BAB II
LANDASAN TEORI

2.1 Penelitian Terdahulu

Tabel 2. 1 Tabel Penelitian Terdahulu

Penulis	Judul	Jurnal	Metode	Hasil
Cahyo Prakoso, Arief Hermawan [18]	Perbandingan Model *Machine learning* dalam Analisis Sentimen Ulasan Pengunjung Keraton Yogyakarta pada *Google Maps* (2023)	KLIK: Kajian Ilmiah Informatika dan Komputer, Vol. 4, No. 3, 2023	SVM, *Logistic Regression, Naive Bayes*	SVM (87,12%), *Logistic Regression* (85,61%), *Naive Bayes* (80,23)
Satya Abdul Halim Bahtiar, Chandra Kusuma Dewa, Ahmad Luthfi [8]	*Comparison of Naïve Bayes and Logistic Regression in Sentiment Analysis on Marketplace Reviews Using Rating-Based Labeling* (2023)	*Journal of Information Systems and Informatics, Vol. 5, No. 3, September 2023*	*Naive Bayes, Logistic Regression*	*Logistic Regression* mencapai akurasi tertinggi di angka 84.58%
Muhammad Riski Prasetyo, Achmad Fahrurozi [9]	Analisa Sentimen Pada Ulasan Google Untuk Hotel Gran Mahakam Jakarta Menggunakan Pendekatan *Machine learning* (2023)	Jurnal Ilmiah Informatika Komputer, Vol. 28, No. 3, 2023	SVM, *Naive Bayes*	SVM mencapai akurasi 92%, sementara *Naive Bayes* mencapai 90%

Penulis	Judul	Jurnal	Metode	Hasil
Jonathan Arya Dhamma, Simon Prananta Barus. [19]	*Sentiment Analysis on Google Reviews Using NaÃ¯ve Bayes, K-Nearest Neighbors, and Logistic Regression to Improve Novotel Services (2025)*	*Journal of Artificial Intelligence and Computing (JAIC), Vol. 9, No. 1, 2025*	*Naïve Bayes, K-Nearest Neighbors, Logistic Regression*	*Logistic Regression* memiliki akurasi tertinggi sebesar 94.54% ketika menggunakan *unigrams* (n = 1) dalam analisis sentimen ulasan *Google* untuk hotel Novote
Lutfi Budi Ilmawan Muhammad Aliyazid Mude [20].	Perbandingan Metode Klasifikasi *Support Vector Machine* dan *Naïve Bayes* untuk Analisis Sentimen pada Ulasan Tekstual di *Google Play Store* (2020)	Jurnal ILKOM, Vol. 12, No. 2, 2020	*Support Vector Machine, Naïve Bayes.*	Jurnal tersebut menunjukkan bahwa *SVM* mencapai akurasi 81.46%, sedangkan *Naive Bayes* mencapai akurasi 75.41% dalam analisis sentimen ulasan di *Google Play Store.*
Putri Marceliana Aryanto, Rodhiyah Mardhiyyah [21]	Analisis Sentimen Terhadap *Review Google Maps* Jogja City Mall Menggunakan Algoritma *Support Vector Machine (2024)*	Journal of Computer System and Informatics (JoSYC), Vol. 6, No. 4, 2024	*Support Vector Machine*	*SVM* mencapai akurasi 84% dalam mengklasifikasi sentimen ulasan *Google Maps* tentang *mall*. Penelitian juga menggunakan 1.694 data ulasan dan menerapkan *K-Fold validation* untuk memvalidasi model.

Penulis	Judul	Jurnal	Metode	Hasil
Muhammad Farid, Muhammad Nizam, Desi Pratiwi, Ahmad Maulana [22]	Komparasi Metode Klasifikasi Dalam Analisis Sentimen Ulasan Pengguna Aplikasi KRL Access Di Google Play Store (2024)	*Journal of Computer and Omputics (JCO), Vol. 4, No. 1, 2024*	*Naïve Bayes, Random Forest, Logistic Regression, Support Vector Machine (SVM), K-Nearest Neighbor.*	Random Forest dan SVM memberikan akurasi terbaik, dengan Random Forest mencatat akurasi hingga 91,38% pada beberapa *dataset*.
Sri Lestanti, Saiful Nur Budiman, Erwan [23].	Analisis Sentimen Berdasarkan Hasil *Review* Lokasi Google Map Menggunakan Natural Language Toolkit TextBlob dan Naïve Bayes (2025)	Jurnal Ahli Muda Indonesia (JAMI), Vol. 5, No. 2, 2025	*Natural Language Toolkit TextBlob dan Naïve Bayes*	Dengan *dataset* 253 data latih dan 64 data uji, model yang dihasilkan mencapai akurasi 94%. Hasil ini menunjukkan efektivitas metode dalam menganalisis sentimen ulasan dapat dijadikan referensi untuk analisis serupa pada ulasan lokasi lain.

Penulis	Judul	Jurnal	Metode	Hasil
I Made Surya Ananta Wijaya, I Gusti Agung Gede Arya Kadyanan, I Made Widiartha [24].	Penerapan Metode *Content Based Filtering* dan *K-Nearest Neighbor* Dalam Sistem Rekomendasi Musik (2022)	Jurnal Ilmu Komputer (JIKA), Vol. 9, No. 2, 2022	*Content-Based Filtering dan K-Nearest Neighbors (KNN).*	Sistem rekomendasi mencapai akurasi 90,49% dengan parameter k=9. Kombinasi TF-IDF (untuk ekstraksi fitur) dan KNN efektif mencocokkan preferensi pengguna.
Muhammad Fikri, Rizki Amalia, Eko Prasetyo [25].	Analisis Sentimen Ulasan Aplikasi *Wattpad* di *Google Play* Menggunakan Metode *Random Forest* (2021)	Jurnal Ilmiah Informatika dan Komputer (JIIK), Vol. 6, No. 2, 2021	*Random Forest*	*Random Forest* mampu mengklasifikasikan sentimen ulasan aplikasi dengan akurasi yang baik, mencapai lebih dari 90% pada beberapa pengujian. Akurasi klasifikasi mencapai 90,2% dengan *precision* 89,5% untuk sentimen positif. Ulasan negatif dominan terkait *bug* aplikasi dan iklan intrusif.

Tabel 2.1 merupakan tabel penelitian didasari oleh berbagai jurnal terdahulu yang membahas aplikasi analisis sentimen dan rekomendasi berbasis konten pada ulasan *online*, terutama yang menggunakan *platform Google Maps* atau *platform* lain yang serupa. Jurnal-jurnal tersebut memberikan kontribusi signifikan dalam membandingkan berbagai algoritma *machine learning* untuk analisis sentimen, serta metode yang digunakan dalam rekomendasi berbasis konten.

Penelitian membandingkan tiga model *machine learning*, yaitu *Support Vector Machine* (SVM), *Logistic Regression*, dan *Naive Bayes* dalam analisis sentimen ulasan pengunjung di *Google Maps* untuk Keraton Yogyakarta. Hasil penelitian menunjukkan bahwa SVM memiliki akurasi tertinggi di antara ketiganya [18]. Penelitian menjadi penting karena menunjukkan efektivitas SVM, yang juga digunakan dalam penelitian untuk menganalisis sentimen ulasan pengunjung Taman Safari Indonesia.

Penelitian lain yang melakukan perbandingan antara *Naive Bayes* dan *Logistic Regression* dalam menganalisis sentimen ulasan di *marketplace*. Hasilnya menunjukkan bahwa *Logistic Regression* memberikan akurasi tertinggi [8]. Hal ini relevan dengan penelitian yang juga menggunakan kedua algoritma tersebut, memberikan dasar untuk memilih model yang paling efektif dalam mengklasifikasikan sentimen ulasan *Google Maps*.

Penelitian yang juga melakukan analisis sentimen pada ulasan Google untuk Hotel Gran Mahakam Jakarta menggunakan SVM dan *Naive Bayes*. Hasil penelitian menunjukkan akurasi yang sangat tinggi, dengan SVM mencapai 92% dan *Naive Bayes* mencapai 90% [9]. Dengan menggunakan hasil ini, penelitian dapat memvalidasi efektivitas SVM untuk analisis sentimen pada ulasan tempat wisata, memberikan wawasan penting mengenai penggunaan model dalam yang serupa.

Penelitian lain yang membandingkan *Naive Bayes*, *K-Nearest Neighbors* (KNN), dan *Logistic Regression* dalam analisis sentimen untuk ulasan Novotel. Penelitian menemukan bahwa *Logistic Regression* memiliki akurasi tertinggi.

Walaupun penelitian menggunakan KNN dalam perbandingan, hasilnya tetap memberikan panduan yang berguna dalam memilih algoritma untuk penelitian [19].

Penelitian yang melakukan perbandingan antara SVM dan *Naïve Bayes* untuk analisis sentimen ulasan *Google Play Store*. Meskipun SVM menunjukkan keunggulan dalam hal akurasi, penelitian memberikan perspektif tambahan terkait penggunaan *Naive Bayes* dalam analisis sentimen berbasis teks. Penelitian mendukung pemilihan algoritma yang relevan dalam penelitian [20].

Penelitian lain yang juga menggunakan SVM untuk menganalisis sentimen ulasan *Google Maps* terkait Jogja City *Mall*. Dengan menggunakan *K-Fold validation*, penelitian menemukan akurasi 84% untuk SVM [21]. Hasil ini membuktikan bahwa efektivitas SVM dalam menganalisis ulasan berbasis *Google Maps*, yang juga digunakan dalam penelitian untuk menganalisis ulasan pengunjung di Taman Safari Indonesia.

Penelitian Muhammad Farid, Muhammad Nizam, Desi Pratiwi, dan Ahmad Maulana (2022) membandingkan berbagai metode klasifikasi, seperti *Naïve Bayes, Random Forest, Logistic Regression, Support Vector Machine* (SVM), dan *K-Nearest Neigh*bor, dalam analisis sentimen ulasan. Hasil penelitian menunjukkan bahwa *Random Forest* dan SVM memberikan performa akurasi terbaik, dengan *Random Forest* mencapai akurasi hingga 91,38% pada beberapa *dataset*. Penelitian memberikan panduan bagi pemilihan metode klasifikasi yang optimal dalam analisis sentimen, khususnya dengan menyoroti keunggulan *Random Forest* dan SVM dalam mengolah data ulasan berbasis teks.

Penelitian Sri Lestanti, Saiful Nur Budiman, dan Erwan (2023) memanfaatkan *TextBlob* dan *Naive Bayes* untuk analisis sentimen pada ulasan *Google Maps*. Dengan mencapai akurasi 94%, penelitian memberikan contoh lain tentang efisiensi *Naive Bayes* dan *TextBlob* dalam menganalisis ulasan berbasis teks, yang dapat digunakan dalam penelitian untuk memperkaya model analisis sentimen yang diterapkan pada ulasan pengunjung Taman Safari Indonesia.

Penelitian I Made Surya Ananta Wijaya, I Gusti Agung Gede Arya Kadyanan, dan I Made Widiartha (2022) menerapkan metode *Content Based*

Filtering dan *K-Nearest Neighbor* (KNN) dalam rekomendasi musik. Hasil penelitian menunjukkan bahwa kombinasi *Content Based Filtering* dan KNN, dengan ekstraksi fitur menggunakan TF-IDF, mampu menghasilkan rekomendasi musik yang sesuai dengan preferensi pengguna dan mencapai akurasi sebesar 90,49% pada parameter k=9. Hal ini menegaskan bahwa integrasi metode *Content Based Filtering* dan KNN efektif dalam meningkatkan akurasi rekomendasi berbasis konten, sehingga dapat menjadi acuan dalam pengembangan rekomendasi musik di masa depan.

Penelitian yang juga menerapkan metode *Random Forest* untuk analisis sentimen pada ulasan aplikasi *Wattpad* di *Google Play*. Hasil penelitian menunjukkan bahwa *Random Forest* mampu mengklasifikasikan sentimen ulasan aplikasi dengan akurasi yang tinggi, yakni mencapai 90,2%, dengan *precision* sebesar 89,5% untuk sentimen positif [22]. Selain itu, penelitian ini juga menemukan bahwa ulasan negatif umumnya didominasi oleh keluhan terkait *bug* aplikasi dan iklan yang mengganggu. Hal ini menegaskan efektivitas *Random Forest* dalam mengelola data ulasan pengguna dan memberikan gambaran mengenai persepsi pengguna terhadap aplikasi berbasis teks.

Jurnal-jurnal yang dipilih dalam memberikan kontribusi yang signifikan terhadap pendekatan yang digunakan dalam analisis sentimen dan rekomendasi berbasis ulasan. Semua jurnal tersebut menggunakan berbagai algoritma *machine learning* (SVM, *Logistic Regression*, *Naive Bayes*, *K-Nearest Neighbor* (KNN) dan *Random Forest*) yang relevan untuk mengklasifikasikan sentimen pada ulasan berbasis teks, serta beberapa menerapkan content-based filtering untuk rekomendasi.

Pemilihan jurnal ini didasarkan pada kesesuaian dengan algoritma yang digunakan dalam penelitian, serta kontribusi dari penelitian sebelumnya yang dapat membantu dalam meningkatkan akurasi dan efektivitas model yang digunakan. Jurnal-jurnal ini juga memberikan perspektif tambahan mengenai teknik-teknik validasi, seperti *K-Fold validation* dan optimasi model dengan teknik seperti PSO,

yang dapat digunakan untuk meningkatkan kualitas hasil analisis dan rekomendasi dalam penelitian.

Terdapat keterkaitan yang signifikan antara penelitian ini dan penelitian terdahulu yang membahas analisis sentimen dan rekomendasi berbasis konten. Semua jurnal yang dianalisis menggunakan algoritma *machine learning* relevan, seperti SVM, *Logistic Regression, Naive Bayes, K-Nearest Neighbors* (KNN), dan *Random Forest*. Penelitian yang menunjukkan efektivitas SVM [18], yang juga diaplikasikan dalam penelitian ini untuk menganalisis sentimen pengunjung Taman Safari Indonesia.

Keterkaitan lainnya yaitu hasil dari penelitian Satya Abdul Halim Bahtiar dan rekan-rekan memperkuat pemilihan *Naive Bayes* dan *Logistic Regression* yang juga diterapkan di sini. Sementara penelitian Muhammad Riski Prasetyo menunjukkan akurasi tinggi SVM dan *Naive Bayes*, memberikan validasi tambahan untuk penggunaan algoritma tersebut. Penerapan metode content-based filtering dalam studi I Made Surya Ananta Wijaya dan tim juga menambah dimensi penting bagi pengembangan rekomendasi dalam penelitian ini. Secara keseluruhan, keterkaitan ini memperkuat argumen untuk algoritma yang dipilih dan memberikan konteks praktis dalam meningkatkan analisis dan rekomendasi yang dihasilkan.

Dengan menggunakan hasil dan metode yang dibahas dalam jurnal-jurnal terdahulu ini, penelitian dapat memberikan hasil yang lebih akurat dan relevan dalam mengoptimalkan analisis sentimen dan rekomendasi berbasis ulasan di Taman Safari Indonesia.

2.2 Teori Penelitian

2.2.1 Analisis Sentimen

Analisis sentimen adalah proses untuk mengidentifikasi dan mengevaluasi opini atau perasaan yang terkandung dalam teks. Di era digital, di mana banyak orang berbagi pengalaman mereka secara *online*, analisis sentimen menjadi alat yang penting untuk memahami bagaimana produk atau layanan diterima oleh publik [26]. Taman Safari Indonesia, analisis sentimen digunakan untuk menilai rasa puas atau tidak puas pengunjung melalui ulasan yang mereka buat di

Google Maps. Dengan mengklasifikasikan ulasan ini menjadi kategori positif, negatif, dan netral, pengelola dapat memperoleh pandangan yang lebih jelas tentang apa yang disukai atau tidak disukai pengunjung [27]. Melalui analisis ini, pengelola dapat mengetahui masalah yang harus diperbaiki serta aspek-aspek yang perlu dipertahankan atau ditingkatkan. Misalnya, jika banyak ulasan mengandung komentar positif tentang interaksi pengunjung dengan hewan, manajemen dapat mempertahankan program tersebut dan bahkan memperluasnya. Sebaliknya, jika terdapat banyak keluhan tentang kebersihan, hal ini bisa menjadi fokus perbaikan di masa mendatang .

2.2.2 Taman Safari Indonesia

Taman Safari Indonesia merupakan salah satu destinasi wisata yang terkenal di Indonesia. Terletak di daerah Cisarua, Bogor, taman ini menampilkan berbagai jenis hewan dari seluruh dunia dalam lingkungan yang menyerupai habitat aslinya. Taman Safari dirancang untuk memberikan pengalaman interaktif, di mana pengunjung dapat melihat hewan-hewan secara langsung dan belajar lebih banyak tentang konservasi satwa [28]. Taman ini juga menawarkan atraksi seperti pertunjukan hewan dan area bermain, yang membuatnya menjadi tujuan favorit bagi keluarga. Dengan banyaknya pengunjung yang datang setiap tahun, pengelolaan Taman Safari Indonesia sangat bergantung pada umpan balik dari pengunjung. Ulasan yang ditinggalkan di *platform* seperti *Google Maps* menjadi sangat penting karena dapat memberikan data yang berharga tentang persepsi dan pengalaman pengunjung terhadap layanan, fasilitas, dan keseluruhan pengalaman mereka di taman [29].

2.2.3 Ulasan Google Maps

Google Maps merupakan *platform* yang memungkinkan pengguna untuk memberikan umpan balik dan berbagi pengalaman mengenai tempat yang telah mereka kunjungi [30]. Ulasan yang ditinggalkan di *Google Maps* mengenai Taman Safari Indonesia tidak hanya mencerminkan tingkat kepuasan pengunjung, tetapi juga memberikan petunjuk tentang area mana yang perlu diperbaiki. Data ini sangat penting karena dapat mempengaruhi keputusan pengunjung lain yang sedang mempertimbangkan untuk mengunjungi taman.

Ulasan di *Google Maps* mencakup berbagai informasi, mulai dari kualitas layanan, kebersihan, hingga interaksi yang dilakukan selama kunjungan [31]. Dengan lebih dari ribuan ulasan yang dikumpulkan setiap tahun, pengelola dapat menganalisis pola dan tren dalam umpan balik pengunjung.

2.2.4 Term Frequency

Term Frequency adalah ukuran yang digunakan dalam analisis teks untuk menentukan seberapa sering suatu kata muncul dalam sebuah dokumen. Pengukuran ini dihitung dengan cara membagi jumlah kemunculan kata yang dimaksud dengan total jumlah kata yang ada dalam dokumen tersebut. Dengan demikian, *Term Frequency* memberikan nilai yang relatif dan memungkinkan perbandingan antara kata-kata dalam dokumen yang memiliki panjang berbeda. Nilai *Term Frequency* yang tinggi menunjukkan bahwa kata tersebut mungkin memiliki relevansi yang lebih besar dalam konteks dokumen tersebut, sehingga dapat membantu dalam memahami fokus utama dari informasi yang disajikan.

Penggunaan *Term Frequency* sangat berguna dalam konteks penambangan teks dan pengolahan informasi. Misalnya, dalam aplikasi pencarian informasi atau sistem rekomendasi, *Term Frequency* dapat membantu mengidentifikasi kata kunci utama yang sering muncul dan relevan bagi pengguna. Namun, meskipun *Term Frequency* memberikan wawasan yang penting tentang frekuensi kemunculan kata, pendekatan ini memiliki keterbatasan karena tidak mempertimbangkan seberapa umum suatu kata muncul dalam keseluruhan koleksi dokumen. Hal ini bisa menyebabkan kata-kata yang sangat umum tetapi kurang bermanfaat memberikan berat yang lebih dalam analisis.

2.2.5 Penggunaan Data Ulasan untuk Rekomendasi dan Analisis Sentimen

Rekomendasi berbasis ulasan dan analisis sentimen dapat memberikan nilai lebih dalam pemahaman pengalaman pengunjung [32]. Rekomendasi pada dasarnya digunakan untuk menyarankan item atau atraksi berdasarkan preferensi pengguna yang terkandung dalam ulasan mereka [33]. *Term Frequency* menjadi salah satu pendekatan utama dalam rekomendasi. *Term*

Frequency berfokus pada fitur dari konten yang ada, seperti kata-kata atau frase yang sering muncul dalam ulasan. Dengan menganalisis ulasan pengunjung di *Google Maps*, sistem ini dapat menyarankan atraksi atau program di Taman Safari Indonesia yang mirip dengan yang sudah disukai oleh pengunjung lain [34].

Rekomendasi berbasis konten bekerja dengan cara menganalisis teks ulasan dan mengekstrak informasi penting seperti tema atau topik yang sering dibicarakan, untuk menemukan kesamaan antara ulasan dan kemudian memberikan rekomendasi yang relevan [35]. Model *machine learning* seperti SVM dan *Naive Bayes* digunakan untuk klasifikasi sentimen pada ulasan yang kemudian digunakan untuk mengklasifikasikan dan memberi skor pada ulasan tersebut, baik dalam kategori positif, negatif, atau netral [36]. Data ini kemudian dimanfaatkan untuk menyarankan pengalaman wisata atau atraksi yang relevan.

Lebih lanjut, penggabungan antara analisis sentimen dan rekomendasi berbasis ulasan memberikan keuntungan ganda. Pengelola Taman Safari Indonesia tidak hanya mendapatkan wawasan lebih dalam tentang kepuasan pengunjung melalui analisis sentimen, tetapi juga dapat memberikan rekomendasi yang lebih personal dan sesuai dengan preferensi pengunjung berdasarkan sentimen yang terdeteksi. Misalnya, jika seseorang memiliki pengalaman positif dengan interaksi dengan hewan, maka rekomendasi akan menyarankan atraksi lain yang memiliki tema serupa, meningkatkan pengalaman pengunjung dan kepuasan secara keseluruhan.

Sebagai tambahan, penggunaan *deep learning* dengan algoritma *seperti K-Nearest Neighbor* (KNN) dan *Random Forest* memungkinkan untuk menggali fitur-fitur yang lebih kompleks dalam teks, seperti hubungan semantik antara kata-kata dalam ulasan yang lebih panjang [37]. Ini memberikan keuntungan lebih besar dalam analisis sentimen yang lebih mendalam dan rekomendasi yang lebih canggih dan relevan bagi pengunjung.

2.2.6 Tujuan dan Manfaat Analisis Sentimen

Tujuan utama dari analisis sentimen adalah untuk memberikan wawasan yang lebih dalam kepada manajemen Taman Safari Indonesia mengenai respons pengunjung terhadap layanan dan atraksi yang disediakan [38]. Dengan mengidentifikasi sentimen pengunjung melalui analisis ulasan, manajemen dapat mengetahui apa yang perlu ditingkatkan dan area mana yang sudah memenuhi ekspektasi pengunjung. Manfaat lain dari analisis ini termasuk penyediaan rekomendasi yang lebih baik kepada pengunjung berdasarkan data ulasan, serta pengembangan strategi pemasaran yang lebih efektif [39]. Dengan menggunakan pengetahuan yang dihasilkan dari analisis ini, diperkirakan Taman Safari Indonesia dapat meningkatkan kualitas layanan dan pengalaman pengunjung secara keseluruhan, sehingga dapat meningkatkan kepuasan dan loyalitas pengunjung.

2.3 Framework dan Algoritma Penelitian

2.3.1 Knowledge Discovery in Databases (KDD)

KDD (*Knowledge Discovery in Databases*) adalah proses yang digunakan untuk mengekstrak pengetahuan yang bermanfaat dari data yang besar dan kompleks [40]. Proses ini terdiri dari beberapa tahapan yang terstruktur, mulai dari pemilihan data hingga evaluasi hasil yang diperoleh. KDD digunakan dalam berbagai domain untuk menemukan pola yang berguna, termasuk di bidang bisnis, kesehatan, dan ilmu sosial. Proses KDD dapat digambarkan dalam lima tahap utama yang saling terkait, yaitu *Selection*, *Preprocessing*, *Transformation*, *Data Mining*, dan *Evaluation*.

a) Selection

Pada tahap ini, data yang relevan untuk analisis dipilih dari berbagai sumber. Data yang tidak relevan atau yang tidak sesuai dengan tujuan analisis akan disaring untuk meningkatkan efisiensi dan relevansi proses selanjutnya. Pemilihan data ini sangat penting untuk memastikan bahwa hanya data yang memiliki kualitas dan nilai yang digunakan dalam analisis.

b) Preprocessing

Data yang dikumpulkan sering kali tidak lengkap, tidak konsisten, atau memiliki kesalahan. Oleh karena itu, tahap *preprocessing* diperlukan untuk membersihkan dan mempersiapkan data agar siap untuk dianalisis [41]. Proses *preprocessing* meliputi beberapa langkah:

a. Penghapusan Duplikat: Menghapus data yang ganda.
b. Penanganan Data yang Hilang: Mengisi data yang hilang dengan teknik imputasi atau menghapusnya jika tidak memungkinkan untuk diisi.
c. Normalisasi atau Standarisasi Data: Mengubah data ke dalam skala yang konsisten agar dapat dibandingkan secara adil.
d. Tokenisasi dan *Stemming* (untuk data teks): Memecah teks menjadi unit-unit terkecil (kata) dan mengurangi kata ke bentuk dasarnya untuk mengurangi variasi kata.

Tujuan utama dari tahap *preprocessing* adalah untuk memastikan bahwa data dalam kondisi yang baik dan siap untuk diproses lebih lanjut oleh algoritma [42].

c) Transformation

Setelah data dibersihkan, tahap berikutnya adalah transformasi. Pada tahap ini, data yang telah diproses akan diubah menjadi format atau representasi yang lebih berguna untuk analisis lebih lanjut. Transformasi ini dapat meliputi:

a. *Feature Engineering*: Pembuatan fitur baru yang dapat membantu dalam proses analisis, seperti ekstraksi kata-kata kunci dari teks atau agregasi data.
b. Reduksi Dimensi: Mengurangi jumlah fitur dalam *dataset* untuk memfokuskan analisis pada fitur-fitur yang paling penting.
c. Normalisasi: Untuk memastikan data dalam skala yang sama agar tidak terjadi ketidakseimbangan dalam analisis.

Transformasi ini penting agar data dapat dianalisis secara efisien oleh algoritma *machine learning* atau teknik statistik lainnya.

d) Data Mining

Data Mining adalah tahap inti dalam proses KDD, di mana teknik-teknik analisis digunakan untuk menemukan pola-pola yang berguna dalam data [43]. Pada tahap ini, algoritma *machine learning* atau teknik statistik diterapkan untuk mengekstraksi pengetahuan dari data yang telah diproses. Beberapa teknik yang digunakan dalam Data Mining antara lain:

a. Klasifikasi: Mengelompokkan data ke dalam kategori atau kelas tertentu berdasarkan fitur-fitur yang ada.
b. *Clustering*: Mengelompokkan data yang serupa ke dalam kelompok-kelompok berdasarkan kesamaan fitur.
c. Asosiasi: Mencari hubungan atau pola antar item dalam data (misalnya dalam analisis pasar).
d. Regresi: Memprediksi nilai kontinu berdasarkan data yang ada.

Tahap ini bertujuan untuk mengidentifikasi pola atau informasi baru yang sebelumnya tidak terlihat dalam data.

e) Evaluation

Evaluasi adalah tahap terakhir dalam proses KDD yang digunakan untuk menilai seberapa baik hasil yang diperoleh pada tahap Data Mining [44]. Model atau pola yang ditemukan dievaluasi menggunakan berbagai metrik untuk memastikan kualitas dan akurasi pengetahuan yang diekstraksi. Beberapa metrik evaluasi yang umum digunakan adalah:

a. Akurasi: Mengukur seberapa banyak prediksi yang benar dibandingkan dengan total prediksi yang dilakukan.
b. *Precision* dan *Recall*: *Precision* mengukur proporsi prediksi yang benar di antara semua prediksi positif, sementara *recall*

mengukur proporsi prediksi yang benar di antara semua data yang sebenarnya positif.

c. F1-Score: Merupakan rata-rata harmonis antara *precision* dan *recall*, memberikan gambaran yang lebih baik mengenai keseimbangan antara keduanya.
d. ROC-AUC (*Receiver Operating Characteristic – Area Under Curve*): Digunakan untuk mengukur kinerja klasifikasi biner.

Evaluasi ini penting untuk memastikan bahwa pengetahuan yang ditemukan dapat digunakan secara praktikal dan efektif, serta untuk mengetahui apakah model atau pola yang ditemukan dapat digeneralisasi pada data yang belum terlihat.

2.3.2 Support Vector Machine (SVM)

Support Vector Machine (SVM) adalah algoritma klasifikasi yang berfungsi untuk mengelompokkan data ke dalam dua kelas yang berbeda dengan sangat efektif. Algoritma ini bekerja dengan cara menemukan batas optimal yang memisahkan data berdasarkan fitur yang telah diekstraksi dari teks. Dalam analisis sentimen, SVM digunakan untuk mengklasifikasikan ulasan sebagai positif atau negatif.[45] SVM memiliki kemampuan yang sangat baik dalam menangani data yang kompleks dan besar, serta dapat meminimalkan kesalahan klasifikasi dengan menjaga jarak maksimum antara data dari dua kelas. Pendekatan ini menjadikan SVM sebagai pilihan yang baik untuk aplikasi analisis sentimen, di mana hasilnya dapat membantu pengelola untuk memahami persepsi pengunjung terhadap layanan yang diberikan [46]. Rumus untuk SVM dapat dinyatakan sebagai:

$$w \cdot x + b = 0 \tag{2.1}$$

Rumus 2. 1 Rumus *Support Vector Machine*

1. w adalah vektor bobot yang menentukan orientasi pemisah,
2. x adalah vektor fitur dari data,
3. b adalah bias yang menentukan posisi pemisah.

Fungsi tujuan untuk SVM diungkapkan dengan rumus:

$$\min \frac{1}{2} \| w \|^2 \tag{2.2}$$

Rumus 2. 2 Rumus Fungsi Tujuan untuk *Support Vector Machine*

Dengan kendala:

$$y_i(w \cdot x_i + b) \geq 1 \; untuk \; i = 1,2, \dots, n \tag{2.3}$$

Rumus 2. 3 Rumus dengan Kendala *Support Vector Machine*

Dalam rumus 2.1, persamaan ini menggambarkan garis pemisah antara dua kelas. Vektor bobot *w* berfungsi untuk menunjukkan arah dan kemiringan dari pemisah, sedangkan bias *b* membantu dalam menggeser garis pemisah sesuai lokasi data. Fungsi tujuan yang dinyatakan dengan rumus 2.2 berusaha untuk mempertahankan bobot sekecil mungkin, yang secara tidak langsung memaksimalkan margin antara kedua kelas. Kendala rumus 2.3 memastikan bahwa semua titik data dari kelas positif berada di satu sisi pemisah dan semua titik dari kelas negatif berada di sisi lainnya.

2.3.3 Logistic Regression

Logistic Regression adalah metode statistik yang digunakan sebagai model klasifikasi biner untuk memprediksi probabilitas suatu kejadian berdasarkan satu atau lebih variabel *input*. Dalam analisis sentimen, model ini berfungsi untuk menentukan sifat dari ulasan pengunjung, apakah bersifat positif atau negatif. *Logistic Regression* mengubah *output* yang bersifat linier menjadi hasil probabilitas dengan menggunakan fungsi logit [47]. Metode ini sangat bermanfaat dalam menghadapi masalah klasifikasi

di mana hasil yang diperkirakan hanya memiliki dua kelas, menjadikannya alat yang efektif untuk memahami dan mengelola sentimen dalam ulasan. Fungsi logit dari *Logistic Regression* dirumuskan sebagai:

$$P(y = 1 \mid x) = \frac{1}{1 + e^{-(w \cdot x + b)}} \tag{2.4}$$

Rumus 2. 4 Rumus *Logistic Regression*

1. $P(y = 1 \mid x)$ adalah probabilitas bahwa *output y* adalah 1 (positif) berdasarkan variabel input *x*.
2. *w* adalah vektor bobot yang menunjukkan pengaruh masing-masing fitur.
3. *b* adalah bias yang menggeser fungsi logit.
4. *e* adalah bilangan *Euler*, yang merupakan basis dari logaritma natural.

Dalam rumus 2.4 fungsi ini mengubah kombinasi linier dari variabel *input* $w \cdot x + b = 0$ menjadi probabilitas yang berkisar antara 0 dan 1. Istilah $w \cdot x$ menggambarkan hubungan linier antara variabel *input* dan *output*, di mana *w* adalah bobot yang diperoleh selama proses pelatihan model. Bias *b* memungkinkan model untuk lebih fleksibel dalam penyesuaian *output*. Dengan probabilitas yang dihasilkan, pengelompokan dilakukan dengan menetapkan batas, biasanya 0,5, yang menentukan apakah sebuah ulasan masuk ke dalam kategori positif atau negatif. *Logistic Regression*, dengan kemampuannya untuk memberikan *output* probabilistik.

2.3.4 Naïve Bayes

Naïve Bayes adalah model klasifikasi berbasis probabilitas yang sangat efisien dan sering digunakan dalam analisis teks [48]. Meskipun asumsi independensi ini tidak selalu mencerminkan keadaan nyata, *Naïve Bayes* tetap terbukti efektif dalam mengklasifikasikan data teks, terutama dalam analisis sentimen. Dalam analisis sentimen, model ini bertugas untuk mengestimasi probabilitas dari setiap kelas sentimen, apakah itu

positif, negatif, atau netral, berdasarkan kata-kata yang muncul dalam ulasan. Teknik ini memanfaatkan *Teorema Bayes* untuk menghitung kemungkinan setiap kelas dengan menggunakan frekuensi kemunculan kata-kata tertentu dalam data pelatihan, memungkinkan model untuk membuat prediksi yang akurat dengan cepat.

Teorema Bayes dapat dinyatakan dengan rumus:

$$P(C_k \mid x) = \frac{P(x \mid C_k) \cdot P(C_k)}{P(x)} \quad (2.5)$$

Rumus 2. 5 Rumus *Naive Bayes*

1. $P(C_k \mid x)$ adalah probabilitas kelas C_k diberikan fitur x
2. $P(x \mid C_k)$ adalah probabilitas fitur x terjadi dalam kelas C_k
3. $P(C_k)$ adalah probabilitas awal dari kelas C_k
4. $P(x)$ adalah probabilitas total dari fitur x

Dalam rumus 2.5 model ini menghitung probabilitas kelas C_k berdasarkan fitur x dengan memanfaatkan informasi yang diberikan oleh *Teorema Bayes*. $P(x \mid C_k)$ mewakili probabilitas bahwa fitur x muncul ketika kelas C_k adalah benar, yang dihitung dengan menghitung frekuensi dari kata-kata yang terkait dalam ulasan. $P(C_k)$ adalah proporsi kelas C_k dalam keseluruhan *dataset*, memberikan awal yang penting dalam klasifikasi. Karena model *Naïve Bayes* membuat asumsi independensi antar fitur-fitur tersebut, perhitungan probabilitas untuk fitur yang beragam dalam ulasan relatif sederhana dan cepat, sehingga memungkinkan prediksi yang efisien terhadap kategori sentimen.

2.3.5 K-Nearest Neighbors (KNN)

K-Nearest Neighbors (KNN) adalah salah satu algoritma pembelajaran mesin yang digunakan untuk klasifikasi dan regresi [49]. KNN adalah metode yang sangat sederhana namun efektif yang mengklasifikasikan data berdasarkan kedekatannya dengan data lainnya [50]**.** *K-Nearest Neighbors* (KNN) adalah algoritma pembelajaran mesin

sederhana yang digunakan untuk klasifikasi dan regresi. Prinsip dasar KNN adalah bahwa objek yang mirip cenderung terletak berdekatan satu sama lain dalam ruang fitur. Algoritma ini bekerja dengan mencari K*K* tetangga terdekat dari data baru dan menentukan klasifikasi atau nilai berdasarkan tetangga tersebut.

$$d(A,B) = \sqrt{\sum_{i=1}^{N} \left(x_i^{(B)} - x_i^{(A)}\right)^2} \quad (2.6)$$

Rumus 2. 6 Rumus *K-Nearest Neighbors*

1. $d(A,B)$: Jarak antara titik A dan B.
2. $x_i^{(A)}$: Nilai fitur ke-i dari titik A.
3. $x_i^{(B)}$: Nilai fitur ke-i dari titik B.
4. N: Jumlah fitur dalam dataset.
5. $\sqrt{\sum_{i=1}^{N} \left(x_i^{(B)} - x_i^{(A)}\right)^2}$: Menyatakan jarak antara dua titik dalam ruang N-dimensi.
6. $x_i^{(B)} - x_i^{(A)}$: Menghitung selisih antara nilai-nilai fitur ke-i dari titik B dan A.
7. $(\dots)^2$: Menguadratkan selisih untuk memastikan jarak selalu positif dan untuk menekan perbedaan yang lebih besar.
8. $\sum$: Menjumlahkan semua kuadrat selisih untuk semua fitur (dari 1 hingga N).
9. $\sqrt{\ }$: Mengambil akar kuadrat total untuk mendapatkan jarak akhir.

Rumus 2.6 menggunakan jarak *Euclidean* untuk mengukur seberapa dekat dua titik dalam ruang fitur. Jarak antara titik A*A* dan titik B*B* dihitung dengan mengekspresikan selisih masing-masing nilai fitur sebagai $(x_i^{(B)} - x_i^{(A)})$ di mana $x_i^{(A)}$ dan $x_i^{(B)}$ merupakan nilai fitur ke-i dari titik A dan B, masing-masing. Selisih ini kemudian dikuadratkan untuk memastikan bahwa semua perbedaan memiliki nilai positif,

menghilangkan kemungkinan efek kontras negatif yang berpotensi mengganggu hasil. Setelah menguadratkan selisih untuk semua fitur, hasilnya dijumlahkan menggunakan simbol $\sum$. Langkah terakhir adalah mengambil akar kuadrat dari total ini, yang memberikan jarak Euclidean $d(A, B)$, memungkinkan kita untuk memahami seberapa dekat kedua titik tersebut. Dengan menghitung jarak ini untuk semua titik dalam dataset dan memilih K tetangga terdekat, KNN dapat menentukan kelas atau nilai yang paling relevan untuk titik baru berdasarkan mayoritas hasil dari tetangga terdekat tersebut.

2.3.6 Random Forest

Random Forest adalah algoritma *machine learning* yang digunakan untuk klasifikasi dan regresi, yang menggabungkan sejumlah pohon keputusan untuk memberikan hasil yang lebih akurat dan stabil. *Random Forest* berfungsi untuk mengklasifikasikan data teks ke dalam kategori, seperti positif, negatif, atau netral [51]. Metode ini bekerja dengan membuat banyak pohon keputusan dari *subset* data acak, yang kemudian memberikan suara untuk prediksi akhir. Fungsi logika dari *Random Forest* dapat dirumuskan sebagai berikut:

$$\hat{y} = \text{argmax}\left(\frac{1}{N}\sum_{i=1}^{N} \quad T_i(x)\right) \tag{2.7}$$

Rumus 2. 7 Rumus *Random Forest*

1. $\hat{y}$ adalah prediksi akhir dari *Random Forest*.
2. N adalah jumlah pohon keputusan yang digunakan dalam model.
3. $T_i(x)$ adalah *output* dari pohon keputusan ke- i untuk input x.

Dalam rumus 2.7, algoritma *Random Forest* mengumpulkan prediksi dari semua pohon keputusan yang dihasilkan selama pelatihan untuk menentukan kelas dengan suara terbanyak. Proses ini memastikan bahwa hasil prediksi lebih *robust* dan mengurangi risiko *overfitting*. Pohon

keputusan dalam *Random Forest* dibangun menggunakan metode pengambilan sampel acak atau *bootstrap*, di mana setiap pohon dilatih dengan *subset* data yang berbeda. Setiap pohon hanya menggunakan *subset* acak dari fitur yang tersedia, yang mendiversifikasi pohon-pohon dan meningkatkan kekuatan prediksi keseluruhan. Dengan cara ini, *Random Forest* mampu menangkap pola yang lebih kompleks dalam data, menjadikannya metode yang efektif untuk analisis sentimen. Kelebihan dari *Random Forest* termasuk akurasi yang tinggi dan toleransi terhadap *overfitting*, sementara kelemahannya adalah kebutuhan akan lebih banyak memori dan waktu pemrosesan dibandingkan dengan model sederhana.

2.3.7 Term Frequency

Term Frequency adalah salah satu ukuran penting dalam pemrosesan teks yang digunakan untuk menghitung frekuensi kemunculan suatu istilah dalam sebuah dokumen. Konsep ini berperan krusial dalam analisis teks, terutama dalam konteks model-model pengambilan informasi dan pemodelan bahasa alami. *Term Frequency* membantu dalam menilai relevansi istilah tertentu terhadap konten dokumen yang dianalisis. *Term Frequency* untuk sebuah istilah *t* dalam dokumen *d* dihitung dengan rumus berikut:

$$TF(t,d) = \frac{f_{t,d}}{N_d} \tag{2.8}$$

Rumus 2. 8 Rumus *Term Frequency*

1. $TF(t,d)$: Frekuensi relatif dari istilah *t* dalam dokumen *d*.
2. $f_{t,d}$: Jumlah kemunculan istilah *t* dalam dokumen *d*.
3. N_d: Total jumlah kata yang terdapat dalam dokumen *d*.

Frekuensi istilah $f_{t,d}$ mengukur seberapa sering istilah t muncul dalam dokumen d. Sementara itu, total jumlah kata dalam dokumen

dilambangkan sebagai N_d, yang menunjukkan keseluruhan kata yang terdapat dalam dokumen tersebut. Dengan menerapkan rumus *Term Frequency*, akan diperoleh proporsi kemunculan istilah t yang relatif terhadap panjang dokumen. Nilai *Term Frequency* yang lebih tinggi mengindikasikan bahwa istilah tersebut lebih dominan atau lebih relevan dalam konteks dokumen. Sebagai implikasi, jika suatu istilah t memiliki nilai *Term Frequency* yang lebih besar dibandingkan dengan istilah lainnya, maka dapat disimpulkan bahwa istilah tersebut memainkan peran penting dalam menyampaikan makna utama dokumen tersebut.

2.4 Tools dan Software Penelitian

2.4.1 Jupyter Notebook

Jupyter Notebook adalah sebuah aplikasi *open-source* berbasis web yang memungkinkan pengguna untuk membuat dan berbagi dokumen yang berisi kode program, visualisasi, serta narasi teks yang interaktif. Jupyter Notebook mendukung berbagai bahasa pemrograman, namun paling populer digunakan dengan *Python*, terutama dalam bidang data science, analisis data, dan *machine learning* [52]. Kelebihan utama Jupyter Notebook adalah kemampuannya menyediakan lingkungan interaktif yang memungkinkan eksekusi kode secara bertahap, sekaligus menampilkan *output* berupa grafik, tabel, dan visualisasi lain secara langsung di sebelah kode yang dijalankan [53]. Ini sangat membantu dalam eksplorasi data secara interaktif, *debugging*, dan dokumentasi proses analisis secara transparan.

Dalam pemrosesan teks dan analisis data, Jupyter Notebook memungkinkan peneliti dan praktisi untuk menjalankan kode pemrograman, mencoba berbagai metode *preprocessing* (seperti tokenisasi, *stemming*,) . Bisa langsung melihat hasilnya tanpa perlu membuat aplikasi terpisah. Hal ini mempercepat pengembangan, eksperimen, dan kolaborasi di antara tim yang bekerja pada data yang sama. Dengan kemudahan instalasi dan dukungan luas dari komunitas, Jupyter Notebook menjadi

salah satu *tools* favorit untuk riset dan edukasi di bidang ilmu komputer dan data.

2.4.2 Apify

Apify adalah *platform* otomatisasi web yang memungkinkan pengguna membuat, menjalankan, dan mengelola *web scraper* serta *crawlers* untuk mengekstrak data dari berbagai situs web secara terstruktur. Khususnya untuk *Google Maps Review*, Apify menyediakan *actor* atau skrip siap pakai yang dirancang untuk mengumpulkan ulasan dari halaman lokasi di *Google Maps*. Dengan menggunakan Apify, pengguna dapat secara otomatis mengunduh data ulasan seperti nama pengguna, *rating*, tanggal ulasan, dan isi teks ulasan tanpa perlu melakukannya secara manual, bahkan untuk data dalam jumlah besar. Penggunaan Apify untuk *Google Maps Review* sangat populer dalam berbagai aplikasi bisnis dan riset karena memungkinkan analisis sentimen pelanggan, pemantauan reputasi, dan riset pasar berbasis ulasan pengguna asli. Apify menyediakan API dan antarmuka pengguna yang memudahkan integrasi data hasil ekstraksi ke dalam sistem analitik atau *database* pengguna. Selain itu, kemampuan Apify untuk mengelola paginasi dan menangani berbagai elemen dinamis pada halaman web membuat proses pengambilan data ulasan menjadi stabil, efisien, dan tepat waktu. Dengan demikian, Apify adalah *tool* yang *powerful* untuk melakukan web *scraping Google Maps* secara otomatis dan terstruktur.

2.4.3 CSV Editor

CSV Editor adalah perangkat lunak atau aplikasi yang digunakan untuk membuka, melihat, mengedit, dan memanipulasi file berformat CSV (*Comma-Separated Values*). File CSV merupakan format teks sederhana yang menyimpan data dalam bentuk tabel, di mana setiap baris mewakili satu *record*, dan setiap kolom dipisahkan oleh tanda koma atau *delimiter* lain seperti titik koma. CSV banyak digunakan untuk penyimpanan data *tabular* karena kompatibilitasnya yang luas dengan berbagai program, termasuk *spreadsheet*, *database*, dan alat analisis data. CSV Editor