

## BAB II

### LANDASAN TEORI

#### 2.1 Penelitian Terkait

Beberapa penelitian terdahulu telah memberikan kontribusi yang berharga dalam mendukung penelitian ini.

Tabel 2. 1 Penelitian Terkait

NO	Author	Title	Journal	Method / Tools	Application / Result
1.	Mamata Tharu, Sitaram Pokhrel, Badri Raj Lamichhane (2023)	Sentiment Analysis of Nepali COVID-19 Tweets using BERT-LSTM	Journal of Engineering and Sciences Volume: 2 Nomor Jurnal: 1 Halaman : 49–56 Tahun Terbit: 2023,	LSTM	Penelitian ini menggabungkan BERT dan LSTM untuk analisis sentimen tweet berbahasa Nepal terkait COVID-19. BERT digunakan untuk mengekstrak makna kata secara kontekstual, sedangkan LSTM menangkap urutan dalam data. Dataset yang digunakan adalah NepCov19Tweets. Model mencapai akurasi 76,04% dan F1-score 76%. Hasilnya lebih baik

NO	Author	Title	Journal	Method / Tools	Application / Result
					dibandingkan metode sebelumnya.
2.	Titis Arwindarti Esther Irawati Setiawan, Syaiful Imron (2023)	Analisis Sentimen Komentar Instagram menggunakan Long Short-Term Memory (LSTM) dan Word2Vec	Teknika Journal  Volume: 13 Nomor: 1 Bulan/Tahun: Maret 2023	LSTM	Model LSTM dikombinasikan dengan Word2Vec untuk menganalisis komentar Instagram. Word2Vec membantu menangkap makna semantik komentar. LSTM dipakai untuk menangani sifat sekuensial data. Akurasi model mencapai 84%
3.	Dany Eka Saputra & Auliya Rahman Isnain (2024)	Implementasi algoritma convolutional neural network untuk analisis sentimen bacapres 2024 pada kolom komentar youtube mata najwa 1.	JIPI (Jurnal Ilmiah Penelitian dan Pembelajaran Informatika)  Volume: 9 Nomor: 3	CNN	Penelitian menggunakan CNN untuk analisis sentimen bacapres 2024 di kolom komentar YouTube. Model menunjukkan skor F1 tinggi hingga 94% di beberapa kategori data. Penggunaan SMOTE dan

NO	Author	Title	Journal	Method / Tools	Application / Result
			Bulan/Tahun: September 2024  Halaman : 1431–1441		tuning parameter meningkatkan performa. Skor F1 makro berkisar antara 60–65%. Hal ini menunjukkan ketidakseimbangan data masih berpengaruh.
4.	Mujiono Sadikin and Abi Fauzan (2023)	Evaluation of Machine Learning Approach for Sentiment Analysis using Yelp Dataset	Nama Jurnal: Advances in Artificial Intelligence and Machine Learning  Volume: 3  Nomor: 3  Tahun: 2023  Halaman : 74–87	LSTM	Model LSTM dan MLP dibandingkan untuk analisis ulasan restoran di Yelp. Word2Vec digunakan dalam tahap ekstraksi fitur. LSTM menunjukkan kinerja lebih unggul dibanding MLP. Akurasi LSTM mencapai 91%, sedangkan MLP hanya 76%. LSTM efektif untuk data sekuensial.
5.	Ekin Fergan	Course Review Sentiment Analysis: A	IEEE Access	LSTM	Penelitian membandingkan model ML dan DL pada

NO	Author	Title	Journal	Method / Tools	Application / Result
	Tsegaye Misikir Tashu (2023)	Comparative Study Of Machine Learning and Deep Learning Methods	Volume: 10 Nomor Jurnal: - Tahun: 2022		ulasan Coursera. Teknik seperti TF-IDF, GloVe, dan berbagai model seperti Naive Bayes, SVM, LSTM, dan BERT digunakan. BERT memberikan hasil terbaik. Masalah ketidakseimbangan data diidentifikasi sebagai tantangan utama. Studi ini menyarankan solusi balancing untuk kinerja lebih baik.
6.	Minji Kim, Subeen Kim, Yoonha Park, Sangwoo Bahn, Sung Hee Ahn, Bhavadh arani	User Sentiment Analysis Based on Securities Application Elements	Nama Jurnal: Behavioral Sciences Volume: 14 Nomor: 3 Tahun: 2024	logistic regression, support vector machine (SVM), LSTM (Long Short-Term Memory), and 1D CNN	Analisis dilakukan terhadap ulasan aplikasi sekuritas Korea. Metode ABSA digunakan untuk mengidentifikasi sentimen terhadap elemen tertentu seperti "update" dan

NO	Author	Title	Journal	Method / Tools	Application / Result
	NambiN arayanan  (2024)		Halaman : 1-19		“login”. Model yang digunakan mencakup logistic regression, SVM, LSTM, dan CNN. Studi menunjukkan potensi ABSA dalam mendukung desain aplikasi berbasis pengguna. ABSA juga hemat biaya dan skalabel.
7.	Rosa Eliviani, Dwi Diana Wazaumi  (2024)	Exploring Sentiment Trends: Deep Learning Analysis of Social Media Reviews on Google Play Store by Netizens	Nama Jurnal: International Journal of Advances in Data and Information Systems (IJADIS)  Volume: 5  Nomor: 1  Tanggal Terbit:	LSTM	Penelitian ini menganalisis 1.500 ulasan aplikasi Instagram. Model LSTM digunakan untuk klasifikasi sentimen. Akurasi model sebesar 77,77%, namun F1-score hanya 0,15. Hal ini menunjukkan kinerja masih moderat. Studi memberi wawasan tentang

NO	Author	Title	Journal	Method / Tools	Application / Result
			28 Maret 2024		kelebihan dan kelemahan aplikasi.
8.	Fitroh, Fahmi Hudaya (2023)	Systematic Literature Review: Analisis Sentimen Berbasis Deep Learning	Nama Jurnal: Nasional dan Sistem Informasi Teknisi  Volume: 9  Nomor: 2  Maret 2023	LSTM	Penelitian ini merupakan tinjauan sistematis terhadap penggunaan deep learning untuk analisis sentimen. LSTM dan CNN adalah model yang paling sering digunakan. LSTM mencatat akurasi tertinggi hingga 99,59%. Rata-rata akurasi di berbagai studi adalah 89%. Studi ini menyoroti efektivitas deep learning secara umum.
9.	Musfiroh, Abu Tholib, Zainal Arifin (2024)	Sentiment Analysis of Shopee App Reviews on Google Play Store Using TF-IDF and Long Short-Term	Nama Jurnal: Journal of Electrical Engineering and Computer	LSTM	Analisis dilakukan pada 3.565 ulasan aplikasi Shopee dari Google Play. Sentimen dibagi menjadi positif, netral, dan negatif.

NO	Author	Title	Journal	Method / Tools	Application / Result
		Memory (LSTM)	(JEECOM) Volume: 6 Nomor: 2 2024		LSTM digunakan dengan TF-IDF untuk praproses data. Akurasi keseluruhan mencapai 83%. Namun, model kesulitan mengklasifikasi sentimen netral karena ketidakseimbangan data.
10.	Fitroh, Fitroh Hudaya, Fahmi (2023)	Analisis Sentimen Berbasis Deep Learning	Nama Jurnal: Nasional dan Sistem Informasi Teknisi Volume: 9 Nomor: 2 Mei 2024	Bidirectional Long Short-Term Memory (BiLSTM)	Model BiLSTM digunakan untuk menangkap konteks dua arah dalam data teks. Model diuji untuk klasifikasi sentimen. Hasil evaluasi menunjukkan akurasi 99,14% dan F1-score 99,12%. Model sangat seimbang dalam hal presisi dan recall. Ini menunjukkan kinerja yang hampir sempurna.

<b>NO</b>	<b>Author</b>	<b>Title</b>	<b>Journal</b>	<b>Method / Tools</b>	<b>Application / Result</b>
11.	Rina Refianti, Achmad Benny Mutiara, Ryan Arya Putra (2024)	A Lexicon-Based Long Short-Term Memory (LSTM) Model for Sentiment Analysis to Classify Halodoc Application Reviews on Google Playstore	Journal of Applied Data Sciences (JADS)  Volume: 5 Nomor Jurnal: 1 Tahun: 2024	LSTM (Long Short-Term Memory)	Model LSTM berbasis leksikon digunakan untuk ulasan aplikasi Halodoc. Sentimen diklasifikasikan menjadi positif, negatif, dan netral. Dataset terdiri dari lebih dari 19.000 ulasan. Model mencapai akurasi 85,3% dan F1-score sebanding. Aplikasi web dengan Flask juga dikembangkan untuk klasifikasi.
12.	Jatla Srikanth, Avula Damodaram Shanmugam (2023)	A Deep LSTM-RNN Classification Method for COVID-19 Twitter Review Based on Sentiment Analysis	Scalable Computing: Practice and Experience (SCPE)  Volume: 24 Nomor Jurnal: 3	Long Short-Term Memory (LSTM) with Synthetic Minority Oversampling Technique (SMOTE)	Model LSTM dengan SMOTE digunakan untuk analisis sentimen tweet COVID-19. Sentimen dibagi menjadi senang, sedih, netral, dan marah. Akurasi model mencapai 97,5% dengan presisi 98%.

NO	Author	Title	Journal	Method / Tools	Application / Result
			Tahun: 2024		Model ini mengungguli metode seperti Bi-LSTM dan CNN-LSTM. Hasil memberikan wawasan berharga tentang persepsi publik.
13.	Angela Lisanthoni, Ellexia Leonie Gunawan, Chelsea Ayu Adhigiadany, Dwi Arman Prasetya (2024)	Penerapan LSTM dalam Analisis Sentimen Berbasis Lexicon untuk Meningkatkan Sistem Pemantauan Citra PLN di Platform Digital	Prosiding Seminar Nasional Sains Data Volume: 4 Nomor Jurnal: 2 Tahun: 2024	LSTM (Long Short-Term Memory)	Metode ini diterapkan untuk mengklasifikasikan sentimen dalam tweet yang terkait dengan COVID-19, mengkategorikannya ke dalam empat kelas: senang, sedih, netral, dan marah. Model deep learning yang diusulkan mencapai hasil yang mengesankan, dengan akurasi 97,5%, presisi 98%, recall 89%, dan mengungguli metode lain seperti Bi-LSTM, LSTM-RNN, dan

NO	Author	Title	Journal	Method / Tools	Application / Result
					CNN-LSTM. Evaluasi tersebut menunjukkan peningkatan kinerja yang signifikan dibandingkan dengan model yang sudah ada,
14.	Jimmy Tjen (2025)	Pengaruh Jenis Stopwords terhadap Akurasi Model Multinomial Naïve Bayes dalam Proses Sentimen Analisis	Vol. 16 No. 01 (2025): Jurnal Buana Informatika, Volume 16, Nomor 01, April 2025	Uji Friedman	Berhasil menemukan cara bagaimana melihat seberapa perbedaan performa antar model signifikan secara statistik

Tabel 2.1 merangkum sejumlah penelitian terdahulu yang menjadi dasar bagi studi ini. Berbagai riset, seperti nomor 7 dan 8, menunjukkan bahwa LSTM efektif dalam mengkategorikan ulasan pengguna ke dalam sentimen positif maupun negatif dengan akurasi tinggi. Penelitian nomor 3 dan 8 menegaskan keunggulan CNN dalam menangkap pola penting dalam teks, sehingga tetap handal untuk klasifikasi baik biner maupun multi kelas. Selanjutnya, penelitian nomor 10 dan 12 mengkaji pendekatan gabungan BiLSTM dan CNN, yang memadukan kemampuan BiLSTM membaca konteks dua arah dan keahlian CNN mengekstrak fitur, menghasilkan representasi teks lebih kaya serta akurasi yang lebih baik dibanding model tunggal. Selain itu, penelitian nomor 14 menyoroti perbedaan performa algoritma berdasarkan metrik accuracy, precision, recall, dan F1-Score dengan melakukan uji

Friedman. Temuan dari penelitian pendekatan ini memberikan fondasi kuat untuk memahami pengalaman pengguna serta menilai kualitas layanan aplikasi kencan. Penelitian-penelitian lain dalam tabel juga menambah wawasan terkait prapemrosesan data yang digunakan dalam studi ini.

Penelitian ini tidak hanya berfokus pada pemetaan sentimen positif dan negatif atau mengevaluasi performa model seperti BiLSTM, LSTM, dan CNN. Kebaruan utama terletak pada tujuan yang lebih spesifik, yaitu menggali pola kata-kata yang berkaitan dengan kasus penipuan yang sering terjadi dalam aplikasi kencan daring. Dengan demikian, studi ini berupaya mengidentifikasi elemen-elemen linguistik yang relevan dengan indikasi penipuan, yang belum banyak dibahas dalam penelitian sebelumnya. Pendekatan ini memberikan kontribusi baru bagi pemahaman dan pencegahan risiko di platform kencan daring sekaligus memperkaya analisis sentimen konvensional dengan aspek konteks keamanan pengguna.

## **2.2 Teori Penelitian**

### **2.2.1 Analisis Sentimen**

Analisis sentimen, yang sering disebut sebagai opinion mining, adalah sebuah disiplin ilmu yang menggabungkan teknik-teknik dari bidang ilmu komputer, linguistik, dan psikologi untuk mengekstrak, mengidentifikasi, dan mengukur sentimen subjektif yang terkandung di dalam teks[15]. Tujuan utamanya adalah untuk memahami opini, emosi, dan sikap yang terkandung dalam teks mengenai suatu topik, produk, layanan, atau individu[15]. Proses ini menggunakan Natural Language Processing (NLP) dan algoritme pembelajaran mesin untuk mengotomatiskan identifikasi polaritas sentimen (positif, negatif, atau netral) dan tingkat intensitas[16]. Secara lebih luas, analisis sentimen memungkinkan pemahaman yang lebih dalam tentang bagaimana orang merespons suatu masalah, menjadikannya alat yang berharga di berbagai bidang mulai dari pemasaran hingga kebijakan publik[16].

Dengan meningkatnya jumlah data teks di era digital, analisis sentimen menjadi semakin penting di berbagai sektor[17]. Dalam dunia bisnis, perusahaan

menggunakan analisis sentimen untuk memantau citra merek di media sosial, menganalisis umpan balik pelanggan, dan mengidentifikasi tren pasar[17]. Di dunia politik, teknik ini digunakan untuk mengukur opini publik terhadap kandidat atau kebijakan tertentu, memprediksi hasil pemilu, dan mengidentifikasi isu-isu yang memicu perdebatan[18]. Selain itu, analisis sentimen juga diterapkan di bidang pendidikan, kesehatan, keuangan, dan keamanan untuk mengidentifikasi pola yang relevan dan membantu membuat keputusan yang lebih tepat. Kemampuan analisis sentimen untuk memproses data dalam jumlah besar dengan cepat memberikan wawasan berharga yang sebelumnya sulit diperoleh[18].

Meskipun analisis sentimen menawarkan banyak potensi, ada beberapa tantangan dan keterbatasan yang harus diperhatikan[18]. Salah satu tantangan terbesar adalah kompleksitas bahasa manusia, termasuk penggunaan sarkasme, ironi, dan konteks spesifik yang dapat memengaruhi pemahaman sentimen[19]. Selain itu, perbedaan budaya dan bahasa dapat memengaruhi cara sentimen diekspresikan dan dipahami[20]. Mengembangkan model analisis sentimen yang akurat dan dapat diandalkan membutuhkan pemahaman yang mendalam tentang nuansa bahasa dan konteks budaya[19]. Oleh karena itu, penelitian terus mengembangkan teknik dan algoritme yang lebih canggih, seperti deep learning dan transfer learning, untuk mengatasi tantangan ini dan meningkatkan efektivitas analisis sentimen di berbagai area aplikasi[21].

### **2.2.2 Aplikasi Kencan**

Perkembangan aplikasi kencan dalam kemajuan teknologi disebabkan oleh kombinasi antara inovasi teknologi dan kebutuhan manusia akan koneksi antarpribadi[7]. Dengan memanfaatkan kemajuan dalam komputasi mobile, kecerdasan buatan, dan analisis data, aplikasi kencan memberikan pengalaman kencan yang lebih efisien dan terhubung[22]. Algoritma pencocokan yang canggih, pemetaan lokasi berbasis GPS, dan integrasi media sosial memberikan kemampuan untuk menciptakan pertemuan yang lebih relevan[22]. Dampak-dampak ini tidak hanya membuat pencarian pasangan menjadi lebih mudah, tetapi juga membentuk pergeseran budaya dalam cara orang menjalin hubungan dan merespons

perkembangan sosial[23]. Oleh karena itu, aplikasi kencan tidak hanya mencerminkan kemajuan teknologi, tetapi juga berperan dalam membentuk dinamika hubungan manusia di era digital ini[24].

Aplikasi kencan melibatkan pemahaman aspek-aspek psikologis, sosial, teknologi, dan budaya. Teori psikologi membantu dalam memahami mengapa orang mencari pasangan dan bagaimana mereka berinteraksi di dunia digital[25]. Aplikasi kencan mencerminkan perkembangan teknologi komunikasi dan bagaimana orang berinteraksi secara daring[24]. Selain itu, aplikasi kencan juga memengaruhi perubahan dalam tatanan sosial dan budaya, dengan teori budaya dan sosiologi membantu dalam memahami perkembangan hubungan, norma sosial, dan bagaimana identitas dibangun dalam era digital[26]. Etika dan privasi juga menjadi perhatian penting, termasuk pertimbangan etika dan hak-hak individu dalam lingkungan digital[27].

### **2.2.3 Tinder**

Tinder merupakan aplikasi pencarian pasangan berbasis digital yang pertama kali diluncurkan pada tahun 2012 dan mengalami pertumbuhan yang signifikan dalam waktu relatif singkat. Aplikasi ini memanfaatkan teknologi geolokasi serta sistem interaksi berbasis “swipe” untuk mempertemukan pengguna yang menunjukkan ketertarikan satu sama lain. Sejak kemunculannya, Tinder terus melakukan inovasi dengan menghadirkan berbagai fitur premium, seperti Tinder Plus dan Tinder Gold, serta memperluas jangkauan layanannya melalui program Tinder U yang secara khusus ditujukan bagi kalangan mahasiswa. Kehadiran berbagai fitur tersebut memperkuat eksistensi Tinder sebagai salah satu platform kencan daring paling populer di tingkat global, khususnya di kalangan generasi muda[28].

Tingkat popularitas Tinder semakin meningkat seiring dengan terjadinya perubahan pola interaksi sosial, terutama selama masa pandemi Covid-19 yang membatasi interaksi langsung antarindividu. Hingga saat ini, aplikasi tersebut telah diunduh lebih dari 400 juta kali dengan jumlah pengguna aktif bulanan yang mencapai jutaan. Kondisi ini menunjukkan bahwa teknologi digital memiliki peran

strategis dalam membentuk cara baru masyarakat dalam membangun relasi interpersonal. Selain menyediakan kemudahan dalam menjalin hubungan, Tinder juga menunjukkan kepedulian terhadap aspek keamanan pengguna melalui penyediaan fitur anti-penipuan, tombol darurat (“panic button”), serta pusat edukasi mengenai keamanan digital. Oleh karena itu, Tinder tidak hanya berfungsi sebagai sarana pencarian pasangan, melainkan juga sebagai cerminan dari dinamika transformasi sosial dalam era digital[28].

#### **2.2.4 Google Play Reviews**

Istilah "Google Play Reviews" mengacu pada tanggapan dan pengamatan yang dilakukan pelanggan di Google Play Store terkait aplikasi seluler yang telah mereka gunakan dan seberapa puas mereka dengan aplikasi tersebut. Evaluasi ini dapat dievaluasi untuk mendapatkan wawasan tentang bagaimana perasaan pengguna tentang aplikasi dan untuk menemukan area yang perlu diperbaiki. Dalam beberapa tahun terakhir, Ulasan Google Play telah berkembang menjadi sumber data penting untuk bisnis dan pengembang, memungkinkan mereka untuk lebih memahami persyaratan dan kecenderungan audiens target mereka masing-masing [29]. Evaluasi tersebut dapat dievaluasi menggunakan berbagai metode, salah satunya dikenal sebagai analisis sentimen, untuk mendapatkan wawasan yang bermanfaat dan meningkatkan pengalaman pengguna aplikasi secara keseluruhan. Proses mengidentifikasi dan mengekstraksi informasi subyektif dari data tekstual disebut sebagai analisis sentimen. Analisis sentimen merupakan metodologi yang mencakup penggunaan pemrosesan bahasa alami, pembelajaran mesin, dan analitik teks. Saat diterapkan pada pengaturan Ulasan Google Play, analisis sentimen dapat digunakan untuk mengkategorikan ulasan secara otomatis sebagai nada positif, negatif, atau tidak memihak berdasarkan bahasa yang digunakan dalam ulasan [30].

Word clouds, bar charts, dan heat maps merupakan beberapa alat dan teknik yang dapat digunakan setelah sentimen Ulasan Google Play dianalisis untuk memberikan sinopsis yang jelas dan ringkas tentang sentimen umum terkait aplikasi [31]. Hal ini dapat dicapai setelah temuan analisis telah diilustrasikan. Ini dapat membantu bisnis dan pengembang dalam menentukan masalah atau kekhawatiran

yang paling sering disuarakan oleh pengguna dan memprioritaskan pekerjaan yang mereka lakukan untuk meningkatkan pengalaman pengguna.

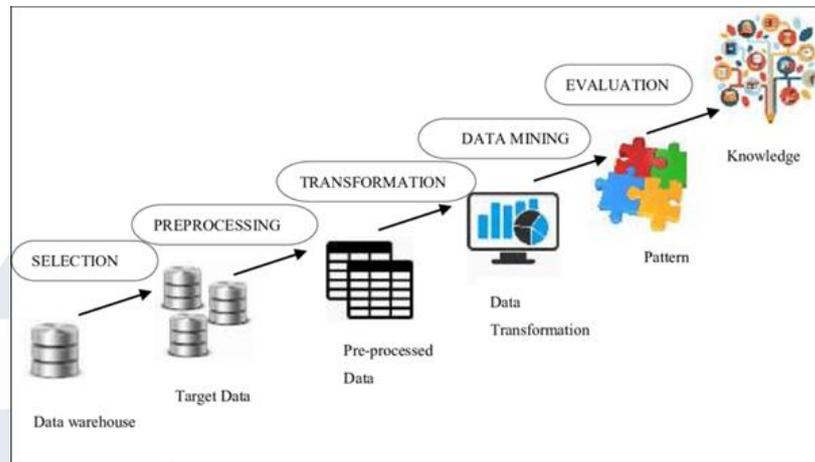
Secara keseluruhan, Ulasan Google Play mampu menawarkan informasi yang mencerahkan terkait perasaan pengguna terhadap aplikasi seluler. Bisnis dan pengembang perangkat lunak dapat mengevaluasi evaluasi ini menggunakan teknik dari bidang analisis sentimen untuk mendapatkan wawasan yang signifikan dan meningkatkan pengalaman pengguna. Menganalisis Ulasan Google Play telah menjadi tahap penting dalam proses peningkatan kualitas dan fungsionalitas aplikasi seluler secara keseluruhan sebagai akibat langsung dari semakin pentingnya komentar pengguna dalam proses pengembangan aplikasi seluler.

## **2.3 Framework dan Algoritma**

### **2.3.1 Knowledge Discovery In Databases (KDD)**

Knowledge Discovery in Database (KDD) merupakan metode yang digunakan untuk melakukan proses penambangan data secara sistematis. Fayyad et al. (1996) mengartikan KDD sebagai serangkaian langkah yang memakai teknik data mining untuk mengekstrak informasi berharga dan pola-pola dalam data yang tersembunyi. Proses ini menggunakan algoritma khusus untuk mendeteksi pola-pola tersebut dalam kumpulan data yang besar. Dunham (2003) menyatakan bahwa KDD melibatkan beberapa tahapan penting, yakni seleksi data, pra-pemrosesan, transformasi, penambangan, serta interpretasi dan evaluasi. Masing-masing tahap tersebut berkontribusi dalam menyiapkan data dan menemukan pengetahuan baru yang bermanfaat[32].

U N I V E R S I T A S  
M U L T I M E D I A  
N U S A N T A R A



Gambar 2. 1 Framework KDD[32].

Dalam pengertian yang lebih luas, KDD adalah serangkaian tahapan yang menyeluruh untuk mengungkap pola dan makna dari data mentah. Proses ini tidak hanya sebatas penerapan teknik data mining semata, melainkan meliputi seluruh prosedur mulai dari pengambilan data yang relevan hingga evaluasi hasil. Tahapan ini mencakup proses pembersihan serta persiapan data agar cocok untuk dianalisis lebih lanjut. Selanjutnya, data diolah melalui transformasi sebelum diterapkan algoritma penambangan untuk menemukan pola tersembunyi. Akhirnya, hasil yang diperoleh dievaluasi dan diinterpretasikan sehingga dapat dimanfaatkan untuk pengambilan keputusan[32].

Seleksi data merupakan tahap awal dalam KDD yang berfokus pada pemilihan subset data yang relevan dari kumpulan data besar. Tujuannya adalah agar proses analisis nantinya dapat berjalan dengan efektif dan terarah. Pada tahap pra-pemrosesan, data yang sudah diseleksi dibersihkan dari kesalahan dan inkonsistensi serta disiapkan dalam bentuk yang dapat diproses lebih lanjut. Transformasi data dilakukan dengan tujuan mengubah data ke bentuk representasi yang sesuai untuk penerapan teknik analitik. Dengan data yang telah siap, proses penambangan dilanjutkan dengan penggunaan algoritma untuk mengekstraksi pola dan informasi penting dari data tersebut[32].

Proses penambangan data (data mining) adalah bagian inti dari KDD yang menggunakan beragam algoritma untuk menemukan pola, hubungan, dan tren yang

tersembunyi secara signifikan dalam data. Metode ini memungkinkan penggalian informasi yang sebelumnya tidak terlihat melalui teknik analisis konvensional. Setelah pola ditemukan, tahap evaluasi dilakukan untuk menilai relevansi dan keakuratan pola-pola tersebut. Interpretasi hasil berguna untuk memberikan makna yang jelas dan aplikatif dari pola yang dihasilkan. Tahapan ini memastikan bahwa pengetahuan yang diperoleh dapat digunakan secara efektif dalam pengambilan keputusan strategis[32].

Tahapan evaluasi dalam KDD berfungsi untuk menguji dan memverifikasi hasil-hasil yang diperoleh dari proses penambangan data. Proses ini melibatkan pengecekan keandalan pola dan informasi yang telah ditemukan sesuai dengan tujuan analisis. Evaluasi juga membantu dalam menyesuaikan algoritma agar menghasilkan informasi yang lebih baik. Interpretasi hasil memungkinkan para praktisi untuk memahami arti dan dampak dari pola yang terdeteksi dalam konteks bisnis atau penelitian. Akhirnya, tahap ini memastikan bahwa pengetahuan yang dihasilkan berguna dan dapat diaplikasikan secara optimal[32].

### **2.3.2 Class Weights**

Class weights adalah cara yang sering digunakan dalam pembelajaran mesin untuk mengatasi masalah data yang tidak seimbang, di mana jumlah data pada satu kelas jauh lebih banyak dibanding kelas lain[33]. Kondisi ini sering membuat model cenderung “memperhatikan” kelas yang dominan saja, sehingga hasil prediksi untuk kelas yang lebih sedikit jumlahnya jadi kurang akurat[33]. Akibatnya, meskipun model terlihat punya performa yang bagus secara keseluruhan, sebenarnya ia kurang mampu mengenali pola penting dari kelas minoritas[33]. Masalah ini sangat umum terjadi terutama pada data nyata yang tidak selalu seimbang. Oleh karena itu, diperlukan teknik khusus agar model bisa belajar secara adil dari semua kelas yang ada.

Teknik class weights bekerja dengan memberikan bobot lebih besar pada kelas yang jumlah datanya lebih sedikit selama proses pelatihan model. Dengan begitu, kesalahan yang terjadi pada kelas minoritas dianggap lebih penting dan model terdorong untuk lebih fokus mempelajari pola dari kelas tersebut[33].

Berbeda dengan metode lain seperti oversampling atau undersampling yang mengubah jumlah data, class weights tidak mengubah data sama sekali sehingga proses pelatihan tetap efisien dan tidak menambah kompleksitas[33]. Pendekatan ini sangat praktis karena bisa langsung diterapkan pada banyak algoritma tanpa perlu manipulasi data[33]. Hal ini membuat class weights menjadi solusi yang elegan untuk masalah ketidakseimbangan data.

Dalam konteks penggunaan algoritma seperti LSTM, BiLSTM, dan CNN yang sering dipakai untuk pemrosesan data berurutan dan teks, class weights menjadi pilihan tepat untuk menangani ketidakseimbangan kelas[33]. Model-model ini biasanya memproses data yang kompleks dan besar, sehingga mengubah jumlah data dengan oversampling atau undersampling bisa memperlambat pelatihan dan menambah risiko overfitting[33]. Dengan class weights, model tetap belajar dari data asli namun dengan perhatian lebih pada kelas minoritas, sehingga performa prediksi menjadi lebih seimbang dan akurat. Selain itu, class weights mudah diintegrasikan dalam framework deep learning yang umum digunakan untuk LSTM, BiLSTM, dan CNN

#### **2.3.4 Natural Language Processing (NLP)**

Natural Language Processing (NLP) adalah cabang dari kecerdasan buatan (AI) yang mempelajari bagaimana komputer dapat memahami, memproses, dan menghasilkan bahasa manusia secara alami[34]. NLP menggabungkan linguistik, ilmu komputer, dan pembelajaran mesin untuk menganalisis teks atau ucapan dalam bentuk yang dapat dipahami mesin. Dalam praktiknya, NLP menggunakan berbagai teknik, seperti tokenisasi, stemming, lemmatization, dan parsing untuk memecah bahasa menjadi unit yang lebih kecil dan mudah diproses[34]. Pendekatan NLP terdiri dari metode berbasis aturan (rule-based) dan metode berbasis statistik atau pembelajaran mesin, termasuk model modern seperti transformator (contohnya BERT dan GPT)[34]. Teknologi ini mendasari berbagai aplikasi, seperti penerjemahan otomatis, analisis sentimen, pengenalan entitas bernama (NER), chatbot, dan sistem rekomendasi berbasis teks.

### 2.3.5 Text Classification

Text Classification merupakan teknik machine learning yang dapat mengorganisir dan mengelompokkan beragam jenis data, termasuk dokumen, rekam medis, dan teks lainnya[35]. Fungsi klasifikasi teks sebagai kerangka kerja efektif dalam pemrosesan data sangat umum, digunakan secara luas untuk tujuan seperti deteksi spam, kategorisasi berita, dan pengelompokan topik[35]. Di samping itu, dalam konteks analisis sentimen, text classification memegang peranan penting[35]. Teknik ini membantu mengategorikan teks ke dalam kelas sentimen positif, dan negatif, dengan menggunakan algoritma machine learning untuk memahami pola linguistik dan kontekstual dalam teks[35]. Pendekatan ini memungkinkan pengenalan dan penilaian sentimen secara otomatis, memudahkan analisis besar-besaran terhadap perasaan yang terungkap dalam teks, seperti ulasan produk atau komentar di media sosial[35]

Text classification dapat dilaksanakan melalui dua pendekatan: manual dan otomatis[35]. Dalam klasifikasi manual, proses mencakup pemilihan kategori, pembuatan panduan klasifikasi, dan penentuan kategori teks oleh annotator manusia. Meskipun memberikan hasil berkualitas tinggi, metode ini memerlukan waktu yang lebih lama. Di sisi lain, klasifikasi otomatis menggunakan teknologi machine learning, NLP, dan metode lain untuk mengklasifikasikan teks secara otomatis dengan efisiensi yang lebih tinggi. Pendekatan otomatis memungkinkan penanganan volume teks besar dan peningkatan kinerja model melalui pembelajaran dari data baru. Meski demikian, klasifikasi manual tetap relevan saat interpretasi kontekstual dan pemahaman mendalam diperlukan untuk klasifikasi yang sangat akurat. Penggabungan kedua pendekatan ini menciptakan solusi yang holistik dan adaptif, memenuhi berbagai kebutuhan klasifikasi teks.

U N I V E R S I T A S  
M U L T I M E D I A  
N U S A N T A R A

Secara umum, klasifikasi teks mencakup empat tingkat cakupan yang berbeda untuk diterapkan[36] :

1. Tingkat Dokumen

Pada tingkat dokumen, algoritme mengidentifikasi kategori yang relevan dari seluruh dokumen, memberikan gambaran umum yang luas tentang konten.

2. Tingkat Paragraf:

Tingkat paragraf membawa analisis lebih dalam, di mana algoritme memperoleh kategori yang relevan dari satu paragraf. Hal ini memberikan pemahaman yang lebih rinci tentang subbagian dokumen.

3. Tingkat Kalimat:

Pada tingkat kalimat, fokus analisis berpindah ke tingkat kalimat tunggal. Algoritme mencari kategori yang relevan untuk setiap kalimat, menghasilkan informasi yang lebih spesifik dan terfokus.

4. Tingkat Sub-Kalimat:

Tingkat sub-kalimat memberikan tingkat perincian tertinggi, di mana algoritme mendapatkan kategori yang relevan dari sub-ekspresi dalam satu kalimat. Hal ini memberikan pemahaman yang terperinci pada tingkat frasa atau struktur kalimat.

Langkah terpenting dalam klasifikasi teks mencakup pemilihan pengklasifikasi terbaik berdasarkan data yang ada dan pemahaman yang menyeluruh tentang konsep setiap metode yang digunakan. Setiap metode memiliki kelebihan dan keefektifannya masing-masing. Klasifikasi teks, sebagai metode pembelajaran mesin yang sering digunakan dalam penelitian, unggul karena kemudahannya untuk dipahami. Pendekatan ini memberikan fleksibilitas untuk menentukan kategori yang relevan pada berbagai tingkat

detail, mulai dari konteks keseluruhan dokumen hingga detail terkecil dalam sebuah kalimat[37].

### 2.3.6 Adam (Adaptive Moment Estimation)

Adam merupakan salah satu metode optimasi yang paling populer di dunia machine learning, khususnya deep learning[38]. Metode ini dirancang untuk membantu model belajar dengan lebih cepat ketika berhadapan dengan data yang kompleks. Fitur yang membuat Adam istimewa adalah kemampuannya untuk melacak dua jenis informasi saat latihan berlangsung: berapa banyak perubahan (gradien) yang dibutuhkan, dan seberapa sering perubahan itu terjadi. Informasi ini kemudian dikoreksi dan digunakan untuk memperbarui bobot dalam model. Dengan pendekatan ini, Adam dapat membuat keputusan yang lebih tepat dalam mengarahkan proses pembelajaran, sehingga model dapat belajar dengan lebih akurat tanpa terlalu banyak melakukan trial and error[38].

Hitung estimasi momentum pertama (mean of gradients):

$$m_t = \beta_1 m_{t-1} + (1 - \beta_1) g_t$$

Rumus 2. 1 Rumus Adam (Adaptive Moment Estimation)

$m_t$  : Estimasi momentum pertama pada iterasi ke-t, yaitu rata-rata bergerak dari gradien saat ini dan sebelumnya. Ini berfungsi untuk mengakumulasi gradien secara eksponensial.

$\beta_1$  : Koefisien decay untuk momentum pertama, biasanya bernilai 0.9. Menentukan seberapa besar pengaruh gradien sebelumnya terhadap estimasi saat ini.

$m_{t-1}$  : Estimasi momentum pertama pada iterasi sebelumnya (ke-t-1).

$g_t$  : Gradien dari fungsi loss pada iterasi ke-t. Ini adalah arah dan besarnya perubahan parameter yang diinginkan.

$1 - \beta_1$  : Bobot untuk gradien saat ini, memastikan bahwa gradien terbaru tetap berkontribusi pada estimasi momentum.

Keunggulan lain dari Adam adalah kemudahan penggunaannya[38]. Metode ini tidak terlalu sensitif terhadap pengaturan awal parameter, seperti laju pembelajaran, sehingga cocok digunakan oleh siapa saja, baik pemula maupun peneliti yang sudah berpengalaman. Dalam banyak kasus, Adam mampu mempercepat proses pelatihan model dan meningkatkan kinerja, bahkan ketika bekerja dengan data dalam jumlah besar atau struktur model yang kompleks. Karena keandalannya, Adam kini menjadi salah satu algoritme bawaan yang paling banyak digunakan dalam platform pembelajaran mesin modern. Hal ini menjadikan Adam pilihan utama dalam berbagai aplikasi machine learning saat ini[38].

### 2.3.7 Bidirectional Long Short Term Memory (LSTM)

Bidirectional Long Short-Term Memory (BiLSTM) merupakan pengembangan dari model Long Short-Term Memory (LSTM) yang dibuat untuk meningkatkan pemahaman konteks pada data yang berurutan, terutama pada pemrosesan teks. LSTM sendiri merupakan salah satu jenis Recurrent Neural Network (RNN) yang didesain untuk mengatasi masalah kehilangan informasi pada jaringan RNN biasa, sehingga mampu mengingat informasi pada data berurutan yang panjang[39]. Namun, LSTM standar hanya memproses data dalam satu arah, dari awal hingga akhir, sehingga informasi yang muncul setelah suatu titik dalam urutan tidak dapat digunakan untuk prediksi pada titik tersebut[39].

$$ht \rightarrow = \text{LSTMforward}(xt, ht - 1 \rightarrow)$$

$$ht \leftarrow = \text{LSTMbackward}(xt, ht + 1 \leftarrow)$$

Rumus 2. 2 Rumus Bidirectional Long Short Term Memory (LSTM)

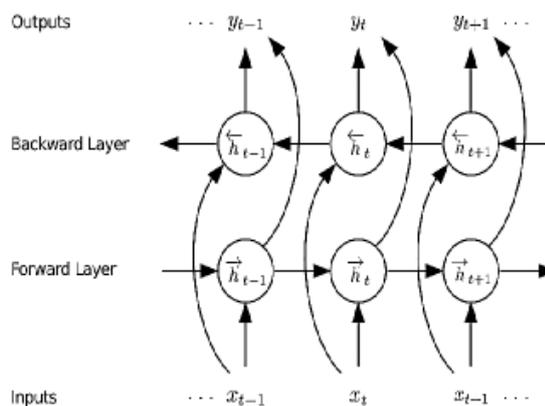
$ht \rightarrow$  : Hidden state dari LSTM arah maju pada waktu  $t$

$ht \leftarrow$  : Hidden state dari LSTM arah mundur pada waktu  $t$

$ht$ : Output gabungan dari kedua arah, biasanya berupa concatenation

BiLSTM mengatasi keterbatasan ini dengan menggunakan dua lapisan LSTM yang berjalan secara bersamaan, satu lapisan memproses data dari awal hingga akhir (maju), dan lapisan lainnya dari akhir hingga awal (mundur). Dengan cara ini, setiap elemen dalam urutan dapat memperoleh konteks dari arah sebelumnya dan yang akan datang. Pendekatan ini membuat BiLSTM sangat efektif untuk tugas-tugas seperti analisis sentimen, klasifikasi teks, dan deteksi spam karena mampu menangkap informasi yang lebih akurat dan tepat[39].

Pada gambar 2.2 terlihat diagram arsitektur jaringan saraf rekuren bidirectional yang terdiri dari dua lapisan utama, yaitu forward layer dan backward layer. Input data diproses secara simultan dari arah maju (forward) dan mundur (backward), sehingga setiap titik waktu dapat memperoleh informasi dari konteks sebelumnya dan sesudahnya. Node-node yang diberi label 'h' menunjukkan status tersembunyi pada setiap langkah waktu, dengan panah yang menggambarkan aliran informasi antar lapisan dan waktu. Pendekatan ini memungkinkan model untuk menangkap pola yang lebih kaya dan kontekstual dalam data berurutan, sehingga meningkatkan performa pada tugas-tugas seperti pemrosesan bahasa alami[39].

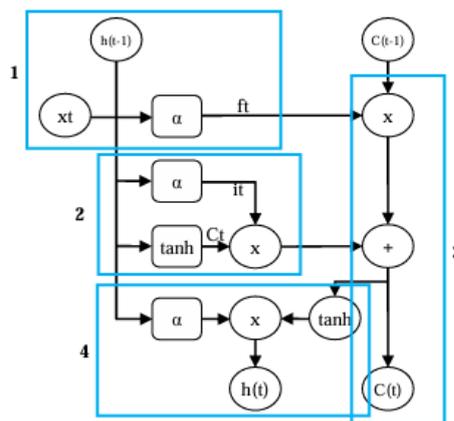


Gambar 2. 2 Struktur BI-LSTM [39]

### 2.3.8 Long Short Term Memory (LSTM)

Long Short-Term Memory (LSTM) adalah jenis Recurrent Neural Network (RNN) yang dikembangkan untuk mengatasi masalah yang sering dihadapi oleh RNN tradisional, terutama dalam memproses data urutan yang panjang. RNN standar cenderung mengalami masalah seperti vanishing gradient dan exploding gradient, yang menyebabkan mereka kesulitan untuk mengingat informasi dalam urutan yang sangat panjang. LSTM mengatasi masalah ini dengan struktur yang lebih kompleks, memungkinkan model untuk mengingat informasi dalam jangka waktu yang lebih lama, bahkan lebih dari seribu langkah waktu, tergantung pada kompleksitas jaringan dan data yang diproses[40].

Pada gambar 2.3 ini merupakan struktur dari jaringan Long Short-Term Memory (LSTM), yang digunakan untuk memproses data sekuensial[41]. LSTM memiliki tiga komponen utama: input, hidden state, dan cell state. Pada langkah pertama, input  $x_t$  dan hidden state sebelumnya  $h(t - 1)$  diproses melalui gate yang disebut forget gate  $f_t$  yang memutuskan informasi mana yang akan dilupakan dari cell state sebelumnya  $C(t - 1)$ . Output dari forget gate kemudian digunakan untuk memperbarui cell state dengan informasi yang relevan[41].



Gambar 2. 3 Prosedur LSTM[40]

Pada langkah kedua, informasi baru yang relevan diproses melalui input gate  $i_t$  yang mengontrol informasi mana yang akan ditambahkan ke dalam cell state.

Selain itu, ada juga cell state candidate  $\check{C}_t$ , yang mewakili informasi baru yang bisa ditambahkan ke dalam memori model. Kedua komponen ini bekerja sama untuk memodifikasi dan memperbarui cell state yang akan digunakan dalam langkah berikutnya[41].

Terakhir, pada langkah ketiga dan keempat, cell state yang baru diperbarui digunakan untuk menghasilkan output[41]. Output gate  $o_t$  memutuskan bagian mana dari cell state yang akan digunakan untuk menghasilkan hidden state  $h_t$ , yang akhirnya menjadi output model[41]. Hidden state ini membawa informasi yang telah diproses, dan output ini selanjutnya dapat digunakan untuk prediksi atau diteruskan ke langkah waktu berikutnya[41]. Dengan demikian, LSTM mampu memproses urutan data dalam waktu panjang dengan memilih untuk mengingat atau melupakan informasi yang ada[41].

LSTM dilengkapi dengan unit memori yang disebut cell state. Cell state ini bertanggung jawab untuk menyimpan dan membawa informasi sepanjang urutan waktu. Struktur LSTM memiliki tiga jenis gerbang utama yang mengatur bagaimana informasi akan diproses, disimpan, dan dikeluarkan dari jaringan. Gerbang-gerbang ini membantu LSTM untuk memutuskan apa yang harus diingat, apa yang harus dilupakan, dan informasi mana yang perlu disalurkan ke langkah waktu berikutnya[42].

$$f_t = \sigma(W_f[h_{t-1}, x_t] + b_f)$$

Rumus 2. 3 Rumus Long Short Term Memory (LSTM)

Analisis sentimen adalah salah satu aplikasi yang paling umum dari LSTM, karena tugas ini sering kali melibatkan pemahaman konteks panjang dalam sebuah urutan teks. LSTM sangat cocok untuk ini karena dapat menangkap dependensi jangka panjang antara kata-kata dalam kalimat. Model LSTM menerima teks yang telah diproses, diubah menjadi urutan kata atau token, dan biasanya diubah menjadi representasi numerik (seperti vektor kata menggunakan Word2Vec, GloVe, atau embeddings lainnya).

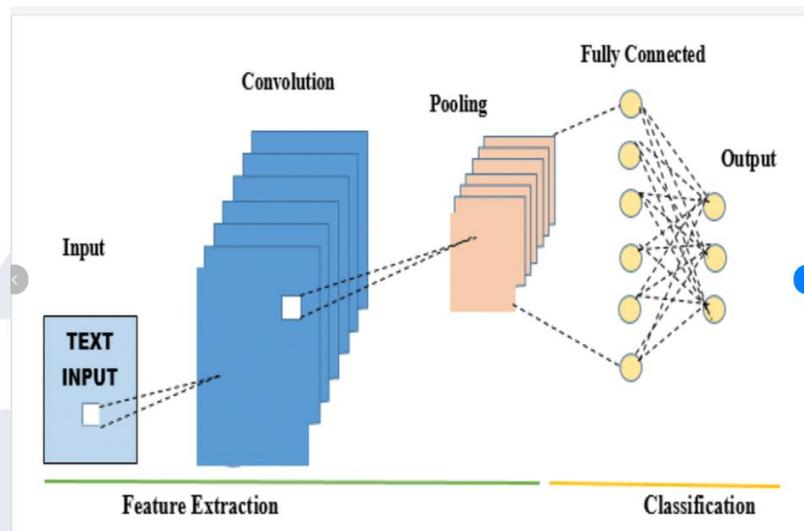
Melalui gerbang-gerbangnya, LSTM menyaring informasi yang relevan untuk menentukan sentimen, di mana kata-kata seperti "kecewa" dan "harapan" akan memiliki dampak signifikan dalam menentukan apakah sentimen secara keseluruhan positif atau negatif, meskipun berada dalam urutan kalimat yang panjang. Setelah memproses urutan kata, LSTM akan mengeluarkan prediksi sentimen (positif, dan negatif,) berdasarkan informasi yang telah diproses, dengan output sering kali berupa kelas sentimen atau skor probabilitas dari sentimen yang diprediks

### **2.3.9 Convolutional Neural Network (CNN)**

Convolutional Neural Network (CNN) merupakan salah satu arsitektur deep learning yang dirancang untuk mengekstrak fitur secara otomatis dari data input yang memiliki dimensi spasial, seperti gambar atau teks[54]. Awalnya dikembangkan untuk pemrosesan gambar digital, CNN telah terbukti sangat efektif dalam memproses data berbasis teks, terutama dalam tugas-tugas seperti klasifikasi dokumen, deteksi emosi, dan analisis sentimen[54]. Salah satu komponen utama CNN adalah lapisan konvolusi, yang menggunakan operasi matematis berupa konvolusi untuk mengekstrak fitur penting dari data input. Secara matematis, output dari lapisan konvolusi pada posisi tertentu dihitung dengan menjumlahkan hasil perkalian antara nilai-nilai input dalam sebuah jendela kecil (kernel atau filter) dengan bobot kernel tersebut, kemudian ditambah dengan bias.

Convolutional Neural Network (CNN) berfungsi dengan mengidentifikasi fitur-fitur lokal dari data masukan, yang dalam konteks teks berupa frasa atau rangkaian kata bermakna. Metode ini sangat efektif dalam menangkap pola-pola penting dalam kalimat yang berperan dalam menentukan sentimen positif, negatif, maupun netral.

Pada gambar 2.5 di bawah ini menggambarkan alur pemrosesan data teks menggunakan arsitektur Convolutional Neural Network (CNN) dalam tugas analisis sentimen[43]. CNN dalam konteks ini digunakan untuk mengekstraksi fitur penting dari teks masukan dan melakukan klasifikasi berdasarkan emosi atau opini yang terkandung di dalamnya[43].



Rumus 2. 4 Prosedur Alur Analisis Sentimen CNN[43].

Secara umum, proses kerja CNN dalam analisis sentimen terbagi menjadi dua tahap utama, yaitu ekstraksi fitur (feature extraction) dan klasifikasi (classification)[43].

### 1. Input

Proses dimulai dari teks yang telah diubah menjadi representasi vektor—biasanya dalam bentuk embedding seperti Word2Vec, GloVe, atau embedding dari transformer. Representasi ini mengubah setiap kata menjadi angka agar bisa diproses oleh jaringan saraf[43].

### 2. Convolutional Layer

Di tahap ini, beberapa filter konvolusional diterapkan ke urutan kata untuk mendeteksi pola lokal seperti frasa penting yang mengandung makna sentimen, misalnya "sangat baik", "mengecewakan", atau "tidak disarankan". Setiap filter bertindak seperti jendela kecil yang mengamati kata-kata secara bertahap[43].

### 3. Pooling Layer

Setelah fitur lokal terdeteksi, hasilnya diproses lebih lanjut melalui pooling, yang berfungsi untuk menyaring informasi paling menonjol dari setiap fitur[43]. Biasanya digunakan max pooling, yang memilih nilai

tertinggi untuk menjaga fitur paling relevan sambil mengurangi dimensi data.

#### 4. Fully Connected Layer

Fitur-fitur penting yang sudah disaring kemudian dikirim ke lapisan saraf penuh (fully connected layer). Di sinilah terjadi penggabungan informasi untuk membuat prediksi sentimen secara menyeluruh[43].

#### 5. Output (Klasifikasi Sentimen)

Akhirnya, CNN menghasilkan output berupa label sentimen, yang bisa berupa dua kelas (positif/negatif) atau tiga kelas (positif, negatif, netral), tergantung pada kebutuhan sistem[43].

### 2.3.10 Confusion Matrix

Confusion Matrix merupakan alat evaluasi yang sangat penting dalam menilai hasil analisis sentimen, karena untuk mengukur tingkat akurasi prediksi yang dihasilkan oleh algoritma klasifikasi. Alat ini bekerja dengan membandingkan hasil prediksi algoritma dengan data nyata yang ada. Sebagaimana terlihat pada Tabel 2.1, Confusion Matrix menyajikan visualisasi dan ringkasan kinerja algoritma dalam bentuk tabel yang memuat data aktual serta hasil klasifikasi. Dengan menggunakan Confusion Matrix dapat mengevaluasi seberapa efektif algoritma dalam mengklasifikasikan data secara tepat dan akurat [11].

Tabel 2. 2 Confusion Matrix

Confusion Matrix	Actual Data		
	Positive		Negative
Prediction Result	Positive	True Positive	False Negative
	Negative	False Positive	True Negative

Melalui tabel Confusion Matrix yang disajikan di atas dapat mengevaluasi sebagai berikut:

- a. **True Positive (TP)** menunjukkan jumlah data yang sebenarnya positif dan berhasil diprediksi sebagai positif oleh model.
- b. **False Positive (FP)** menggambarkan jumlah data yang sebenarnya negatif namun salah diklasifikasikan sebagai positif oleh model.
- c. **False Negative (FN)** mengacu pada jumlah data yang sebenarnya positif tetapi keliru diprediksi sebagai negatif oleh model.
- d. **True Negative (TN)** mencerminkan jumlah data yang sebenarnya negatif dan berhasil diprediksi sebagai negatif oleh model.

Menggunakan keempat data tersebut dapat mengestimasi berbagai nilai yang berguna untuk mengukur kinerja metode yang telah dikembangkan[44]. Dengan memanfaatkan keempat data tersebut, dapat menghitung sejumlah metrik yang bermanfaat untuk mengevaluasi kinerja metode yang telah disusun.

Setelah Confusion Matrix terbentuk, kinerja algoritma klasifikasi dapat dievaluasi melalui sejumlah parameter, termasuk akurasi, presisi, recall, dan skor F1.

#### 1. Accuracy

Nilai Accuracy mencerminkan tingkat keakuratan model yang telah dibentuk dan diukur sebagai perbandingan antara data yang berhasil diklasifikasikan dengan benar (TP+TN) terhadap seluruh jumlah data

U N I V E R S I T A S  
M U L T I M E D I A  
N U S A N T A R A

$(TP+TN+FP+FN)$ [45]. Berikut merupakan cara perhitungan akurasi dapat diperjelas dengan berikut:

$$\text{Accuracy} = \frac{TP + TN}{\text{Jumlah Data}}$$

Rumus 2. 5 Rumus Accuracy

## 2. Precision

Precision merupakan ukuran yang mengevaluasi sejauh mana model mampu mengenali kelas yang diminta dengan benar dalam perbandingan dengan semua hasil prediksi dari kelas tersebut. Dalam konteks ini, precision mencerminkan kemampuan model untuk memprediksi data aktual positif dari semua hasil prediksi yang dinyatakan sebagai kelas positif[45]. Rumus perhitungan precision merupakan sebagai berikut:

$$\text{Precision} = \frac{TP}{TP + FP}$$

Rumus 2. 6 Rumus Precision

## 3. Recall

Recall, yang juga dikenal sebagai "sensitivitas" atau "true positive rate," mengukur sejauh mana model dapat mengenali kelas yang diminta dari seluruh data aktual yang termasuk dalam kelas tersebut. Dalam konteks ini, recall mencerminkan kemampuan model dalam memprediksi data aktual positif dari total keseluruhan data aktual positif dalam dataset[45]. Perhitungan recall dapat dinyatakan dalam rumus berikut:

$$\text{Recall} = \frac{TP}{TP + FN}$$

Rumus 2. 7 Rumus Recall

#### 4. F1 score

F1-score, atau F-Measure, merupakan suatu metrik yang memberikan gambaran tentang keseimbangan antara precision dan recall, khususnya ketika terjadi ketidakseimbangan dalam kelas data[45]. Rumus untuk menghitung F1-score merupakan sebagai berikut:

$$F1 - score = 2 * \frac{Precision * Recall}{Precision + Recall}$$

Rumus 2. 8 Rumus F1 - Score

#### 2.3.11 Uji Friedman

Uji Friedman adalah metode statistik non-parametrik yang digunakan untuk membandingkan tiga atau lebih algoritma atau model pada data yang berulang atau berpasangan. Uji ini sangat berguna ketika data tidak memenuhi asumsi normalitas yang diperlukan dalam ANOVA berulang. Fungsi utamanya adalah menguji hipotesis nol yang menyatakan tidak ada perbedaan signifikan antara median performa dari beberapa kelompok atau perlakuan yang diuji.[46]

Prinsip kerja uji Friedman melibatkan pemberian peringkat terhadap hasil pengujian tiap model secara terpisah untuk setiap sampel data, lalu membandingkan peringkat tersebut untuk melihat konsistensi perbedaan performa. Statistik uji yang dihitung kemudian digunakan untuk memperoleh nilai p-value. Jika p-value kurang dari tingkat signifikansi (seperti 0,05), maka hipotesis nol ditolak, dan diartikan bahwa ada perbedaan performa yang signifikan secara statistik antar model[46].

Uji Friedman biasanya dilakukan dengan mempertimbangkan berbagai parameter evaluasi kinerja model, seperti akurasi, presisi, recall, dan F1-score. Parameter-parameter ini memberikan gambaran menyeluruh tentang performa model dalam berbagai aspek, mulai dari kemampuan klasifikasi yang benar hingga keseimbangan antara presisi dan recall. Dengan menggunakan uji Friedman terhadap metrik-metrik ini, peneliti dapat secara statistik membuktikan apakah ada perbedaan yang signifikan antar model atau algoritma yang diuji. Pendekatan ini sangat membantu dalam memilih model terbaik secara objektif berdasarkan bukti

data yang konsisten dan tidak bergantung pada asumsi distribusi data. Oleh karena itu, uji Friedman menjadi alat penting dalam evaluasi dan validasi performa berbagai model machine learning dalam penelitian[46].

## **2.4 Tools dan Software Penelitian**

### **2.4.1 Google Colaboratory / Google Colab**

Google Colaboratory, atau yang sering disebut sebagai Google Colab, merupakan sebuah platform komputasi berbasis cloud yang menyerupai Jupyter Notebook[47]. Google Colab memberikan kemampuan kepada pengguna untuk menjalankan kode Python serta menjalankan proyek-proyek terkait data science dan pemrograman tanpa harus menghadapi kerumitan dalam konfigurasi atau pengaturan lingkungan pengembangan pada komputer lokal mereka. Dengan Google Colab, pengguna memiliki akses ke lingkungan pengembangan Python yang sangat kuat dan lengkap melalui peramban web yang umumnya digunakan, seperti Google Chrome, Mozilla Firefox, atau Opera. Ini berarti tidak perlu lagi menghabiskan waktu berharga untuk menginstal dan mengonfigurasi perangkat lunak di komputer lokal. Salah satu keunggulan utama dari Google Colab merupakan ketersediaan *library* dan paket Python yang sangat beragam tanpa perlu melakukan instalasi manual[48]. Semua *library* yang sering digunakan dalam ilmu data dan pengembangan Python sudah terinstal secara default[48]. Ini merupakan hal yang sangat berguna bagi pemula yang ingin memulai perjalanan mereka dalam pemrograman Python, sekaligus memberikan kemudahan bagi para profesional yang ingin menjalankan proyek tanpa hambatan teknis[48]. Google Colab memudahkan proses data scraping dengan menyediakan lingkungan pengembangan Python berbasis cloud[49]. Platform ini memberikan akses cepat dan mudah ke berbagai perpustakaan dan alat yang sering digunakan dalam proses web scraping[49]. Dengan sumber daya cloud yang kuat, Google Colab mempermudah pengumpulan, analisis, dan penyimpanan data dari berbagai sumber secara efisien. Oleh karena itu, Google Colab merupakan pilihan yang sangat bijak untuk para pengembang, peneliti, atau siapa pun yang ingin meningkatkan keterampilan pemrograman Python tanpa harus repot dengan konfigurasi perangkat

lunak di komputer lokal mereka, sekaligus dengan akses yang mudah ke sumber daya komputasi yang dibutuhkan.

#### **2.4.2 Python**

Python, yang merupakan produk ciptaan Guido van Rossum, merupakan sebuah bahasa pemrograman tingkat tinggi yang menawarkan sejumlah keunggulan penting. Salah satunya merupakan fleksibilitas yang diberikan dalam pemrograman berorientasi objek (OOP), yang memungkinkan pengembang untuk membangun solusi perangkat lunak yang kuat. Python juga dikenal dengan kemampuannya beroperasi secara lintas platform, yang artinya kode yang ditulis dalam Python dapat berjalan di berbagai sistem operasi. Untuk memperluas fungsionalitasnya, Python memiliki lebih dari 100.000 paket yang tersedia dalam perpustakaan Python. Keberagaman paket ini memberikan pengembang alat yang kuat untuk mengembangkan model, aplikasi, dan perangkat lunak dengan beragam fungsi dan fitur. Kombinasi fleksibilitas, lintas platform, dan keberagaman paket membuat Python menjadi pilihan yang sangat kuat dan serbaguna untuk berbagai keperluan pengembangan perangkat lunak[50].

#### **2.4.3 Visual Studio Code (VS Code)**

Visual Studio Code (VS Code) adalah sebuah teks editor sumber terbuka (open-source) yang dikembangkan oleh Microsoft untuk membantu pengembang dalam menulis, mengedit, dan menjalankan kode secara efisien. Aplikasi ini ringan, cepat, dan mendukung berbagai bahasa pemrograman seperti Python, JavaScript, C++, dan banyak lagi. Dengan fitur-fitur unggulan seperti syntax highlighting, autocompletion berbasis IntelliSense, integrasi Git bawaan, debugger, serta terminal terintegrasi, VS Code menjadi pilihan populer di kalangan pengembang. Selain itu, aplikasi ini dapat diperluas dengan ribuan ekstensi melalui marketplacenya untuk menyesuaikan kebutuhan pengembangan proyek, baik itu membangun aplikasi web, API, maupun proyek berbasis data. Dukungan lintas platform (Windows, macOS, dan Linux) menjadikan VS Code fleksibel untuk digunakan oleh pengembang di berbagai lingkungan kerja.