa cenderung menurun jika terdapat hubungan antar kata yang sangat kompleks atau tidak linear.

3.9.2 Support Vector Machine (SVM)

Cara Kerja: SVM bekerja dengan memetakan data ke ruang dimensi tinggi untuk menemukan Hyperplane (garis pemisah) terbaik. Garis ini ditentukan berdasarkan jarak maksimal (Margin) antara dua kelas sentimen yang berbeda untuk meminimalkan kesalahan prediksi.

- Kelebihan: Sangat akurat dalam menangani ribuan fitur kata hasil TF-IDF dan lebih tahan terhadap masalah *overfitting*.
- Kekurangan: Membutuhkan waktu proses yang lebih lama dan penggunaan memori yang lebih besar dibandingkan Logistic Regression.

3.10 Evaluasi Model

Tahap evaluasi dilakukan untuk mengukur sejauh mana keandalan model Support Vector Machine (SVM) dan Logistic Regression dalam mengklasifikasikan sentimen ulasan. Proses penilaian ini didasarkan pada Confusion Matrix, yaitu tabel kontingensi yang membandingkan hasil prediksi model dengan label asli (data aktual). Melalui matriks ini, diperoleh empat indikator utama untuk menghitung metrik performa sebagai berikut:

- Akurasi (*Accuracy*): Menggambarkan tingkat kedekatan antara hasil prediksi model dengan data keseluruhan. Metrik ini menunjukkan persentase total prediksi yang benar (positif dan negatif) dari seluruh jumlah data uji.
- Presisi (*Precision*): Menilai kualitas prediksi model pada kelas tertentu. Metrik ini merepresentasikan rasio ulasan yang benar-benar positif dibandingkan dengan seluruh ulasan yang diprediksi positif oleh model.
- Recall (*Sensitivity*): Mengukur kemampuan model dalam menemukan kembali seluruh data yang berlabel positif. Metrik ini menunjukkan

- seberapa banyak ulasan positif yang berhasil diidentifikasi secara tepat dari total keseluruhan data positif yang ada.
- F1-Score: Merupakan nilai rata-rata harmonik yang menggabungkan presisi dan *recall*. Metrik ini digunakan untuk melihat keseimbangan performa model, terutama jika terjadi ketimpangan jumlah data antara kelas positif dan negatif.

Penerapan keempat metrik ini bertujuan untuk memastikan bahwa model yang dihasilkan tidak hanya memiliki akurasi yang tinggi secara angka, tetapi juga memiliki kemampuan klasifikasi yang konsisten dan objektif pada setiap kategori sentimen.

3.11 Flowchart Penelitian

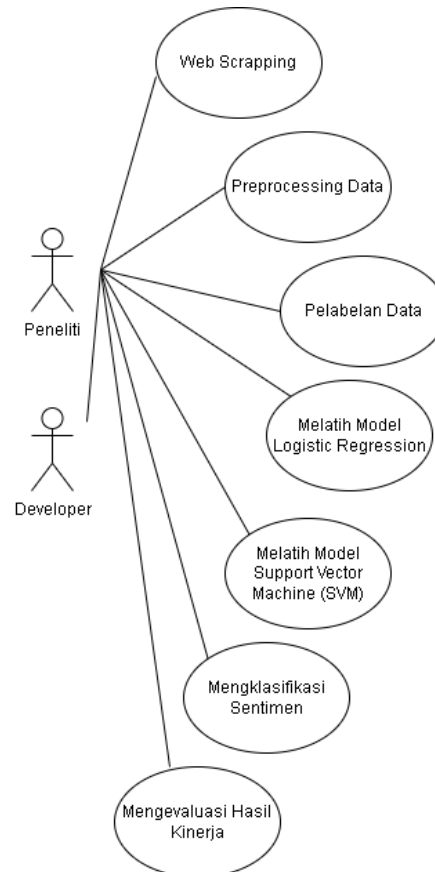


Gambar 3.1 Flowchart Proses Analisis Sentimen

Flowchart ini menggambarkan alur proses analisis sentimen terhadap ulasan pengguna game e-Football. Proses dimulai dari pengambilan data ulasan melalui teknik web scraping, dilanjutkan dengan validasi dan preprocessing data (tokenisasi, penghapusan stopwords, dan stemming). Setelah data diproses, dilakukan pelabelan sentimen menjadi positif atau negatif, yang kemudian diklasifikasikan menggunakan algoritma Logistic Regression dan Support Vector

Machine (SVM). Hasil klasifikasi dievaluasi menggunakan metrik akurasi, precision, recall, dan F1-score.

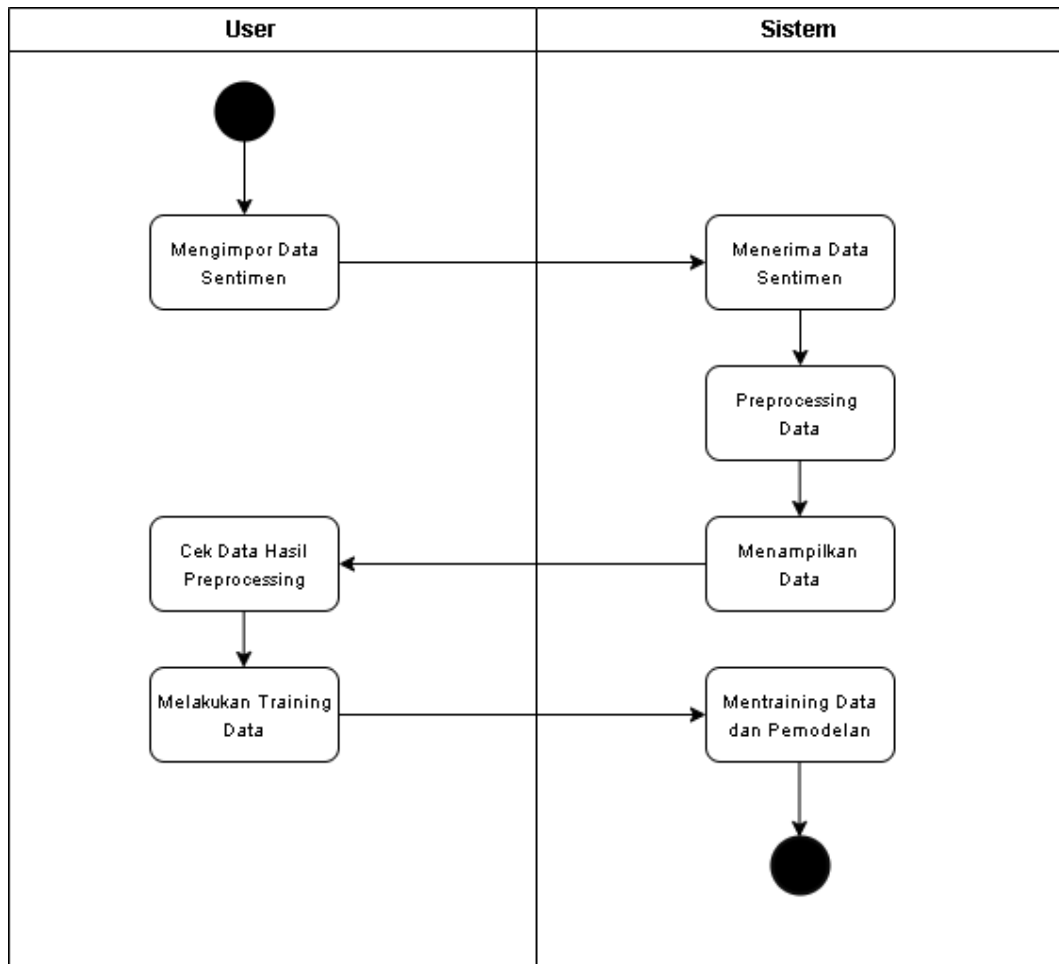
3.12 Use Case Diagram



Gambar 3.2 Use Case Diagram

Use case diagram ini menunjukkan interaksi antara peneliti dengan sistem analisis sentimen terhadap ulasan pengguna game e-Football. Peneliti berperan dalam seluruh proses utama, mulai dari melakukan web scraping data ulasan, preprocessing data, pelabelan sentimen, pelatihan model dengan algoritma Logistic Regression dan SVM, hingga proses klasifikasi dan evaluasi hasil kinerja model. Diagram ini menggambarkan peran sentral peneliti dalam mengelola seluruh tahapan analisis.

3.13 Activity Diagram



Gambar 3.3 Activity Diagram

Gambar ini menggambarkan kolaborasi antara user dan sistem dalam proses pengolahan dan analisis data sentimen secara efektif.

BAB IV

HASIL DAN PEMBAHASAN

4.1 Scrapping Data

Scraping data merupakan metode pengumpulan data otomatis dari sumber daring dengan cara mengekstraksi informasi yang terdapat pada struktur dokumen digital, umumnya berupa HTML. Proses ini dilakukan menggunakan perangkat lunak atau pustaka pemrograman (misalnya *BeautifulSoup* atau *Selenium*) untuk mengakses, menyalin, dan menyimpan data yang relevan ke dalam format terstruktur seperti CSV atau database. Dalam penelitian akademik, teknik ini banyak dimanfaatkan untuk mengakses ulasan aplikasi, data media sosial, maupun informasi pada situs e-commerce yang tidak menyediakan API resmi. Melalui scraping data, peneliti dapat memperoleh data dalam jumlah besar secara efisien dan real-time, sehingga mendukung analisis sentimen, *data mining*, maupun *machine learning* (Fadhillah, 2024).

Menurut Fadhillah (2024), penerapan *data scraping* pada ulasan aplikasi di Google Play Store mampu menghasilkan dataset yang kaya dan autentik untuk analisis berbasis aspek. Teknik ini terbukti efektif dalam mengumpulkan ribuan ulasan yang kemudian diproses melalui tahap *preprocessing* sebelum dilakukan klasifikasi dengan algoritma *Multinomial Logistic Regression*. Dengan demikian, scraping data bukan sekadar teknik pengumpulan informasi, melainkan fondasi penting dalam riset akademik berbasis *big data* dan kecerdasan buatan.

Pada tahap ini, peneliti menggunakan library *google-play-scraper*. Kode tersebut berfungsi untuk menginstal dan memanfaatkan tools scraping yang dapat mengambil data aplikasi secara otomatis. Selain itu, peneliti juga mempersiapkan tools analisis data dengan mengimpor library *Pandas* (untuk pengelolaan data) dan *NumPy* (untuk proses numerik). Proses scraping dilakukan dengan mendefinisikan aplikasi target melalui *jp.konami.pesam*, kemudian mengambil seluruh ulasan terbaru berbahasa Indonesia dari pengguna di wilayah Indonesia

Cleaning, yaitu proses penghapusan karakter yang tidak relevan seperti tanda baca, angka, simbol, URL, dan emotikon yang tidak berpengaruh terhadap analisis sentimen.

content	cleaning
game rusak. pengalaman yang sangat buruk bange...	game rusak pengalaman yang sangat buruk banget...
jangan pada main efootball lagi mulai sekarang...	jangan pada main efootball lagi mulai sekarang...
efotball plis lah kenapa sih pas kalo main ama...	efotball plis lah kenapa sih pas kalo main ama...
Apalah nih game udah sering stuck di ingame ja...	Apalah nih game udah sering stuck di ingame ja...
Lemot engga akan maju nih game kalo gini terus...	Lemot engga akan maju nih game kalo gini terus...

Gambar 4.3 Hasil Cleaning

Case folding, yaitu proses mengubah seluruh teks ulasan menjadi huruf kecil untuk menghindari perbedaan makna akibat penggunaan huruf kapital.

cleaning	case_folding
game rusak pengalaman yang sangat buruk banget...	game rusak pengalaman yang sangat buruk banget...
jangan pada main efootball lagi mulai sekarang...	jangan pada main efootball lagi mulai sekarang...
efotball plis lah kenapa sih pas kalo main ama...	efotball plis lah kenapa sih pas kalo main ama...
Apalah nih game udah sering stuck di ingame ja...	apalah nih game udah sering stuck di ingame ja...
Lemot engga akan maju nih game kalo gini terus...	lemot engga akan maju nih game kalo gini terus...

Gambar 4.4 Hasil Case Folding

Normalisasi, Selanjutnya dilakukan proses normalisasi, yaitu mengganti kata-kata tidak baku atau singkatan menjadi bentuk baku. Untuk mendukung proses ini, peneliti menggunakan kamus kata baku yang diunduh dari Github

case_folding	normalisasi
game rusak pengalaman yang sangat buruk banget...	permainan rusak pengalaman yang sangat buruk s...
jangan pada main efootball lagi mulai sekarang...	jangan pada main efootball lagi mulai sekarang...
efotball plis lah kenapa sih pas kalo main ama...	efotball plis lah kenapa sih pas kalo main ama...
apalah nih game udah sering stuck di ingame ja...	apalah nih permainan udah sering stuck di inga...
lemot engga akan maju nih game kalo gini terus...	lemot engga akan maju nih permainan kalo gini ...

Gambar 4.5 Hasil Normalisasi Kata

Tokenizing, yaitu proses pemecahan teks ulasan menjadi unit kata (*token*) agar dapat diproses lebih lanjut.

normalisasi	tokenize
permainan rusak pengalaman yang sangat buruk s...	[permainan, rusak, pengalaman, yang, sangat, b...
jangan pada main efootball lagi mulai sekarang...	[jangan, pada, main, efootball, lagi, mulai, s...
efotball plis lah kenapa sih pas kalo main ama...	[efotball, plis, lah, kenapa, sih, pas, kalo, ...
apalah nih permainan udah sering stuck di inga...	[apalah, nih, permainan, udah, sering, stuck, ...

Gambar 4.6 Hasil Tokenize

Stopword removal, yaitu proses penghapusan kata-kata umum yang tidak memiliki kontribusi signifikan terhadap penentuan sentimen, seperti kata penghubung dan kata keterangan umum.

tokenize	stopword removal
[permainan, rusak, pengalaman, yang, sangat, b...	[permainan, rusak, pengalaman, buruk, bermain,...
[jangan, pada, main, efootball, lagi, mulai, s...	[main, efootball, stopp, buang, kuota, doangg,...
[efotball, plis, lah, kenapa, sih, pas, kalo, ...	[efotball, plis, sih, pas, kalo, main, ama, or...

Gambar 4.7 Hasil Stopword Removal

Stemming, yaitu proses mengubah kata ke dalam bentuk dasar untuk mengurangi variasi kata yang memiliki makna yang sama.

stopword removal	stemming_data
[permainan, rusak, pengalaman, buruk, bermain,...	main rusak alam buruk main main buang buang pl...
[main, efootball, stopp, buang, kuota, doangg,...	main efootball stopp buang kuota doangg banyak...
[efotball, plis, sih, pas, kalo, main, ama, or...	efotball plis sih pas kalo main ama orang tuh ...
[apalah, nih, permainan, udah, stuck, ingame, ...	apa nih main udah stuck ingame jaring bagus le...

Gambar 4.8 Hasil Stemming Data

termasuk tokenisasi, stopwords removal, dan stemming. Setiap nilai mempresentasikan bobot kepentingan suatu kata terhadap dokumen tertentu, dimana nilai bobot yang lebih besar menunjukkan kata tersebut lebih relevan dalam mempresentasikan isi ulasan.

	Ulasan_Stemming	main	nya	bagus	update	tolong	jaring
0	main rusak alam buruk main main buang buang pl...	0.176143	0.044418	0.045109	0.000000	0.000000	0.123089
1	main efootball stopp buang kuota doangg banyak...	0.092459	0.000000	0.000000	0.056327	0.000000	0.000000
2	efotball plis sih pas kalo main ama orang tuh ...	0.232999	0.282028	0.000000	0.000000	0.081057	0.000000
3	apa nih main udah stuck ingame jaring bagus le...	0.108827	0.000000	0.055739	0.000000	0.000000	0.076049
4	lot engga maju nih main kalo gin main play aca...	0.237470	0.000000	0.072977	0.000000	0.000000	0.000000
5	main anjg passing bug ganti main bug pas adu s...	0.164900	0.000000	0.000000	0.000000	0.000000	0.000000
6	efotball main cocok main offline vs ai saran ...	0.237858	0.000000	0.000000	0.000000	0.000000	0.000000
7	laga vs ai wajar terkadang sinyal stabil main ...	0.075394	0.000000	0.000000	0.000000	0.000000	0.000000
8	main gravik nya malesin stadion nya ganti	0.092777	0.280748	0.000000	0.000000	0.000000	0.000000
9	klw main online lawan player ngelagpatah patah...	0.065443	0.099018	0.000000	0.000000	0.000000	0.000000

Gambar 4.11 Hasil TF-IDF

4.4 Split Data

Setelah proses pembobotan selesai, data dibagi menjadi dua bagian menggunakan metode train-test split. Pembagian dilakukan dengan rasio 80% untuk data latih dan 20% untuk data uji, agar model bisa dilatih dan dievaluasi secara terpisah.

```
Ukuran X_train: (6000, 11495)
Ukuran X_test : (1500, 11495)
Ukuran y_train: (6000,)
Ukuran y_test : (1500,)
```

Gambar 4.12 Hasil Split Data

4.5 Hasil Klasifikasi Dengan Logistic Regression

Bagian ini menyajikan dan menganalisis hasil implementasi model klasifikasi yang dikembangkan. Data Uji (Testing set) yang digunakan berjumlah 1500 intans data. Jumlah ini merupakan 20% dari total 7479 sampel keseluruhan, setelah melalui proses pemisahan (Data Splitting) dengan rasio 80% untuk

pelatihan (Training). Setiap instans dievaluasi berdasarkan 9 atribut/fitur yang merupakan hasil dari tahap pra-pemrosesan (*preprocessing*), termasuk operasi stemming dan stopwords removal.

```

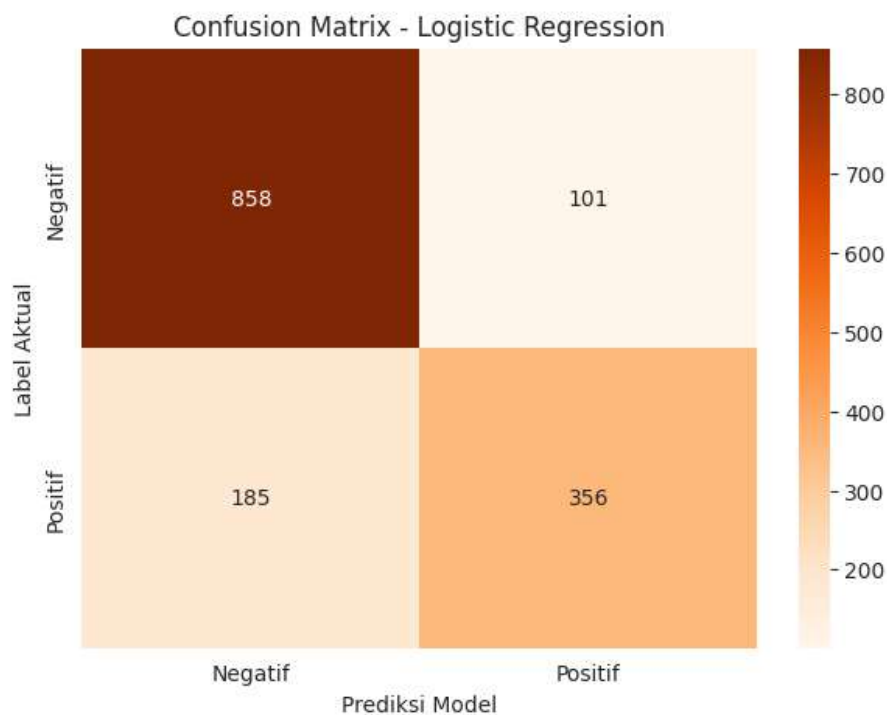
AKURASI LOGISTIC REGRESSION: 80.93%

LAPORAN KLASIFIKASI LENGKAP:

```

	precision	recall	f1-score	support
Negatif	0.82	0.89	0.86	959
Positif	0.78	0.66	0.71	541
accuracy			0.81	1500
macro avg	0.80	0.78	0.79	1500
weighted avg	0.81	0.81	0.81	1500

Gambar 4.13 Hasil Klasifikasi Logistic Regression



Gambar 4.14 Confusion Matrix

Berdasarkan hasil pengujian model Logistic Regression pada data testing, diperoleh confusion matrix yang menunjukkan jumlah prediksi benar dan salah pada masing-masing kelas sentimen.

- Nilai True Positive (TP) sebanyak 356 menunjukkan data sentimen positif yang berhasil diprediksi dengan benar.
- Nilai True Negative (TN) sebanyak 858 menunjukkan data sentimen negatif yang berhasil diprediksi dengan benar.
- Nilai False Positive (FP) sebanyak 101 merupakan data sentimen negatif yang salah diprediksi sebagai sentimen positif.
- Nilai False Negative (FN) sebanyak 185 merupakan data sentimen positif yang salah diprediksi sebagai sentimen negatif.

4.6 Hasil Klasifikasi Dengan SVM (Support Vector Machine)

Bagian ini menyajikan dan menganalisis kinerja model Support Vector Machine (SVM) yang diimplementasikan untuk klasifikasi sentimen. Model SVM diuji menggunakan 1500 instans data uji (20% dari total 7497 sampel) yang terdiri dari 9 atribut/fitur hasil *preprocessing*.

```

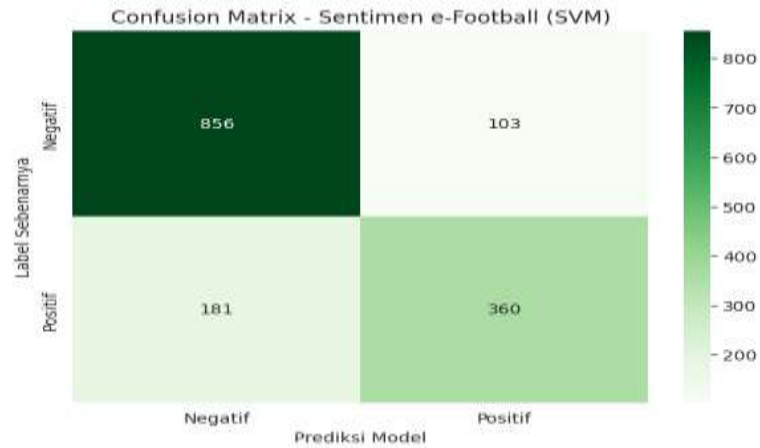
AKURASI MODEL SVM: 81.07%

LAPORAN KLASIFIKASI:

```

	precision	recall	f1-score	support
Negatif	0.83	0.89	0.86	959
Positif	0.78	0.67	0.72	541
accuracy			0.81	1500
macro avg	0.80	0.78	0.79	1500
weighted avg	0.81	0.81	0.81	1500

Gambar 4.15 Hasil Klasifikasi SVM



Gambar 4.16 Confusion Matrix

Berdasarkan hasil pengujian model Support Vector Machine (SVM) pada data testing, diperoleh confusion matrix yang menunjukkan jumlah prediksi benar dan salah pada masing-masing kelas sentimen.

- Nilai True Positive (TP) sebanyak 362 menunjukkan data sentimen positif yang berhasil diprediksi dengan benar.
- Nilai True Negative (TN) sebanyak 848 menunjukkan data sentimen negatif yang berhasil diprediksi dengan benar.
- Nilai False Positive (FP) sebanyak 111 merupakan data sentimen negatif yang salah diprediksi sebagai sentimen positif.
- Nilai False Negative (FN) sebanyak 179 merupakan data sentimen positif yang salah diprediksi sebagai sentimen negatif.

4.7 Hasil Evaluasi Logistic Regression

Akurasi model Logistic Regression pada data testing diperoleh sebesar 80,97%, yang dihitung berdasarkan nilai *confusion matrix* menggunakan rumus sebagai berikut:

$$\begin{aligned}
 \text{Akurasi} &= \frac{TP + TN}{TP + TN + FP + FN} \\
 &= \frac{356 + 858}{356 + 858 + 101 + 185} \\
 &= \frac{1214}{1500} = 0.81
 \end{aligned}$$

Hasil tersebut menunjukkan bahwa model Logistic Regression mampu memprediksi data *testing* dengan benar sebesar 80,97% dari keseluruhan data yang digunakan dalam proses pengujian.

Classification Report

Tabel 4.1 Confusion Matrix Tanpa Nilai

	Prediksi Negatif	Prediksi Positif
Aktual Negatif	TN	FP
Aktual Positif	FN	TP

Tabel 4.2 Confusion Matrix Keseluruhan

	Prediksi Negatif	Prediksi Positif
Aktual Negatif	858	101
Aktual Positif	185	356

1. Sudut Pandang Kelas Positif

(positif dianggap sebagai “positif” dalam perhitungan metrik)

- **TP = 356** (data positif yang berhasil diprediksi positif)
- **FN = 185** (data positif yang salah diprediksi negatif)
- **FP = 101** (data negatif yang salah diprediksi positif)
- **TN = 858** (data negatif yang berhasil diprediksi negatif)

Tabel 4.3 Confusion Matrix Keseluruhan

	Prediksi Negatif	Prediksi Positif	Total
Aktual Negatif	TN = 858	FP = 101	959
Aktual Positif	FN = 185	TP = 356	541

Presisi (Positif)

$$\text{Presisi} = \frac{TP}{TP + FP} = \frac{356}{356 + 101} = \frac{356}{457} = 0,7790 = 77,90\%$$

Recall (Positif)

$$\text{Recall} = \frac{TP}{TP + FN} = \frac{356}{356 + 185} = \frac{356}{541} = 0,6580 = 65,80\%$$

f1 Score (Positif)

$$\begin{aligned} F1 &= 2 \times \frac{\text{Presisi} \times \text{Recall}}{\text{Presisi} + \text{Recall}} \\ &= 2 \times \frac{0,7790 \times 0,6580}{0,7790 + 0,6580} = \frac{1,0251}{1,4370} = 0,7133 = 71,33\% \end{aligned}$$

2. Sudut Pandang Kelas Negatif

(negatif dianggap sebagai “positif” dalam perhitungan metrik)

- **TP = 356** (data positif yang berhasil diprediksi positif)
- **FN = 185** (data positif yang salah diprediksi negatif)
- **FP = 101** (data negatif yang salah diprediksi positif)
- **TN = 858** (data negatif yang berhasil diprediksi negatif)

Tabel 4.4 Confusion Matrix Keseluruhan

	Prediksi Negatif	Prediksi Positif	Total
Aktual Negatif	TN = 858	FP = 101	959
Aktual Positif	FN = 185	TP = 356	541

Presisi (Negatif)

$$\text{Presisi} = \frac{TN}{TN + FN} = \frac{858}{858 + 185} = \frac{858}{1043} = 0,8226 = 82,26\%$$

Recall (Negatif)

$$\text{Recall} = \frac{TN}{TN + FN} = \frac{858}{858 + 101} = \frac{858}{959} = 0,8946 = 89,46\%$$

f1 Score (Negatif)

$$\begin{aligned} F1 &= 2 \times \frac{\text{Presisi} \times \text{Recall}}{\text{Presisi} + \text{Recall}} \\ &= 2 \times \frac{0,8226 \times 0,8946}{0,8226 + 0,8946} = \frac{1,4717}{1,7172} = 0,8570 = 85,70\% \end{aligned}$$

4.8 Hasil Evaluasi SVM (Support Vector Machine)

Akurasi model Support Vector Machine (SVM) pada data *testing* diperoleh sebesar 81,13%, yang dihitung berdasarkan nilai *confusion matrix* menggunakan rumus sebagai berikut:

$$\begin{aligned} \text{Akurasi} &= \frac{TP + TN}{TP + TN + FP + FN} \\ &= \frac{359 + 858}{359 + 858 + 101 + 182} \\ &= \frac{1217}{1500} = 81,13\% \end{aligned}$$

Hasil tersebut menunjukkan bahwa model Support Vector Machine (SVM) mampu memprediksi data *testing* dengan benar sebesar 81,13% dari keseluruhan data yang digunakan dalam proses pengujian.

Classification Report

Tabel 4.5 Confusion Matrix Tanpa Nilai

	Prediksi Negatif	Prediksi Positif
Aktual Negatif	TN	FP
Aktual Positif	FN	TP

Tabel 4.6 Confusion Matrix Keseluruhan

	Prediksi Negatif	Prediksi Positif
Aktual Negatif	856	103
Aktual Positif	181	360

1. Sudut Pandang Kelas Positif

(positif dianggap sebagai “positif” dalam perhitungan metrik)

- **TP = 360** (data positif yang berhasil diprediksi positif)
- **FN = 181** (data positif yang salah diprediksi negatif)
- **FP = 103** (data negatif yang salah diprediksi positif)
- **TN = 856** (data negatif yang berhasil diprediksi negatif)

Tabel 4.7 Confusion Matrix Keseluruhan

	Prediksi Negatif	Prediksi Positif	Total
Aktual Negatif	TN = 856	FN = 181	959
Aktual Positif	FP = 103	TP = 360	541

Presisi(Positif)

$$\text{Presisi} = \frac{TP}{TP + FP} = \frac{360}{360 + 103} = \frac{360}{463} = 0,7775 = 77,75\%$$

Recall (Positif)

$$\text{Recall} = \frac{TP}{TP + FN} = \frac{360}{360 + 181} = \frac{360}{541} = 0,6654 = 66,54\%$$

f1 Score (Positif)

$$\begin{aligned} F1 &= 2 \times \frac{\text{Presisi} \times \text{Recall}}{\text{Presisi} + \text{Recall}} \\ &= 2 \times \frac{0,7775 \times 0,6654}{0,7775 + 0,6654} = \frac{1,0347}{1,4429} = 0,7171 = 71,71\% \end{aligned}$$

2. Sudut Pandang Kelas Negatif

(negatif dianggap sebagai “positif” dalam perhitungan metrik)

- **TP = 360** (data positif yang berhasil diprediksi positif)
- **FN = 181** (data positif yang salah diprediksi negatif)
- **FP = 103** (data negatif yang salah diprediksi positif)
- **TN = 856** (data negatif yang berhasil diprediksi negatif)

Tabel 4.8 Confusion Matrix Keseluruhan

	Prediksi Negatif	Prediksi Positif	Total
Aktual Negatif	TN = 856	FN = 181	959
Aktual Positif	FP = 103	TP = 360	541

Presisi (Negatif)

$$Presisi = \frac{TN}{TN + FN} = \frac{856}{856 + 181} = \frac{856}{1037} = 0,8254 = 82,54\%$$

Recall (Negatif)

$$Recall = \frac{TN}{TN + FP} = \frac{856}{856 + 103} = \frac{856}{959} = 0,8925 = 89,25\%$$

f1 Score (Negatif)

$$F1 = 2 \times \frac{Presisi \times Recall}{Presisi + Recall}$$

$$= 2 \times \frac{0,8254 \times 0,8925}{0,8254 + 0,8925} = \frac{1,4733}{1,7179} = 0,8576 = 85,76\%$$

4.9 Hasil Visualisasi

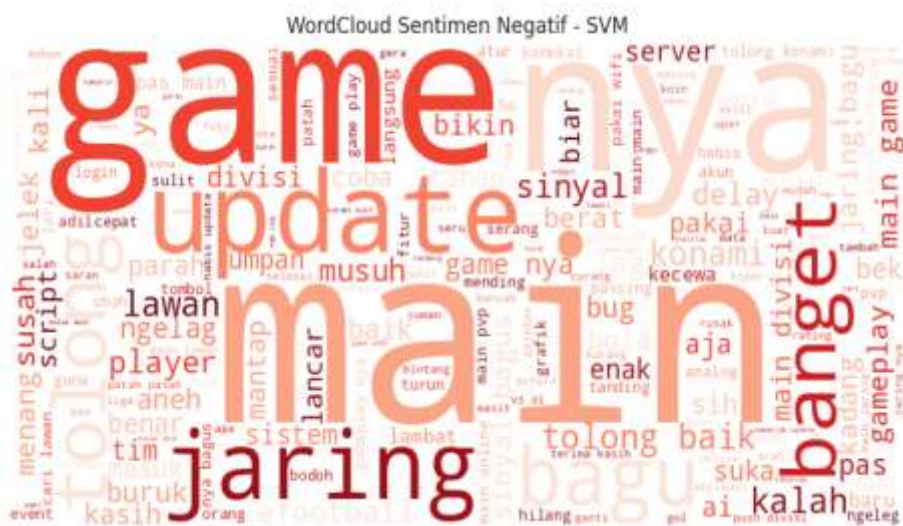
Tahap akhir analisis dilakukan dengan membuat visualisasi WordCloud untuk melihat kata-kata yang paling sering muncul pada masing-masing kategori sentimen.

4.9.2 Hasil Visualisasi SVM (Support Vector Machine)



Gambar 4.19 WorldCloud Sentimen Positif

Visualisasi WordCloud mengungkapkan dominasi kata 'main', 'update', dan 'bagus' pada sentimen positif, yang menunjukkan apresiasi tinggi pengguna terhadap kualitas gameplay serta pembaruan fitur di e-Football mobile

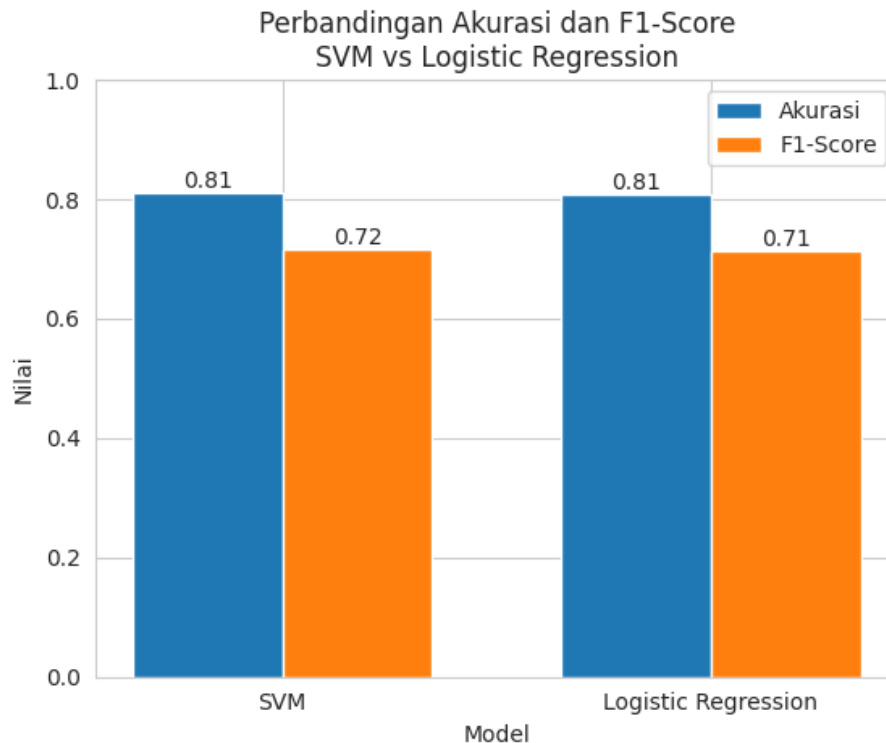


Gambar 4.20 WorldCloud Sentimen Negatif

Visualisasi WordCloud sentimen negatif menonjolkan kata 'lag', 'scripting', dan 'kecewa', yang mengindikasikan keluhan utama pengguna terkait stabilitas koneksi dan sistem permainan.

4.10 Analisis Kinerja Algoritma Logistic Regression Dan SVM (Support Vector Machine)

Berikut adalah hasil visualisasi antara kedua algoritma Logistic Regression dan SVM (Support Vector Machine) :



Gambar 4.21 Perbandingan Akurasi dan F1-Score

Berdasarkan hasil komparasi performa yang telah dilakukan, dapat disimpulkan bahwa Support Vector Machine (SVM) menunjukkan keunggulan sebagai model klasifikasi yang paling optimal dalam penelitian ini. Secara empiris, meskipun kedua algoritma menghasilkan tingkat akurasi yang sama yaitu sebesar 0,81, namun SVM menunjukkan dominasi pada metrik *F1-score* dengan nilai 0,72 dibandingkan Logistic Regression yang memperoleh 0,71. Capaian *F1-score* yang lebih tinggi ini menjadi indikator penting bahwa SVM memiliki kemampuan yang lebih solid dalam menjaga keseimbangan antara nilai presisi dan *recall*. Hal ini membuktikan bahwa SVM lebih efektif dalam meminimalisir kesalahan klasifikasi, terutama saat menghadapi karakteristik dataset ulasan *e-Football* yang

memiliki sebaran kelas tidak seimbang. Dengan stabilitas yang lebih baik dalam mengidentifikasi pola sentimen positif maupun negatif secara proporsional, maka Support Vector Machine (SVM) ditetapkan sebagai metode final untuk proses analisis dalam penelitian ini.

BAB V

KESIMPULAN DAN SARAN

5.1 Kesimpulan

Berdasarkan hasil penelitian dan pembahasan yang telah dilakukan mengenai analisis sentimen ulasan pengguna aplikasi *e-Football* menggunakan algoritma *Support Vector Machine* (SVM) dan *Logistic Regression*, maka dapat ditarik kesimpulan sebagai berikut:

1. Algoritma *Support Vector Machine* (SVM) menunjukkan performa yang lebih unggul dibandingkan *Logistic Regression* dalam mengklasifikasikan sentimen ulasan pengguna di Google Play Store. Hal ini dibuktikan oleh nilai evaluasi model yang lebih tinggi, di mana SVM memperoleh akurasi sebesar 0,81 dan *f1-score* sebesar 0,72, sementara *Logistic Regression* memiliki akurasi sebesar 0,81 namun dengan *f1-score* yang lebih rendah yaitu 0,71.
2. *Support Vector Machine* (SVM) terbukti lebih stabil dan seimbang dalam mengenali kelas sentimen positif maupun negatif. Keunggulan nilai *f1-score* pada SVM menunjukkan bahwa algoritma ini lebih efektif dalam menangani tantangan klasifikasi pada data yang memiliki persebaran tidak seimbang antara ulasan positif dan negatif.
3. Penelitian ini mengonfirmasi bahwa algoritma *Support Vector Machine* (SVM) merupakan metode yang lebih optimal dan akurat untuk digunakan dalam tugas klasifikasi sentimen berbasis teks pada ulasan aplikasi seluler, khususnya dalam memberikan gambaran tingkat kepuasan pengguna terhadap aplikasi *e-Football*.
4. Dataset yang digunakan dalam penelitian ini berjumlah 7.497 ulasan yang terbagi menjadi 5.997 data latih dan 1.500 data uji dengan rasio pembagian 80:20. Seluruh data telah melalui tahapan *preprocessing* yang sistematis (seperti *cleaning*, *case folding*, *normalisasi*, *tokenizing*, *stopword removal*, dan *stemming*) serta ekstraksi fitur menggunakan

5. metode TF-IDF sebelum memasuki tahap evaluasi model menggunakan metrik akurasi, presisi, *recall*, dan *f1-score*.

5.2 Saran

Untuk penyempurnaan studi di masa mendatang, terdapat beberapa rekomendasi yang dapat dipertimbangkan oleh peneliti selanjutnya:

1. Peneliti selanjutnya disarankan untuk menambahkan kategori label netral agar sistem dapat membedakan antara opini yang benar-benar memiliki sentimen dengan ulasan yang hanya bersifat informasi umum atau saran teknis kepada pengembang.
2. Perluasan dataset dengan mengambil sumber ulasan dari platform media sosial lain seperti Twitter atau forum komunitas e-Football di Reddit sangat dianjurkan untuk memperoleh variasi bahasa dan konteks sentimen yang lebih luas selain dari Google Play Store.
3. Pengembangan kamus normalisasi slang dan kata baku perlu dilakukan secara lebih mendalam mengingat banyaknya istilah spesifik dalam komunitas game e-Football seperti *scripting*, *gacha*, dan *bapuk* yang sering kali belum terakomodasi dalam kamus bahasa Indonesia umum.
4. Penggunaan metode pembobotan kata selain TF-IDF atau penerapan algoritma berbasis *Deep Learning* seperti *Long Short-Term Memory (LSTM)* dapat dipertimbangkan pada penelitian mendatang untuk melihat perbandingan performa dalam menangani konteks kalimat yang lebih kompleks dan panjang.
5. Penambahan teknik *Hyperparameter Tuning* seperti *Grid Search* atau *Random Search* pada algoritma SVM dan *Logistic Regression* disarankan guna mendapatkan parameter model yang paling optimal dalam meningkatkan nilai akurasi dan *F1-Score*.

DAFTAR PUSTAKA

- Alengka, S. G. (2025). *Analisis Sentimen Game Mobile Legends di Google Play Store Menggunakan Metode Support Vector Machine dan Logistic Regression*. [Skripsi, Universitas Amikom Purwokerto]. Repository Amikom Purwokerto. <https://eprints.amikompurwokerto.ac.id/id/eprint/2926/>
- Hanafi, M. R., & R, R. K. (2024). Analisis Sentimen pada Ulasan Aplikasi Sirekap di Google Play Menggunakan Algoritma Naive Bayes. *MALCOM: Indonesian Journal of Machine Learning and Computer Science*, 4(4), 1578-1586. <https://journal.irpi.or.id/index.php/malcom/article/view/1693>
- Hapsari, G. P. P. (2020). *Rancang Bangun Sistem Informasi Manajemen Laboratorium Berbasis Web Menggunakan Unified Modeling Language (UML)*. [Skripsi, Universitas Islam Indonesia]. <https://dspace.uii.ac.id/handle/123456789/18616>
- Haris, M., Suharso, A., & Nurkifli, E. H. (2024). Analisis Sentimen pada Game eFootball di Google Play Store Menggunakan Algoritma IndoBERT. *JATI (Jurnal Mahasiswa Teknik Informatika)*, 8(6), 12108-12115. <https://www.ejournal.itn.ac.id/jati/article/view/11810>
- Irawaty. (2024). Mengenal Sistem Informasi Berbasis Cloud: Keuntungan dan Tantangannya dalam Dunia Bisnis. *AKSELERASI: Jurnal Ilmiah Nasional*, 6(3), 151-161. <https://ejournal.goacademica.com/index.php/ja/article/view/1221>
- Kulsum, U., Jajuli, M., & Sulistiyowati, N. (2022). Analisis Sentimen Aplikasi WETV di Google Play Store Menggunakan Algoritma Support Vector Machine. *Journal of Applied Informatics and Computing (JAIC)*, 6(2), 205-212. <https://jurnal.polibatam.ac.id/index.php/JAIC/article/view/4802>
- Prasetyo, A. (2024). Perbandingan Algoritma Logistic Regression, SVM, dan Random Forest pada Analisis Sentimen Aplikasi Gopay. *Jurnal Informatika -*

Ejournal Poltek Harber, 5(2).

<https://ejournal.poltekharber.ac.id/index.php/informatika/article/view/8796>

Putra, I. G. A., & Zuliarso, E. (2025). Analisis Sentimen Berdasarkan Ulasan Google Play Store pada Aplikasi DANA Menggunakan Metode Support Vector Machine (SVM). *INTECOMS: Journal of Information Technology and Computer Science*, 8(5).

<https://journal.ipm2kpe.or.id/index.php/INTECOM/article/view/16347>

Ramadhan, G. F., dkk. (2024). Analisis Sentimen Ulasan Aplikasi Mobile Menggunakan Algoritma Klasifikasi dan Text Mining. *JITET (Jurnal Informatika dan Teknik Elektro Terapan)*, 12(3).

<https://journal.eng.unila.ac.id/index.php/jitet/article/view/6191>

Rinardi, M., dkk. (2024). Analisis Sentimen Terhadap Marketplace Menggunakan Algoritma Naïve Bayes. *JATI (Jurnal Mahasiswa Teknik Informatika)*, 8(4).

<https://ejournal.itn.ac.id/jati/article/view/10965>

Sihotang, J. I., Siregar, J., & Simanjuntak, M. (2023). Analisis Sentimen Terhadap Pembayaran Digital Menggunakan Algoritma Support Vector Machine. *Tekinkom (Teknik Informasi dan Komputer)*, 6(1), 322-329.

<https://www.jurnal.murnisadar.ac.id/index.php?journal=Tekinkom&page=article&op=view&path%5B%5D=773>

Susanto, D., dkk. (2022). Analisis Sentimen Pada Game E-Football Di Google Play Store Menggunakan Algoritma IndoBERT. *JATI (Jurnal Mahasiswa Teknik Informatika)*, 6(2). <https://ejournal.itn.ac.id/jati/article/view/11810>

Wati, M. C., dkk. (2022). Analisis Sentimen Pengguna Aplikasi Mobile pada Google Play Store Menggunakan Metode Naïve Bayes. *JTS: Jurnal Teknologi dan Sistem*, 1(2). <https://journal.admi.or.id/index.php/JTS/article/view/434>

Yuspriyadi, A., dkk. (2023). Perbandingan Algoritma Naïve Bayes dan Support Vector Machine (SVM) dalam Analisis Sentimen Kebijakan Kemdikbudristek. *Jurnal Processor*, 18(1).

<https://ejournal.unama.ac.id/index.php/processor/article/view/897>

Taufiq, R., dkk. (2021). Perancangan Sistem Informasi Penjualan Berbasis Web Menggunakan Unified Modeling Language (UML). *JUSITI: Jurnal Sistem Informasi dan Teknologi Informasi*, 10(1).

<https://ejournal.undipa.ac.id/index.php/jusiti/article/view/62>

Andriyanto, A., dkk. (2024). Implementasi Machine Learning Untuk Analisis Sentimen Ulasan Aplikasi Pada Google Play Store Menggunakan Google Colab. *JUSTIFY : Jurnal Sistem Informasi Ibrahimy*, 3(1).

<https://journal.ibrahimy.ac.id/index.php/JUSTIFY/article/view/5569>

Kurniawan, H., dkk. (2024). Pemanfaatan Google Colab Dalam Pengembangan Model Klasifikasi Data Berbasis Artificial Intelligence. *Jurnal Teknologi Informasi*, 9(1). <https://ejournal.urindo.ac.id/index.php/TI/article/view/5223>

Al Amin, I. H., dkk. (2024). Analisis Sentimen Terhadap Ulasan Aplikasi Grab pada Google Play Store Menggunakan Algoritma Logistic Regression. *Jurnal Informatika: Jurnal Pengembangan IT*, 9(1).

<https://ejournal.poltekharber.ac.id/index.php/informatika/article/view/8796>

A'yun, Q., dkk. (2024). Analisis Sentimen Ulasan Pengguna Aplikasi MyPertamina pada Google Play Store Menggunakan Algoritma Support Vector Machine. *G-Tech: Jurnal Teknologi Terapan*, 8(3), 1332-1341.

<https://ejournal.uniramalang.ac.id/g-tech/article/view/5089>

Haq, M. F. I., Rosyadi, I., Nasir, M., & Khambali, A. (2024). *Sentiment analysis of Livin app reviews on Google Play Store*. *Jurnal Surya Informatika*, 14(1), 24–29. https://jurnal.umpp.ac.id/index.php/surya_informatika/article/view/1785

LAMPIRAN

Lampiran ini berisi script Python yang digunakan dalam analisis sentimen menggunakan algoritma Logistic Regression dan SVM (Support Vector Machine):

1. Scrapping Data

```
from google_play_scraper import Sort, reviews

# mengambil data ulasan dari google play store
result, continuation_token = reviews(
    'jp.konami.pesam',
    lang='id',
    country='id',
    sort=Sort.MOST_RELEVANT,
    count=7500,
    filter_score_with=None
```

2. Data Cleaning

```
import re
import string
import nltk

# menghapus URL
def remove_URL(tweet):
    if tweet is not None and isinstance(tweet, str):
        url = re.compile(r'https?://\S+|www\.\S+')
        return url.sub(r'', tweet)
    else:
        return tweet

# menghapus HTML
def remove_html(tweet):
    if tweet is not None and isinstance(tweet, str):
        html = re.compile(r'<.*?>')
        return html.sub(r'', tweet)
    else:
        return tweet
```

```

# menghapus simbol
def remove_symbols(tweet):
    if tweet is not None and isinstance(tweet, str):
        tweet = re.sub(r'[^a-zA-Z0-9\s]', '', tweet)
        return tweet
    else:
        return tweet

# menghapus angka
def remove_numbers(tweet):
    if tweet is not None and isinstance(tweet, str):
        tweet = re.sub(r'\d', '', tweet)
        return tweet
    else:
        return tweet

df['cleaning'] = df['content'].apply(lambda x : remove_URL(x))
df['cleaning'] = df['cleaning'].apply(lambda x : remove_html(x))
df['cleaning'] = df['cleaning'].apply(lambda x : remove_emoji(x))
df['cleaning'] = df['cleaning'].apply(lambda x : remove_symbols(x))
df['cleaning'] = df['cleaning'].apply(lambda x : remove_numbers(x))

df.head(20)

```

3. Case Folding

```

def case_folding(text):
    if isinstance(text, str):
        lowercase_text = text.lower()
        return lowercase_text
    else:
        return text

df['case_folding'] = df['cleaning'].apply(case_folding)
df.head(20)

```

4. Normalisasi

```

import pandas as pd

#fungsi penggantian kata tidak baku
def replace_taboo_words(text, kamus_tidak_baku):
    if isinstance(text, str):
        words = text.split()
        replaced_words = []
        kalimat_baku = []
        kata_diganti = []
        kata_tidak_baku_hash = []

        for word in words:
            if word in kamus_tidak_baku:
                baku_word = kamus_tidak_baku[word]
                if isinstance(baku_word, str) and all(char.isalpha() or char.isspace() for char in baku_word):
                    replaced_words.append(baku_word)
                    kalimat_baku.append(baku_word)
                    kata_diganti.append(word)
                    kata_tidak_baku_hash.append(hash(word))
            else:
                replaced_words.append(word)
        replaced_text = ' '.join(replaced_words)
    else:
        replaced_text = ''
        kalimat_baku = []
        kata_diganti = []
        kata_tidak_baku_hash = []

    return replaced_text, kalimat_baku, kata_diganti, kata_tidak_baku_hash

```

5. Tokenisasi

```
def tokenize(text):
    tokens = text.split()
    return tokens

# Tokenisasi (ubah menjadi daftar kata)
df['tokenize'] = df['normalisasi'].apply(tokenize)

df.head(40)
```

6. Stopwords

```
from nltk.corpus import stopwords
nltk.download('stopwords')
stop_words = stopwords.words('indonesian')

[nltk_data] Downloading package stopwords to /root/nltk_data...
[nltk_data] Unzipping corpora/stopwords.zip.

def remove_stopwords(text):
    return [word for word in text if word not in stop_words]

df['stopword removal'] = df['tokenize'].apply(lambda x: remove_stopwords(x))

df.head(50)
```

7. Stemming

```
from Sastrawi.Stemmer.StemmerFactory import StemmerFactory

from nltk.stem import PorterStemmer
from nltk.stem.snowball import SnowballStemmer

Collecting Sastrawi
  Downloading Sastrawi-1.0.1-py2.py3-none-any.whl.metadata (909 bytes)
  Downloading Sastrawi-1.0.1-py2.py3-none-any.whl (209 kB)
----- 209.7/209.7 kB 7.4 MB/s eta 0:00:00
Installing collected packages: Sastrawi
Successfully installed Sastrawi-1.0.1

factory = StemmerFactory()
stemmer = factory.create_stemmer()

def stem_text(text):
    return [stemmer.stem(word) for word in text]

df['stemming_data'] = df['stopword removal'].apply(lambda x: ' '.join(stem_text(x)) )

df.head(5)
```

8. Labeling

```
import pandas as pd

# Fungsi untuk menentukan sentimen
def determine_sentiment(text):
    positive_count = sum(1 for word in text.split() if word in positive_lexicon)
    negative_count = sum(1 for word in text.split() if word in negative_lexicon)
    if positive_count > negative_count:
        return "Positif"
    elif positive_count < negative_count:
        return "Negatif"

# Baca kamus leksikon positif dan negatif

positive_lexicon = set(pd.read_csv("positive.tsv", sep="\t", header=None)[0])
negative_lexicon = set(pd.read_csv("negative.tsv", sep="\t", header=None)[0])

def replace_none_sentiment(sentiments):
    replace_flag = "Positif"
    for i in range(len(sentiments)):
        if sentiments[i] is None:
            sentiments[i] = replace_flag
            replace_flag = "Negatif" if replace_flag == "Positif" else "Positif"
    return sentiments

df['sentiment'] = df['stemming_data'].apply(determine_sentiment)
df['sentiment'] = replace_none_sentiment(df['sentiment'].tolist())

df.head()
```

9. Mengitung Jumlah Sentimen Positif & Negatif

```
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns

sentiment_count = df['sentiment'].value_counts()
sns.set_style('whitegrid')

fig, ax = plt.subplots(figsize=(6, 4))
ax = sns.barplot(x=sentiment_count.index, y=sentiment_count.values, palette='pastel')
plt.title('Jumlah Analisis Sentimen', fontsize=14, pad=20)
plt.xlabel('Class sentiment', fontsize=12)
plt.ylabel('Jumlah Ulasan', fontsize=12)

total = len(df['sentiment'])

for i, count in enumerate(sentiment_count.values):
    percentage = f'{100 * count / total:.2f}%'
    ax.text(i, count + 0.10, f'{count}\n({percentage})', ha='center', va='bottom')

plt.show()
```

10. Pembagian Data Latih dan Data Uji

```
from sklearn.model_selection import train_test_split

# Membagi Data: 80% untuk latih, 20% untuk uji
X_train, X_test, y_train, y_test = train_test_split(
    X, y,
    test_size=0.2,
    random_state=42
)
```

11. TF-IDF

```
from sklearn.feature_extraction.text import TfidfVectorizer
tfidf = TfidfVectorizer()
X_train_tfidf = tfidf.fit_transform(X_train) # Ulasan latihan jadi angka
X_test_tfidf = tfidf.transform(X_test)      # Ulasan uji jadi angka
```

12. Hasil Klasifikasi Model Logistic Regression

```
# Melatih Logistic Regression
from sklearn.linear_model import LogisticRegression
lr_model = LogisticRegression()
lr_model.fit(X_train_tfidf, y_train)
```

```
#Prediksi & Evaluasi Model
from sklearn.metrics import accuracy_score, classification_report

# Melakukan prediksi pada data uji
y_pred_lr = lr_model.predict(X_test_tfidf)

# Menghitung Skor Akurasi
akurasi_lr = accuracy_score(y_test, y_pred_lr)

print(f"AKURASI LOGISTIC REGRESSION: {akurasi_lr * 100:.2f}%")
print("\nLAPORAN KLASIFIKASI LENGKAP:\n")
print(classification_report(y_test, y_pred_lr))
```

13. Hasil Klasifikasi Model Support Vector Machine

```
# Pemodelan SVM
from sklearn.svm import SVC

# Membuat model SVM dengan kernel Linear
svm_model = SVC(kernel='linear', probability=True)

# Melatih model dengan data latih
svm_model.fit(X_train, y_train)
```

```
#Prediksi & Evaluasi akurasi
from sklearn.metrics import accuracy_score, classification_report

# Prediksi data uji
y_pred = svm_model.predict(X_test)

# Menampilkan Skor
akurasi = accuracy_score(y_test, y_pred)
print(f"AKURASI MODEL SVM: {akurasi * 100:.2f}%")
print("\nLAPORAN KLASIFIKASI:\n")
print(classification_report(y_test, y_pred))
```

14. Evaluasi Model Logistic Regression

```
#Visualisasi Confusion Matrix Logistic Refression (LR)
import matplotlib.pyplot as plt
import seaborn as sns
from sklearn.metrics import confusion_matrix

# Membuat confusion matrix
cm_lr = confusion_matrix(y_test, y_pred_lr)

# Visualisasi dengan Heatmap
plt.figure(figsize=(7, 5))
sns.heatmap(cm_lr, annot=True, fmt='d', cmap='Oranges',
            xticklabels=lr_model.classes_,
            yticklabels=lr_model.classes_)

plt.title('Confusion Matrix - Logistic Regression')
plt.xlabel('Prediksi Model')
plt.ylabel('Label Aktual')
plt.show()
```

15. Evaluasi Model Support Vector Machine

```
import matplotlib.pyplot as plt
import seaborn as sns
from sklearn.metrics import confusion_matrix

# Membuat Confusion Matrix
cm = confusion_matrix(y_test, y_pred)

# visualisasi dengan Heatmap
plt.figure(figsize=(7, 5))
sns.heatmap(cm, annot=True, fmt='d', cmap='Greens',
            xticklabels=svm_model.classes_,
            yticklabels=svm_model.classes_)

plt.title('Confusion Matrix - Sentimen e-Football (SVM)')
plt.xlabel('Prediksi Model')
plt.ylabel('Label Sebenarnya')
plt.show()
```

16. Hasil Kedua Model

```
import matplotlib.pyplot as plt
from sklearn.linear_model import LogisticRegression
from sklearn.svm import SVC
from sklearn.ensemble import VotingClassifier
from sklearn.metrics import accuracy_score, f1_score
```

```
# Logistic Regression
lr = LogisticRegression(max_iter=1000)

# SVM
svm = SVC(kernel='linear', probability=True)
```

```
acc_lr = accuracy_score(y_test_enc, y_pred_lr_enc)
f1_lr = f1_score(y_test_enc, y_pred_lr_enc)

acc_svm = accuracy_score(y_test_enc, y_pred_svm_enc)
f1_svm = f1_score(y_test_enc, y_pred_svm_enc)

acc_ens = accuracy_score(y_test_enc, y_pred_ens_enc)
f1_ens = f1_score(y_test_enc, y_pred_ens_enc)
```

17. Visualisasi Kedua Model

```
# grafik
models = ['SVM', 'Logistic Regression']
akurasi = [acc_svm, acc_lr]
f1_scores = [f1_svm, f1_lr]

x = range(len(models))
width = 0.35

plt.figure()
bars1 = plt.bar(x, akurasi, width, label='Akurasi')
bars2 = plt.bar([i + width for i in x], f1_scores, width, label='F1-Score')

plt.xlabel('Model')
plt.ylabel('Nilai')
plt.title('Akurasi dan F1-Score SVM vs Logistic Regression')
plt.xticks([i + width/2 for i in x], models)
plt.legend()
plt.ylim(0, 1.0)

# angka di atas bar
for bar in bars1:
    plt.text(bar.get_x() + bar.get_width()/2, bar.get_height(),
             f"{bar.get_height():.2f}", ha='center', va='bottom')

for bar in bars2:
    plt.text(bar.get_x() + bar.get_width()/2, bar.get_height(),
             f"{bar.get_height():.2f}", ha='center', va='bottom')

plt.show()
```

18. LoA Jurnal



SURAT PENERIMAAN No. 1000/INVENTA/CV/SIM/2026

Kepada Yth.
Penulis:
Athary Zikry, Farid Akhar Siragar.

Di:
¹² Universitas Muhammadiyah Sumatera Utara

Dengan hormat,
Redaksi *Inventa: Journal of Science, Technology, and Innovation* (ISSN: 3123-3147 (Print) dan 3123-3155 (Electronic)), menyampaikan bahwa artikel Bapak/Ibu dengan judul:

Performance Analysis of Logistic Regression and SVM (Support Vector Machine) Algorithms on E-Football Mobile Game Review Sentiment

Telah diterima (ACCEPTED) setelah melalui proses review dan dinyatakan layak untuk diterbitkan pada Vol. 1 No. 3 (April 2026) *Inventa: Journal of Science, Technology, and Innovation*. Kami menyampaikan apresiasi yang sebesar-besarnya atas kontribusi ilmiah serta kepercayaan Bapak/Ibu kepada *Inventa: Journal of Science, Technology, and Innovation*.

Kami menghargai kontribusi ilmiah Bapak/Ibu dan berharap kerja sama ini dapat berlanjut pada penerbitan edisi berikutnya.

Semarang, 9 Mei 2026
Hormat kami,
Direktur CV, Scripta Intelektual Mandiri



Deo Renaldi Saputra, S.S., S.H., M.H.

19. Hasil Turnitin

UNIVERSITY REPORT			
15%	14%	6%	9%
SIMILARITY INDEX	INTERNET SOURCES	PUBLICATIONS	STUDENT PAPERS
PRIMARY SOURCES			
1	repository.umsu.ac.id Internet Source		9%
2	repositori.umrah.ac.id Internet Source		1%
3	repositori.uma.ac.id Internet Source		<1%
4	eprints.itn.ac.id Internet Source		<1%
5	jurnal.soratekno.co.id Internet Source		<1%
6	publikasi.dinus.ac.id Internet Source		<1%
7	ejournal.nusamandiri.ac.id Internet Source		<1%
8	Submitted to Sogang University Student Paper		<1%
9	repository.bakrie.ac.id Internet Source		<1%
10	repository.uigm.ac.id Internet Source		<1%
Submitted to STKIP Sumatera Barat			
Secondary Sources			
11	Student Paper		<1%
12	Submitted to Tabor College Student Paper		<1%
13	ejournal.warunayama.org Internet Source		<1%
14	journaleng.unifa.ac.id Internet Source		<1%
15	Submitted to Sriwijaya University Student Paper		<1%
16	Submitted to Universitas Muhammadiyah Sumatera Utara Student Paper		<1%
17	Submitted to Universitas Sumatera Utara Student Paper		<1%
18	openjournal.unpam.ac.id Internet Source		<1%
19	Submitted to University of Bolton Student Paper		<1%