BAB II

LANDASAN TEORI

2.1 Penelitian Terdahulu

Penelitian terdahulu digunakan sebagai dasar untuk melihat metode, objek, dan hasil penelitian yang berkaitan dengan analisis sentimen, ABSA, LLM-assisted labeling, model transformer, dan Semantic Network Analysis. Ringkasan penelitian terdahulu ditampilkan pada Tabel 2.1.

Tabel 2. 1 Penelitian Terdahulu

No	Judul dan Peneliti Artikel	Nama Jurnal	Metode	Hasil Penelitian	Research Gap
1	**Judul:** Analisis Sentimen Berbasis Aspek pada Ulasan Aplikasi Gojek **Peneliti:** Rahman, Pranatawijaya, dan Sari (2024) [13]	KONSTELASI: Konvergensi Teknologi dan Sistem Informasi	ABSA, LDA, BERT, dan distilbert-base-uncased-finetuned-sst-2-english	Akurasi sentimen 96,67%; aspek yang perlu diperbaiki: user experience, service, payment	Belum mengintegrasikan hasil ABSA ulasan Gojek dengan *Semantic Network Analysis* untuk memetakan isu dominan pada ulasan negatif.
2	**Judul:** Penerapan Naïve Bayes Classifier Untuk Pendukung Keputusan Berbasis Analisis	Jurnal Teknologi dan Sistem Komputer (JTSiskom)	TF-IDF, Smote, Multinomial Naïve Bayes	Algoritma Naïve Bayes mencapai Akurasi 86% dan F1-score 85% pada ulasan Gojek.	Masih berfokus pada sentimen umum (global) dan belum menganalisis sentimen berdasarkan tingkat aspek layanan (ABSA).

	Sentimen Ulasan Aplikasi Gojek **Peneliti:** Alita, Putra, dan Darmawan (2021) [14]				
3	**Judul:** ABSA of Indonesian Customer Reviews Using IndoBERT: Single-Sentence and Sentence-Pair Classificatio n Approaches **Peneliti:** Yulianti dan Nissa (2024) [15]	Bulletin of Electrical Engineering and Informatics	Tiga varian IndoBERT untuk ABSA	Fine-tuned IndoBERT meningkat 23,6% dibanding Word2Vec-CNN-XGBoost	Objek penelitian bukan Gojek dan belum membandingkan IndoBERT dengan mDeBERTa-v3.
4	**Judul:** Knowledge Distillation in Automated Annotation: Supervised Text Classificatio	Proceedings of NLP+CSS 2024	LLM-generated labels untuk supervised text classification	Label LLM sebanding dengan label manusia pada 14 tugas klasifikasi	Belum diterapkan secara khusus pada ABSA ulasan aplikasi Gojek berbahasa Indonesia.

	n with LLM-Generated Training Labels **Peneliti:** Pangakis dan Wolken (2024) [7]				
5	**Judul:** Instruct-DeBERTa: A Hybrid Approach for Aspect-based Sentiment Analysis on Textual Reviews **Peneliti:** Jayakody dkk. (2024) [16]	arXiv preprint	InstructABSA + DeBERTa-V3	Model Instruct-DeBERTa terbaik untuk tugas ABSA pada dataset SemEval	Belum berfokus pada ulasan aplikasi Gojek dan belum membandingkan DeBERTa-v3 dengan IndoBERT pada konteks bahasa Indonesia.
6	**Judul:** Aspect-Based Sentiment Analysis with LDA and IndoBERT Algorithm on Mental Health App: Riliv	Journal of Applied Informatics and Computing	*LDA + IndoBERT pada ABSA*	Akurasi IndoBERT 95% pada ulasan aplikasi berbahasa Indonesia	Objek penelitian berbeda dan belum menghubungkan ABSA dengan pemetaan isu dominan menggunakan SNA.

	Peneliti: Aryanti, Luthfiarta, dan Soeroso (2025) [17]				
7	**Judul:** Sentiment Analysis on Google Play Store Reviews to Measure User Perception of the Gojek Application Using CNN **Peneliti:** Anissa, Tania, dan Sari (2025) [18]	Journal of Applied Informatics and Computing	CNN dengan tiga rasio pembagian data	Akurasi tertinggi 84,29% pada klasifikasi tiga kelas	Masih menggunakan klasifikasi sentimen umum dan belum menerapkan ABSA berbasis *aspect category classification*.
8	**Judul:** Optimization of IndoBERT for Sentiment Analysis of FOMO on Social Media Through Fine-Tuning and Hybrid Labeling	Journal of Applied Informatics and Computing	Fine-tuning IndoBERT + hybrid labeling + hyperparameter tuning	Akurasi 94,50%, F1-score 94,52%; hybrid labeling meningkatkan performa model	Konteks data bukan ulasan aplikasi Gojek dan belum menggunakan SNA untuk melihat keterkaitan isu.

	Peneliti: Adhim dan Cahyono (2025) [19]				
9	**Judul:** Text Mining for Consumers' Sentiment Tendency and Strategies for Promoting Cross-Border E-Commerce Marketing Using Consumers' Online Review Data **Peneliti:** Liu, Chen, Pu, dan Jin (2025) [20]	Journal of Theoretical and Applied Electronic Commerce Research	SNA, LDA, dan LSTM	Keyword co-occurrence network berhasil memetakan isu dan distribusi sentiment	Konteks penelitian bukan aplikasi Gojek dan belum menghubungkan *network analysis* dengan ABSA berbasis model *transformer*.
10	**Judul:** Aspect-Based Sentiment Analysis of Access by KAI Application Reviews Using IndoBERT for Multi-	Jurnal Teknik Informatika (JUTIF)	Hybrid labeling + fine-tuned IndoBERT untuk multi-label ABSA	IndoBERT accuracy 0,928; F1-score 0,785 pada klasifikasi aspek	Objek penelitian berbeda, belum membandingkan IndoBERT dengan mDeBERTa-v3, dan belum menggunakan SNA.

	Label Classification Tasks **Peneliti:** Alfiana, Doewes, dan Widoyono (2026) [21]				
11	**Judul:** Knowledge Discovery in Sharia Mobile Banking Reviews Using Aspect-Based Sentiment Analysis and Machine Learning **Peneliti:** Ramadhani, Tania, dan Afrina (2026) [22]	Journal of Applied Informatics and Computing	ABSA + Naïve Bayes, SVM, dan Random Forest	SVM accuracy 0,95; F1-score 0,92; aspek features dominan negative	Objek penelitian berbeda dan belum menggunakan model *transformer* seperti IndoBERT dan mDeBERTa-v3.
12	**Judul:** IndoLEM and IndoBERT: A Benchmark Dataset and Pre-trained	Findings of the ACL: EMNLP	Pre-training Transformer (Arsitektur BERT)	IndoBERT mengungguli algoritma *machine learning* konvensional secara signifikan.	Berfokus pada arsitektur dasar; belum diaplikasikan spesifik untuk ABSA ulasan aplikasi *multiservis* dengan SNA.

	Language Model... **Peneliti:** Koto, Rahimi, Lau, dan Baldwin (2020)				
13	**Judul:** DeBERTaV 3: Improving DeBERTa using ELECTRA-Style Pre-Training... **Peneliti:** He, Gao, dan Chen (2021)	ICLR / arXiv preprint	RTD dan *disentangled attention*	Varian mDeBERTa-v3 mencapai performa *state-of-the-art* pada NLP multilingual.	Belum dikomparasikan secara *head-to-head* dengan IndoBERT pada ABSA bahasa Indonesia non-formal.
14	**Judul:** Qwen2.5 Technical Report **Peneliti:** Yang dkk. (Alibaba Group) (2024)	arXiv preprint	*Instruction-tuned* LLM	Model seri Qwen2.5 memiliki kemampuan instruksi terstruktur yang konsisten.	Belum diuji spesifik efisiensinya sebagai alat bantu anotasi ABSA (*LLM-assisted labeling*) secara lokal.

Berdasarkan Tabel 2.1, penelitian terdahulu menunjukkan bahwa ulasan pengguna aplikasi telah banyak digunakan sebagai sumber data dalam analisis sentimen dan *Aspect-Based Sentiment Analysis* (ABSA). Penelitian yang secara khusus menggunakan objek aplikasi Gojek telah dilakukan oleh Rahman, Pranatawijaya, dan Sari [13], Alita, Putra, dan Darmawan [14], serta Anissa, Tania, dan Sari [18]. Ketiga penelitian tersebut membuktikan bahwa ulasan aplikasi Gojek

sangat relevan untuk dianalisis karena memiliki volume data yang besar dan memuat opini langsung dari pengguna.

Namun, terdapat keterbatasan serius pada pendekatan yang digunakan dalam literatur terdahulu mengenai objek ini. Penelitian oleh Alita dkk. [14] serta Anissa dkk. [18] masih tertahan pada tingkat analisis sentimen umum (global). Pendekatan ini secara kritis dinilai tidak memadai untuk mengevaluasi aplikasi multiservis seperti Gojek, karena gagal menangkap fenomena polaritas ganda di mana seorang pengguna dapat memberikan sentimen positif pada satu aspek (misalnya, kelayakan mitra pengemudi) tetapi sekaligus memberikan sentimen negatif pada aspek lain (misalnya, galat pada aplikasi) dalam satu teks ulasan yang sama. Sementara itu, meskipun Rahman dkk. [13] telah menerapkan ABSA pada Gojek, penelitian tersebut belum mengintegrasikan hasil klasifikasinya ke dalam pemetaan isu dominan di tahap hilir.

Efektivitas model berbasis *transformer* dalam menangani karakteristik teks bahasa Indonesia telah dibuktikan oleh Yulianti dan Nissa [15], Aryanti, Luthfiarta, dan Soeroso [17], serta Adhim dan Cahyono [19]. Penelitian-penelitian tersebut menegaskan bahwa IndoBERT [12] memiliki keunggulan performa yang signifikan dibandingkan algoritma konvensional. Kendati demikian, kajian kritis terhadap literatur ini menunjukkan adanya celah pada komparasi arsitektur model. Sebagian besar penelitian cenderung bersifat menggunakan model tunggal atau variasi internal pada rumpun IndoBERT saja. Belum ditemukan evaluasi *head-to-head* yang membandingkan keunggulan leksikal IndoBERT sebagai model monolingual dengan kemampuan kontekstual mDeBERTa-v3 [13] sebagai model multilingual mutakhir pada konteks ulasan aplikasi Gojek. Kesenjangan komparasi ini krusial untuk menguji apakah spesifikasi bahasa (monolingual) lebih determinan dibandingkan keunggulan arsitektur ekstraksi fitur lintas bahasa.

Tantangan dalam penyediaan dataset berlabel juga menjadi perhatian penting. Pangakis dan Wolken [7] memberikan landasan empiris bahwa pelabelan otomatis menggunakan bantuan *Large Language Model* (LLM) memiliki keandalan yang

sebanding dengan anotasi manusia pada tugas klasifikasi teks. Temuan ini relevan dengan penerapan model *instruction-tuned* Qwen2.5-0.5B-Instruct [14] melalui pendekatan *LLM-assisted labeling* dalam penelitian ini. Namun, celah yang masih menyelimuti penelitian pelabelan otomatis adalah potensi bias dan ketidaksesuaian label pada anotasi tingkat aspek yang ketat. Oleh karena itu, penelitian ini tidak sekadar menggunakan hasil LLM secara mentah, melainkan menerapkan mekanisme kontrol kualitas melalui validasi manual terhadap sampel data sebelum dataset digunakan pada tahap pemodelan.

Di sisi lain, eksplorasi ABSA pada domain aplikasi layanan publik lainnya [17], [21], [22] serta penerapan arsitektur DeBERTa-v3 [16] dan analisis jaringan [20] telah menunjukkan potensi pengembangan yang luas. Walau demikian, seluruh penelitian tersebut masih bergerak secara terfragmentasi. Belum ada penelitian yang mengintegrasikan secara utuh ketajaman klasifikasi ABSA berbasis *transformer* dengan kedalaman diagnostik masalah menggunakan *Semantic Network Analysis* (SNA) pada aplikasi multiservis.

Berdasarkan analisis kritis terhadap literatur terdahulu, celah penelitian (*research gap*) utama dalam penelitian ini terletak pada ketiadaan alur kerja yang terintegrasi secara *end-to-end*. Penelitian ini tidak hanya melakukan klasifikasi sentimen umum, tetapi membangun dataset ABSA melalui pendekatan *LLM-assisted labeling* dengan skema *aspect category classification*. Penelitian ini juga mengomparasikan secara langsung performa IndoBERT dan mDeBERTa-v3 pada tugas klasifikasi sentimen berbasis aspek berbahasa Indonesia. Lebih jauh, penelitian ini menghubungkan hasil klasifikasi ABSA ke dalam *Semantic Network Analysis* berbasis *keyword co-occurrence* untuk memetakan keterkaitan kata dan isu dominan pada ulasan bersentimen positif dan negatif pengguna aplikasi Gojek. Sentimen netral secara metodologis tidak diikutsertakan dalam SNA karena tidak menunjukkan arah opini yang kuat. Dengan demikian, penelitian ini diharapkan memberikan kontribusi yang komprehensif, tidak hanya pada peningkatan performa model, tetapi juga pada pemahaman diagnostik atas akar kepuasan dan keluhan pengguna.

2.2 Teori yang berkaitan

Teori yang berkaitan digunakan sebagai dasar konseptual dalam penelitian ini. Pembahasan teori disusun berdasarkan tahapan penelitian. Tahapan tersebut meliputi pengumpulan ulasan dari Google Play Store, pengolahan teks, pelabelan berbasis aspek, pemodelan transformer, evaluasi model, dan pemetaan isu menggunakan Semantic Network Analysis.

2.2.1 Google Play Store

Google Play Store merupakan *platform* distribusi aplikasi digital untuk perangkat Android. *Platform* ini menyediakan berbagai aplikasi yang dapat diunduh, dipasang, diperbarui, dan diberi penilaian oleh pengguna. Selain itu, Google Play Store juga menyediakan fitur *rating* dan ulasan pengguna pada halaman aplikasi [2]. Fitur tersebut memungkinkan pengguna memberikan penilaian terhadap pengalaman mereka setelah menggunakan suatu aplikasi.

Dalam penelitian berbasis teks, ulasan pada Google Play Store dapat digunakan sebagai sumber data karena memuat opini pengguna secara langsung. Ulasan tersebut dapat berisi komentar, keluhan, apresiasi, kritik, dan saran terhadap aplikasi yang digunakan. Pada konteks aplikasi layanan digital seperti Gojek, ulasan pengguna dapat merepresentasikan pengalaman terhadap fitur aplikasi, kualitas layanan, mitra pengemudi, tarif, pembayaran, promosi, dan pengalaman penggunaan secara umum [2], [4].

Dalam penelitian ini, Google Play Store digunakan sebagai sumber data utama karena menyediakan ulasan pengguna aplikasi Gojek dalam jumlah besar. Data ulasan tersebut kemudian digunakan sebagai dasar untuk melakukan analisis sentimen berbasis aspek. Oleh karena itu, Google Play Store berperan sebagai sumber data penelitian, bukan sebagai algoritma atau metode komputasi.

Subbab ini tidak memuat rumus perhitungan karena Google Play Store bukan metode analisis atau algoritma. Rumus perhitungan dijelaskan pada

subbab yang berkaitan dengan proses komputasi, seperti metrik evaluasi model dan pengukuran sentralitas pada *Semantic Network Analysis*.

2.2.2 Text Mining

Text mining merupakan proses pengambilan informasi dari data teks tidak terstruktur. Data teks seperti ulasan pengguna, komentar, dan opini tidak dapat langsung dianalisis tanpa proses pengolahan. Oleh karena itu, *text mining* digunakan untuk mengubah teks bebas menjadi informasi yang dapat diproses secara komputasional [23].

Dalam penelitian ini, *text mining* digunakan untuk mengolah ulasan pengguna aplikasi Gojek. Proses tersebut mencakup pengumpulan data ulasan, pembersihan data, *preprocessing* teks, pelabelan aspek dan sentimen, pemodelan klasifikasi, serta pemetaan kata melalui *keyword co-occurrence*. Tahapan tersebut diperlukan agar ulasan pengguna yang awalnya berbentuk teks tidak terstruktur dapat diubah menjadi data yang lebih terstruktur dan siap dianalisis [4], [23].

Secara umum, proses *text mining* dapat dipahami sebagai proses transformasi kumpulan dokumen teks menjadi representasi yang lebih terstruktur. Kumpulan dokumen teks dapat dinyatakan sebagai berikut.

$$D=\{d1,d2,d3,\ldots,dn\}$$

2. 1 Kumpulan dokumen teks

Keterangan:

(D) = kumpulan dokumen teks dalam dataset.

(d_i) = dokumen ke-(i) dalam dataset.

(n) = jumlah seluruh dokumen teks.

Dalam penelitian ini, setiap (d_i) merepresentasikan satu ulasan pengguna aplikasi Gojek.

$$d_i' = f(d_i)$$

2. 2 Rumus Pengolahan Teks

Keterangan:

(d_i) = teks ulasan awal sebelum diproses.

(f) = fungsi pengolahan teks yang mencakup pembersihan data dan *preprocessing*.

(d_i') = teks ulasan hasil pengolahan.

Dalam penelitian ini, fungsi (f) mencakup proses seperti *cleaning*, *case folding*, normalisasi kata, tokenisasi, penghapusan kata tidak relevan, dan penyusunan teks akhir.

Text mining menjadi dasar awal dalam penelitian ini karena ulasan pengguna aplikasi Gojek memiliki karakteristik tidak terstruktur. Satu ulasan dapat memuat keluhan, apresiasi, saran, atau opini terhadap beberapa aspek layanan sekaligus. Oleh karena itu, *text mining* diperlukan sebelum data masuk ke tahap *Aspect-Based Sentiment Analysis*, pemodelan menggunakan IndoBERT dan mDeBERTa-v3, serta *Semantic Network Analysis* [4], [20].

2.2.3 Natural Language Processing

Natural Language Processing atau NLP merupakan bidang dalam kecerdasan buatan yang berfokus pada pemrosesan, pemahaman, dan analisis bahasa manusia oleh komputer. NLP digunakan agar data teks dalam bentuk bahasa alami dapat diproses secara sistematis oleh mesin. Pada data ulasan pengguna, NLP berperan penting karena teks ulasan masih berbentuk tidak terstruktur dan perlu diolah terlebih dahulu sebelum digunakan dalam proses analisis [25].

Dalam penelitian ini, NLP digunakan pada beberapa tahap analisis, yaitu *preprocessing* teks, pelabelan berbantuan *Large Language Model*, klasifikasi sentimen berbasis aspek, serta pemetaan hubungan kata menggunakan *Semantic*

Network Analysis. NLP menjadi dasar teknis dalam proses *Aspect-Based Sentiment Analysis* karena model perlu memahami teks ulasan, mengenali aspek layanan yang dibahas, dan menentukan sentimen pada aspek tersebut [6], [25].

Secara umum, proses NLP dapat dipahami sebagai proses mengubah teks hasil pengolahan menjadi representasi yang dapat digunakan oleh model. Proses tersebut dapat dinyatakan sebagai berikut.

$$x_i = g(d_i')$$

2. 3 Rumus Proses NLP

Keterangan:

(d_i') = teks ulasan hasil pengolahan.

(g) = fungsi representasi teks dalam proses NLP.

(x_i) = representasi teks ke-(i) yang dapat diproses oleh model.

Dalam konteks model berbasis *transformer*, teks yang telah diproses akan diubah menjadi token dan representasi numerik agar dapat dipahami oleh model. Representasi tersebut memungkinkan model mengenali pola bahasa, hubungan antarkata, dan konteks kalimat. Kemampuan ini penting dalam penelitian karena satu ulasan pengguna aplikasi Gojek dapat memuat beberapa opini terhadap aspek layanan yang berbeda.

NLP juga mendukung proses pemodelan berbasis *transformer*. Model seperti IndoBERT dan mDeBERTa-v3 memanfaatkan representasi kontekstual untuk memahami hubungan antarkata dalam teks. Representasi tersebut digunakan agar model dapat mengenali pola sentimen pada ulasan pengguna berdasarkan konteks kalimat dan aspek layanan yang dianalisis [9], [11].

2.2.4 Sentiment Analysis

Sentiment analysis merupakan teknik dalam *Natural Language Processing* yang digunakan untuk mengidentifikasi opini, sikap, atau polaritas emosi dalam teks. Teknik ini umumnya digunakan untuk mengelompokkan teks ke dalam kelas sentimen tertentu, seperti positif, negatif, atau netral. Pada data ulasan aplikasi, *sentiment analysis* dapat digunakan untuk mengetahui kecenderungan penilaian pengguna terhadap suatu layanan, fitur, atau pengalaman penggunaan aplikasi [5].

Dalam penelitian ini, *sentiment analysis* digunakan untuk menganalisis opini pengguna terhadap ulasan aplikasi Gojek. Ulasan pengguna dapat memuat apresiasi, keluhan, kritik, maupun saran terhadap layanan yang digunakan. Melalui *sentiment analysis*, opini tersebut dapat diubah menjadi informasi terstruktur dalam bentuk label sentimen. Kelas sentimen yang digunakan dalam penelitian ini terdiri atas positif, negatif, dan netral.

Secara umum, proses *sentiment analysis* dapat dinyatakan sebagai proses pemetaan teks ke dalam kelas sentimen. Proses tersebut dapat dituliskan sebagai berikut.

$$y_i = h(x_i)$$

$$y_i \in \{\text{positif}, \text{negatif}, \text{netral}\}$$

2. 4 Rumus Proses Sentiment Analysis

Keterangan:

(x_i) = representasi teks ulasan ke-(i).

(h) = fungsi klasifikasi sentimen.

(y_i) = label sentimen hasil prediksi untuk ulasan ke-(i).

({\text{positif}, \text{negatif}, \text{netral}}) = himpunan kelas sentimen yang digunakan dalam penelitian.

Pada tingkat umum, *sentiment analysis* dapat dilakukan pada level dokumen atau kalimat. Analisis pada level dokumen memberikan satu label sentimen untuk satu teks secara keseluruhan. Namun, pendekatan tersebut belum cukup untuk menganalisis ulasan aplikasi Gojek karena satu ulasan dapat memuat beberapa opini terhadap aspek layanan yang berbeda. Misalnya, pengguna dapat memberikan penilaian positif terhadap promo, tetapi memberikan keluhan terhadap aplikasi atau driver dalam ulasan yang sama.

Oleh karena itu, penelitian ini tidak hanya menggunakan *sentiment analysis* secara umum, tetapi menerapkannya dalam pendekatan *Aspect-Based Sentiment Analysis*. Pendekatan tersebut digunakan agar sentimen dapat dikaitkan dengan aspek layanan tertentu, seperti Aplikasi, Layanan, Driver, Tarif & Harga, GoPay & Pembayaran, serta Promo & Iklan. Dengan demikian, hasil analisis tidak hanya menunjukkan polaritas sentimen secara umum, tetapi juga menunjukkan aspek layanan yang menjadi sumber sentimen pengguna [4], [6].

2.2.5 Aspect-Based Sentiment Analysis

Aspect-Based Sentiment Analysis atau ABSA merupakan pengembangan dari *sentiment analysis* yang bertujuan untuk mengidentifikasi sentimen terhadap aspek tertentu dalam suatu teks. Berbeda dengan *sentiment analysis* umum yang biasanya memberikan satu label sentimen untuk satu dokumen atau kalimat, ABSA memberikan analisis yang lebih rinci karena sentimen dapat dikaitkan dengan aspek yang dibahas dalam teks [6].

ABSA relevan digunakan pada ulasan aplikasi multiservice seperti Gojek karena satu ulasan dapat memuat opini terhadap beberapa aspek layanan sekaligus. Pengguna dapat membahas pengalaman terhadap Aplikasi, Layanan, Driver, Tarif & Harga, GoPay & Pembayaran, serta Promo & Iklan dalam satu teks ulasan. Oleh karena itu, analisis sentimen perlu dilakukan pada tingkat aspek agar sumber kepuasan atau keluhan pengguna dapat diketahui secara lebih spesifik [4], [6].

Secara umum, ABSA dapat dinyatakan sebagai proses pemetaan teks ulasan ke dalam pasangan aspek dan sentimen. Proses tersebut dapat dituliskan sebagai berikut.

$$ABSA(d_i) = \{(a_1, s_1), (a_2, s_2), \dots, (a_k, s_k)\}$$

2. 5 Rumus Proses ABSA

Keterangan:

(d_i) = dokumen atau ulasan ke-(i).

(a_k) = aspek ke-(k) yang teridentifikasi atau diklasifikasikan dalam ulasan.

(s_k) = sentimen terhadap aspek ke-(k).

(k) = jumlah aspek yang muncul atau relevan pada satu ulasan.

Dalam penelitian ini, setiap aspek diberi label sentimen positif, negatif, atau netral. Dengan demikian, satu ulasan dapat menghasilkan lebih dari satu pasangan aspek dan sentimen apabila ulasan tersebut membahas lebih dari satu aspek layanan. Proses ini membuat hasil analisis lebih rinci dibandingkan analisis sentimen umum.

Dalam implementasinya, ABSA dapat dilakukan melalui beberapa bentuk tugas. *Aspect extraction* digunakan untuk menemukan aspek atau entitas yang muncul secara langsung dalam teks. Sementara itu, *aspect category classification* digunakan ketika kategori aspek telah ditentukan sebelumnya, lalu teks diklasifikasikan ke dalam kategori aspek tersebut. Penelitian ini menggunakan pendekatan *aspect category classification* karena aspek layanan Gojek telah ditetapkan sejak awal sesuai ruang lingkup penelitian.

Dengan pendekatan tersebut, fokus penelitian bukan pada pencarian aspek baru secara otomatis dari teks ulasan, melainkan pada pemberian label aspek dan sentimen berdasarkan kategori yang telah ditentukan. Keenam aspek yang digunakan dalam penelitian ini adalah Aplikasi, Layanan, Driver, Tarif & Harga, GoPay & Pembayaran, serta Promo & Iklan. Pendekatan ABSA

digunakan agar hasil analisis dapat menunjukkan hubungan antara opini pengguna, aspek layanan, dan arah sentimen secara lebih terstruktur [6], [15].

2.2.6 Aspect Category Classification

Aspect category classification merupakan salah satu pendekatan dalam *Aspect-Based Sentiment Analysis* yang digunakan untuk mengklasifikasikan teks ke dalam kategori aspek yang telah ditentukan sebelumnya. Berbeda dengan *aspect extraction* yang bertujuan menemukan aspek baru secara langsung dari teks, *aspect category classification* bekerja berdasarkan daftar kategori aspek yang sudah disusun sebelum proses analisis dilakukan. Dengan demikian, proses analisis lebih terarah karena setiap ulasan diklasifikasikan berdasarkan kategori aspek yang sesuai dengan ruang lingkup penelitian.

Dalam penelitian ini, *aspect category classification* digunakan karena aspek layanan aplikasi Gojek telah ditentukan sejak awal. Aspek tersebut terdiri atas Aplikasi, Layanan, Driver, Tarif & Harga, GoPay & Pembayaran, serta Promo & Iklan. Penentuan kategori aspek dilakukan agar opini pengguna dapat dianalisis berdasarkan bagian layanan yang relevan dengan aplikasi Gojek. Pendekatan ini juga membantu menjaga konsistensi proses pelabelan karena model tidak diminta untuk menemukan aspek baru, melainkan menentukan apakah suatu ulasan berkaitan dengan kategori aspek yang telah disediakan.

Secara umum, proses *aspect category classification* dapat dinyatakan sebagai pemetaan teks ulasan ke dalam satu atau lebih kategori aspek. Proses tersebut dapat dituliskan sebagai berikut.

$$A_i = c(d_i)$$

2. 6 Rumus Proses Aspect Category Classification

Keterangan:

(d_i) = dokumen atau ulasan ke-(i).

(c) = fungsi klasifikasi aspek.

(A_i) = himpunan aspek yang diklasifikasikan dari ulasan ke-(i).

Himpunan kategori aspek dalam penelitian ini dapat dinyatakan sebagai berikut.

$$A = \{\text{Aplikasi}\},\text{Layanan}\},\text{Driver}\},\ \text{Tarif \& Harga}\},\text{GoPay \& Pembayaran}\},\ \text{Promo \& Iklan}\}\}$$

$$(d_i, a_j) \rightarrow s_j$$

2. 7 Rumus Himpunan Kategori Aspek

Keterangan:

(A) = himpunan kategori aspek yang digunakan dalam penelitian.

Setelah aspek diklasifikasikan, setiap aspek kemudian diberi label sentimen. Dengan demikian, hasil akhir analisis tidak hanya berupa kategori aspek, tetapi berupa pasangan aspek dan sentimen. Proses tersebut dapat dinyatakan sebagai berikut.

$$(d_i, a_j) \rightarrow s_j$$

2. 8 Rumus proses Label Sentiment

Keterangan:

(d_i) = ulasan ke-(i).

(a_j) = aspek ke-(j) yang termasuk dalam kategori aspek penelitian.

(s_j) = sentimen terhadap aspek ke-(j), yaitu positif, negatif, atau netral.

Pendekatan aspect category classification relevan digunakan dalam penelitian ini karena ulasan aplikasi Gojek dapat memuat lebih dari satu aspek layanan. Misalnya, satu ulasan dapat membahas kendala aplikasi, pengalaman dengan driver, dan masalah pembayaran secara bersamaan. Dengan pendekatan ini, setiap aspek yang dibahas dalam ulasan dapat diberi label sentimen secara terpisah. Oleh karena itu, hasil analisis menjadi lebih rinci dibandingkan

analisis sentimen umum yang hanya memberikan satu label sentimen untuk keseluruhan teks.

2.2.7 Definisi Aspek Layanan

Definisi aspek layanan digunakan sebagai acuan dalam proses pelabelan aspek pada penelitian ini. Karena penelitian menggunakan pendekatan *Aspect-Based Sentiment Analysis* berbasis *aspect category classification*, kategori aspek telah ditentukan sebelum proses pelabelan dilakukan. Penentuan aspek bertujuan untuk membatasi ruang lingkup analisis agar opini pengguna dapat diklasifikasikan secara konsisten ke dalam kategori layanan yang relevan dengan aplikasi Gojek.

Aspek layanan yang digunakan dalam penelitian ini terdiri atas Aplikasi, Layanan, Driver, Tarif & Harga, GoPay & Pembayaran, serta Promo & Iklan. Keenam aspek tersebut dipilih karena merepresentasikan komponen utama pengalaman pengguna dalam menggunakan aplikasi Gojek. Setiap aspek memiliki ruang lingkup yang berbeda agar proses pelabelan dapat dilakukan secara lebih terarah.

Aspek Aplikasi mengacu pada opini pengguna yang berkaitan dengan performa, fungsi, dan pengalaman penggunaan aplikasi Gojek. Aspek ini mencakup stabilitas aplikasi, kemudahan penggunaan, tampilan antarmuka, proses pemesanan, aplikasi lambat, aplikasi keluar sendiri, kendala sistem, kesalahan teknis, serta masalah lain yang berkaitan langsung dengan penggunaan aplikasi.

Aspek Layanan mengacu pada opini pengguna terhadap kualitas layanan Gojek secara umum. Aspek ini mencakup pengalaman menggunakan layanan, kecepatan layanan, kenyamanan, respons layanan, kepuasan terhadap proses layanan, serta keluhan umum yang tidak secara spesifik mengarah pada aplikasi, driver, tarif, pembayaran, atau promosi.

Aspek Driver mengacu pada opini pengguna yang berkaitan dengan mitra pengemudi. Aspek ini mencakup perilaku driver, komunikasi dengan pengguna,

ketepatan penjemputan, kesesuaian lokasi, pembatalan pesanan, sikap dalam memberikan layanan, serta pengalaman pengguna selama berinteraksi dengan driver.

Aspek Tarif & Harga mengacu pada opini pengguna terkait biaya layanan yang dikenakan dalam aplikasi Gojek. Aspek ini mencakup harga perjalanan, ongkos kirim, biaya layanan, perubahan tarif, perbandingan harga dengan layanan lain, serta persepsi pengguna terhadap mahal atau tidaknya biaya yang ditampilkan dalam aplikasi.

Aspek GoPay & Pembayaran mengacu pada opini pengguna yang berkaitan dengan metode pembayaran dan dompet digital dalam aplikasi Gojek. Aspek ini mencakup penggunaan GoPay, saldo, transaksi pembayaran, kegagalan pembayaran, pengembalian dana, potongan saldo, serta kendala dalam proses pembayaran digital.

Aspek Promo & Iklan mengacu pada opini pengguna yang berkaitan dengan promosi, voucher, diskon, iklan, dan penawaran yang ditampilkan dalam aplikasi Gojek. Aspek ini mencakup ketersediaan promo, kesesuaian promo, kendala penggunaan voucher, informasi iklan, serta persepsi pengguna terhadap manfaat promosi yang diberikan.

Secara umum, hubungan antara ulasan pengguna dan kategori aspek layanan dapat dinyatakan sebagai berikut.

$$a_j \in A$$

2. 9 Rumus Hubungan Ulasan Pengguna dan Kategori Aspek Layanan

Keterangan:

(a_j) = aspek ke-(j) yang diberikan pada ulasan pengguna.

(A) = himpunan kategori aspek layanan yang digunakan dalam penelitian.

Himpunan kategori aspek layanan dalam penelitian ini dapat dinyatakan sebagai berikut.

$$A = \{\text{Aplikasi},\text{Layanan},\text{Driver},\ \text{Tarif \& Harga},\text{GoPay \& Pembayaran},\ \text{Promo \& Iklan}\}$$

2. 10 Rumus Himpunan Kategori Aspek Layanan

Dengan adanya definisi aspek layanan tersebut, proses pelabelan dapat dilakukan secara lebih konsisten. Definisi ini juga digunakan sebagai acuan dalam penyusunan *prompt* pada tahap *LLM-assisted labeling* agar model memberikan label sesuai kategori aspek yang telah ditentukan. Selain itu, definisi aspek membantu memastikan bahwa hasil analisis sentimen dapat diinterpretasikan berdasarkan ruang lingkup aspek yang sama.

2.2.8 *Preprocessing* Teks

Preprocessing teks merupakan tahap pembersihan dan penyiapan data teks sebelum digunakan dalam proses analisis. Data ulasan pengguna biasanya memiliki karakteristik tidak terstruktur, seperti penggunaan huruf kapital yang tidak konsisten, simbol, angka, emoji, singkatan, kata tidak baku, kesalahan penulisan, serta elemen teks lain yang tidak relevan. Kondisi tersebut dapat menjadi *noise* yang memengaruhi kualitas data dan hasil analisis teks [4].

Dalam penelitian ini, *preprocessing* dilakukan untuk menyiapkan ulasan pengguna aplikasi Gojek sebelum masuk ke tahap *LLM-assisted labeling*, pemodelan, dan *Semantic Network Analysis*. Tahapan yang dilakukan meliputi pembersihan karakter, *case folding*, normalisasi kata, tokenisasi, penghapusan kata tidak relevan, dan penyusunan teks akhir. Proses ini dilakukan agar data ulasan menjadi lebih terstandar dan dapat dianalisis secara lebih konsisten [4], [20].

Secara umum, proses *preprocessing* dapat dinyatakan sebagai fungsi transformasi dari teks ulasan awal menjadi teks hasil pengolahan. Proses tersebut dapat dituliskan sebagai berikut.

$$d_i' = p(d_i)$$

2. 11 Rumus Proses Preprocessing

Keterangan:

(d_i) = teks ulasan ke-(i) sebelum dilakukan *preprocessing*.

(p) = fungsi *preprocessing* teks.

(d_i') = teks ulasan ke-(i) setelah dilakukan *preprocessing*.

Pada penelitian ini, fungsi *preprocessing* mencakup beberapa tahap pengolahan teks. Tahapan tersebut dapat dinyatakan sebagai berikut.

$$p(d_i) = t\left(r\left(n\left(c(d_i)\right)\right)\right)$$

2. 12 Rumus Fungsi Preprocessing

Keterangan:

(c) = proses *cleaning* atau pembersihan karakter tidak relevan.

(n) = proses normalisasi kata.

(r) = proses penghapusan kata tidak relevan.

(t) = proses tokenisasi atau penyusunan teks akhir.

Proses *preprocessing* juga digunakan untuk memastikan bahwa teks ulasan memiliki informasi yang cukup untuk dianalisis. Ulasan yang menjadi kosong atau terlalu pendek setelah proses pembersihan dapat dikeluarkan dari dataset. Langkah ini dilakukan agar data yang digunakan pada tahap pelabelan dan pemodelan tetap layak digunakan [36].

Pada penelitian ini, *preprocessing* tidak dilakukan secara berlebihan. Model *transformer* seperti IndoBERT dan mDeBERTa-v3 masih membutuhkan konteks bahasa dari teks ulasan. Oleh karena itu, *preprocessing* difokuskan

pada pengurangan *noise* tanpa menghilangkan makna utama dalam ulasan. Pendekatan ini digunakan agar teks tetap mempertahankan konteks opini pengguna terhadap aspek layanan yang dianalisis [9], [11].

2.2.9 Large Language Model

Large Language Model atau LLM merupakan model bahasa berskala besar yang dilatih menggunakan korpus teks dalam jumlah besar. Model ini dirancang untuk memahami konteks bahasa, mengikuti instruksi, menghasilkan teks, mengklasifikasikan informasi, serta membantu berbagai tugas dalam *Natural Language Processing*. Kemampuan tersebut membuat LLM dapat digunakan dalam proses pengolahan teks, termasuk untuk membantu anotasi atau pelabelan data [27].

Dalam konteks penelitian ini, LLM digunakan sebagai alat bantu anotasi untuk memberikan label aspek dan sentimen pada ulasan pengguna aplikasi Gojek. LLM tidak digunakan sebagai model klasifikasi utama, melainkan digunakan pada tahap pembentukan dataset awal melalui proses pelabelan berbantuan. Model klasifikasi akhir dalam penelitian ini tetap menggunakan IndoBERT dan mDeBERTa-v3 yang dilatih menggunakan dataset hasil pelabelan.

Penggunaan LLM dalam pelabelan dilakukan melalui *prompt* terstruktur. *Prompt* tersebut berisi daftar aspek layanan, aturan pelabelan, kelas sentimen, dan format keluaran. Dengan adanya *prompt* terstruktur, LLM diarahkan untuk menghasilkan label sesuai dengan kategori aspek yang telah ditentukan. Pendekatan ini sesuai dengan penelitian yang menggunakan skema *aspect category classification*, karena model tidak diminta untuk menemukan aspek baru secara bebas, tetapi memilih kategori aspek yang sudah tersedia.

Secara umum, proses pelabelan berbantuan LLM dapat dinyatakan sebagai fungsi pemetaan teks dan instruksi ke label aspek serta sentimen. Proses tersebut dapat dituliskan sebagai berikut.

$$L_i = M(d_i, P)$$

2. 13 Rumus Proses Pelabelan LLM

Keterangan:

(d_i) = ulasan pengguna ke-(i).

(P) = *prompt* atau instruksi pelabelan.

(M) = model LLM yang digunakan untuk membantu proses anotasi.

(L_i) = hasil label untuk ulasan ke-(i), berupa pasangan aspek dan sentimen.

Hasil label yang dihasilkan LLM dapat dinyatakan sebagai berikut.

$$L_i = \{(a_1, s_1), (a_2, s_2), \dots, (a_k, s_k)\}$$

2. 14 Rumus Hasil Label LLM

Keterangan:

(a_k) = aspek ke-(k) yang diberikan pada ulasan.

(s_k) = sentimen terhadap aspek ke-(k).

(k) = jumlah aspek yang relevan dalam satu ulasan.

Pada penelitian ini, model LLM yang digunakan adalah Qwen2.5-0.5B-Instruct. Model tersebut dipilih karena dapat dijalankan secara lokal, memiliki ukuran yang relatif kecil, dan termasuk model *instruction-tuned* sehingga dapat mengikuti instruksi berbasis *prompt*. Pemilihan model ini juga mempertimbangkan efisiensi komputasi dan kebutuhan pelabelan data dalam jumlah besar tanpa bergantung sepenuhnya pada layanan API eksternal.

Meskipun demikian, hasil pelabelan dari LLM tetap memiliki keterbatasan. Label yang dihasilkan dapat mengandung kesalahan, bias, atau ketidaksesuaian dengan konteks ulasan. Oleh karena itu, hasil pelabelan LLM tidak langsung digunakan tanpa pemeriksaan. Penelitian ini melakukan validasi dan audit terhadap hasil pelabelan sebelum dataset digunakan pada tahap pemodelan.

Dengan demikian, LLM berperan sebagai alat bantu pembentukan dataset, bukan sebagai pengganti penuh anotasi manusia atau model klasifikasi akhir.

2.2.10 LLM-Assisted Labeling

LLM-assisted labeling merupakan pendekatan pelabelan data yang memanfaatkan *Large Language Model* sebagai alat bantu anotasi. Pendekatan ini digunakan untuk membantu menghasilkan label awal pada data teks dalam jumlah besar. Pada tugas *Aspect-Based Sentiment Analysis*, proses pelabelan menjadi lebih kompleks karena setiap ulasan tidak hanya perlu diberi label sentimen, tetapi juga perlu dikaitkan dengan aspek layanan tertentu [7].

Dalam penelitian ini, *LLM-assisted labeling* digunakan untuk memberikan label aspek dan sentimen pada ulasan pengguna aplikasi Gojek. Proses pelabelan dilakukan menggunakan *prompt* terstruktur yang berisi daftar aspek layanan, aturan pemberian label, kelas sentimen, dan format keluaran. Daftar aspek yang digunakan terdiri atas Aplikasi, Layanan, Driver, Tarif & Harga, GoPay & Pembayaran, serta Promo & Iklan. Dengan demikian, proses pelabelan dilakukan berdasarkan skema *aspect category classification*, bukan *aspect extraction*.

Secara umum, proses *LLM-assisted labeling* dapat dinyatakan sebagai proses pemetaan ulasan dan instruksi pelabelan menjadi label aspek dan sentimen. Proses tersebut dapat dituliskan sebagai berikut.

$$\{L\}_i = M(d_i, P)$$

2. 15 Rumus Proses LLM-assisted labeling

Keterangan:

(d_i) = ulasan pengguna ke-(i).

(P) = *prompt* atau instruksi pelabelan.

(M) = model LLM yang digunakan untuk membantu proses pelabelan.

(\hat{L}_i) = label hasil prediksi LLM untuk ulasan ke-(i).

Label hasil LLM dapat dinyatakan sebagai pasangan aspek dan sentimen sebagai berikut.

$$\{L\}_i = \{(a_1, \widehat{\ }\{s\}_1), (a_2, \widehat{\ }\{s\}_2), \dots, (a_k, \widehat{\ }\{s\}_k)\}$$

2. 16 Rumus Label Hasil LLM

Keterangan:

(a_k) = aspek ke-(k) yang diberikan pada ulasan.

(\hat{s}_k) = sentimen hasil pelabelan LLM terhadap aspek ke-(k).

(k) = jumlah aspek yang relevan dalam satu ulasan.

Pada penelitian ini, hasil pelabelan dari LLM tidak langsung digunakan sebagai dataset akhir tanpa pemeriksaan. Data hasil pelabelan diperiksa melalui validasi format, pengecekan label tidak valid, pemeriksaan duplikasi, dan penyaringan data yang tidak memenuhi struktur keluaran. Pemeriksaan ini dilakukan agar dataset yang digunakan pada tahap pemodelan memiliki format yang konsisten dan sesuai dengan kategori aspek serta kelas sentimen yang telah ditentukan.

Selain pemeriksaan format, hasil pelabelan LLM juga perlu dipahami sebagai label berbantuan, bukan sebagai *gold standard label* yang sepenuhnya bebas dari kesalahan. Label yang dihasilkan LLM dapat mengandung ketidaksesuaian, bias, atau kesalahan interpretasi terhadap konteks ulasan. Oleh karena itu, validasi manual terhadap sampel hasil pelabelan diperlukan untuk melihat tingkat kesesuaian label aspek dan sentimen sebelum dataset digunakan untuk melatih model klasifikasi.

Model LLM yang digunakan dalam penelitian ini adalah Qwen2.5-0.5B-Instruct. Model tersebut digunakan untuk membantu menghasilkan label aspek dan sentimen pada ulasan Gojek. Setelah proses pelabelan selesai, hasil label dikonversi ke dalam format *aspect-level* agar dapat digunakan untuk proses *fine-tuning* IndoBERT dan mDeBERTa-v3. Dengan demikian, *LLM-assisted*

labeling berperan sebagai tahap pembentukan dataset, sedangkan model klasifikasi akhir tetap dibangun menggunakan model *transformer*.

2.2.11 Transformer

Transformer merupakan arsitektur *deep learning* yang banyak digunakan dalam pemrosesan bahasa alami. Arsitektur ini menggunakan mekanisme *self-attention* untuk menangkap hubungan antar-*token* dalam teks. Berbeda dengan model rekuren yang memproses kata secara berurutan, *transformer* dapat memperhatikan seluruh bagian teks secara bersamaan. Kemampuan tersebut membuat model mampu memahami konteks kalimat dengan lebih baik [28].

Dalam penelitian ini, *transformer* menjadi dasar pemilihan IndoBERT dan mDeBERTa-v3. Kedua model tersebut digunakan karena mampu membangun representasi teks berdasarkan konteks. Kemampuan ini penting dalam analisis sentimen berbasis aspek karena makna kata dapat berubah bergantung pada kalimat dan aspek layanan yang dibahas. Misalnya, kata "mahal" dapat berkaitan dengan aspek Tarif & Harga, sedangkan kata "error" dapat berkaitan dengan aspek Aplikasi.

Konsep utama dalam *transformer* adalah mekanisme *self-attention*. Mekanisme ini menghitung hubungan antara *query*, *key*, dan *value* untuk menentukan bagian teks yang perlu diberi perhatian lebih besar oleh model. Secara umum, mekanisme *scaled dot-product attention* dapat dinyatakan sebagai berikut.

$$Attention(Q, K, V) = softmax\left(\backslash frac\{QK^T\}\left\{\sqrt{\{d_k\}}\right\}\right)V$$

2. 17 Rumus mekanisme scaled dot-product

Keterangan:

(Q) = matriks *query*.

(K) = matriks *key*.

(V) = matriks *value*.

(d_k) = dimensi dari vektor *key*.

(\sqrt{d_k}) = faktor skala untuk menstabilkan nilai hasil perkalian (QK^T).

Melalui mekanisme tersebut, model dapat memberikan bobot perhatian yang berbeda pada setiap bagian teks. Hal ini memungkinkan model memahami hubungan antarkata dalam ulasan pengguna, termasuk hubungan antara aspek layanan dan opini yang diberikan. Dalam penelitian ini, kemampuan *transformer* digunakan untuk membantu model mengenali pola sentimen pada ulasan aplikasi Gojek.

Dengan penggunaan model berbasis *transformer*, proses klasifikasi tidak hanya bergantung pada frekuensi kata. Model dapat mempelajari konteks kalimat, hubungan antarkata, serta pola sentimen dalam teks ulasan. Oleh karena itu, arsitektur *transformer* relevan digunakan sebagai dasar dalam pemodelan klasifikasi sentimen berbasis aspek menggunakan IndoBERT dan mDeBERTa-v3 [9], [11], [28].

2.2.12 IndoBERT

IndoBERT merupakan model bahasa berbasis BERT yang dikembangkan untuk pemrosesan bahasa Indonesia. Model ini dilatih menggunakan korpus berbahasa Indonesia sehingga memiliki kemampuan untuk memahami struktur bahasa, kosakata, dan konteks kalimat dalam bahasa Indonesia [9]. IndoBERT menggunakan arsitektur *transformer* berbasis BERT yang memanfaatkan mekanisme *self-attention* untuk membangun representasi teks secara kontekstual.

Dalam penelitian ini, IndoBERT digunakan sebagai salah satu model klasifikasi sentimen berbasis aspek. Pemilihan IndoBERT didasarkan pada karakteristik data penelitian yang berupa ulasan pengguna aplikasi Gojek berbahasa Indonesia. Karena data ulasan didominasi oleh bahasa Indonesia, IndoBERT dinilai relevan digunakan untuk memahami pola bahasa, konteks opini, serta hubungan antara aspek layanan dan sentimen pengguna [9], [10].

Pada proses klasifikasi teks menggunakan IndoBERT, teks masukan terlebih dahulu diproses oleh *tokenizer* menjadi deretan token. Token tersebut kemudian dimasukkan ke dalam model IndoBERT untuk menghasilkan representasi kontekstual. Representasi dari token khusus [CLS] umumnya digunakan sebagai representasi keseluruhan teks untuk tugas klasifikasi. Secara sederhana, proses representasi teks menggunakan IndoBERT dapat dinyatakan sebagai berikut.

$$h_i = IndoBERT(x_i)$$

2. 18 Rumus Proses Representasi Teks

Keterangan:

(x_i) = representasi input teks ulasan ke-(i) setelah proses tokenisasi.

(IndoBERT) = model bahasa IndoBERT yang digunakan untuk menghasilkan representasi kontekstual.

(h_i) = representasi kontekstual teks ulasan ke-(i).

Setelah representasi teks diperoleh, hasil tersebut masuk ke lapisan klasifikasi untuk menghasilkan probabilitas kelas sentimen. Proses klasifikasi dapat dinyatakan sebagai berikut.

$$\{y\}_i = softmax(W\ h_i + b)$$

2. 19 Rumus Proses Klarifikasi

Keterangan:

(\hat{y}_i) = probabilitas prediksi kelas sentimen untuk ulasan ke-(i).

(W) = matriks bobot pada lapisan klasifikasi.

(h_i) = representasi kontekstual dari IndoBERT.

(b) = bias pada lapisan klasifikasi.

(softmax) = fungsi aktivasi untuk menghasilkan probabilitas pada setiap kelas.

Pada penelitian ini, model IndoBERT yang digunakan adalah indobenchmark/indobert-base-p1. Model tersebut di-*fine-tune* menggunakan dataset hasil *LLM-assisted labeling* yang telah dikonversi ke format *aspect-level*. Dengan format tersebut, model tidak hanya mempelajari teks ulasan secara umum, tetapi juga mempelajari hubungan antara teks, aspek layanan, dan label sentimen.

IndoBERT digunakan sebagai model utama karena memiliki kesesuaian dengan bahasa data penelitian. Model ini diharapkan mampu menangkap konteks bahasa Indonesia dalam ulasan pengguna Gojek, termasuk penggunaan kata tidak baku, opini pendek, dan ekspresi keluhan atau apresiasi terhadap layanan. Oleh karena itu, IndoBERT relevan digunakan untuk tugas klasifikasi sentimen berbasis aspek pada ulasan aplikasi Gojek berbahasa Indonesia.

2.2.13 mDeBERTa-v3

mDeBERTa-v3 merupakan model *transformer* multilingual yang berasal dari keluarga DeBERTa-v3. DeBERTa atau *Decoding-enhanced BERT with Disentangled Attention* dikembangkan sebagai pengembangan dari BERT dengan mekanisme representasi yang memisahkan informasi posisi dan konten token. Pemisahan ini bertujuan agar model dapat memahami hubungan antarkata dan posisi kata dalam kalimat secara lebih efektif [11].

DeBERTa-v3 menggunakan pendekatan *pre-training* berbasis *replaced token detection*. Pendekatan ini berbeda dari metode *masked language modeling* pada BERT karena model dilatih untuk mendeteksi token yang digantikan dalam teks. Selain itu, DeBERTa-v3 juga menggunakan mekanisme *gradient-disentangled embedding sharing* untuk meningkatkan efisiensi dan kualitas representasi selama proses *pre-training* [11].

Dalam penelitian ini, mDeBERTa-v3 digunakan sebagai model pembanding terhadap IndoBERT. IndoBERT mewakili model monolingual yang dikembangkan khusus untuk bahasa Indonesia, sedangkan mDeBERTa-v3 mewakili model multilingual yang memiliki cakupan lintas bahasa.

Perbandingan ini dilakukan untuk mengetahui apakah model monolingual bahasa Indonesia atau model multilingual memberikan performa yang lebih sesuai pada tugas klasifikasi sentimen berbasis aspek terhadap ulasan aplikasi Gojek.

Secara sederhana, proses pembentukan representasi teks menggunakan mDeBERTa-v3 dapat dinyatakan sebagai berikut.

$$h_i = mDeBERTa(x_i)$$

2. 20 Rumus Proses Pembentukkan Representasi Teks

Keterangan:

(x_i) = representasi input teks ulasan ke-(i) setelah proses tokenisasi.

(mDeBERTa) = model mDeBERTa-v3 yang digunakan untuk menghasilkan representasi kontekstual.

(h_i) = representasi kontekstual teks ulasan ke-(i).

Setelah representasi kontekstual diperoleh, hasil tersebut masuk ke lapisan klasifikasi untuk menghasilkan probabilitas kelas sentimen. Proses klasifikasi dapat dinyatakan sebagai berikut.

$$\{y\}_i = softmax(W\, h_i + b)$$

2. 21 Rumus Proses Klarifikasi

Keterangan:

(\hat{y}_i) = probabilitas prediksi kelas sentimen untuk ulasan ke-(i).

(W) = matriks bobot pada lapisan klasifikasi.

(h_i) = representasi kontekstual dari mDeBERTa-v3.

(b) = bias pada lapisan klasifikasi.

(softmax) = fungsi aktivasi untuk menghasilkan probabilitas pada setiap kelas.

Pada penelitian ini, model yang digunakan adalah microsoft/mdeberta-v3-base. Model tersebut di-*fine-tune* menggunakan dataset yang sama dengan IndoBERT, yaitu dataset hasil *LLM-assisted labeling* yang telah dikonversi ke format *aspect-level*. Penggunaan dataset yang sama dilakukan agar hasil evaluasi kedua model dapat dibandingkan secara lebih adil.

Pemilihan mDeBERTa-v3 tidak dimaksudkan untuk menggantikan IndoBERT sebagai model khusus bahasa Indonesia, melainkan untuk menyediakan pembanding berbasis multilingual. Dengan adanya perbandingan ini, penelitian dapat melihat apakah model multilingual mampu bersaing dengan model monolingual dalam konteks ulasan aplikasi Gojek berbahasa Indonesia. Oleh karena itu, mDeBERTa-v3 relevan digunakan sebagai model pembanding dalam penelitian klasifikasi sentimen berbasis aspek.

2.2.14 Semantic Network Analysis

Semantic Network Analysis atau SNA merupakan pendekatan analisis yang digunakan untuk memetakan hubungan antarkata atau konsep dalam teks. Dalam SNA, kata atau konsep direpresentasikan sebagai *node*, sedangkan hubungan antarkata direpresentasikan sebagai *edge*. Hubungan tersebut dapat terbentuk ketika dua kata muncul bersama dalam satu konteks tertentu, seperti dalam satu ulasan atau satu dokumen [12].

Dalam penelitian ini, SNA digunakan sebagai analisis pendukung terhadap hasil *Aspect-Based Sentiment Analysis*. ABSA digunakan untuk menentukan aspek dan sentimen pada setiap ulasan, sedangkan SNA digunakan untuk memetakan kata atau isu yang membentuk pola opini pada ulasan bersentimen positif dan negatif. Sentimen positif digunakan untuk melihat pola apresiasi atau kepuasan pengguna, sedangkan sentimen negatif digunakan untuk melihat pola keluhan atau permasalahan layanan. Sentimen netral tidak digunakan dalam pembentukan jaringan karena tidak menunjukkan arah opini yang kuat. Dengan demikian, hubungan antara ABSA dan SNA terletak pada penggunaan hasil

label aspek-sentimen dari ABSA sebagai dasar untuk membangun jaringan *keyword co-occurrence*.

SNA dalam penelitian ini dilakukan menggunakan pendekatan *keyword co-occurrence*. Pendekatan ini menghitung kemunculan dua kata secara bersama dalam satu unit teks. Apabila dua kata sering muncul bersama, maka hubungan antara kedua kata tersebut dianggap lebih kuat. Secara sederhana, bobot hubungan antara dua kata dapat dinyatakan sebagai berikut.

$$w_{\{ij\}} = \sum_{\{d \in D\} I\left(t_i \in d \wedge t_j \in d\right)}$$

2. 22 Rumus Bobot Hubungan Antara dua kata

Keterangan:

(w_{ij}) = bobot hubungan antara kata (t_i) dan kata (t_j).

(D) = kumpulan dokumen atau ulasan.

(d) = satu dokumen atau satu ulasan dalam dataset.

(t_i) dan (t_j) = dua kata yang diamati dalam jaringan.

(I(\cdot)) = fungsi indikator yang bernilai 1 jika kondisi terpenuhi dan 0 jika tidak terpenuhi.

Selain melihat hubungan antarkata, SNA juga dapat digunakan untuk menghitung sentralitas kata dalam jaringan. Salah satu ukuran yang digunakan adalah *degree centrality*, yaitu ukuran yang menunjukkan banyaknya hubungan suatu *node* dengan *node* lain. *Degree centrality* dapat dinyatakan sebagai berikut.

$$C_{D(v)} = \backslash frac\{\deg(v)\}\{n - 1\}$$

2. 23 Rumus Degree Centrality

Keterangan:

(C_D(v)) = nilai *degree centrality* pada *node* (v).

(deg(v)) = jumlah hubungan yang dimiliki *node* (v).

(n) = jumlah seluruh *node* dalam jaringan.

Ukuran lain yang dapat digunakan adalah *betweenness centrality*. Ukuran ini menunjukkan seberapa sering suatu *node* berada pada jalur terpendek antara *node* lain. *Betweenness centrality* dapat dinyatakan sebagai berikut.

$$C_{B(v)} = \sum_{\{s \neq v \neq t\}} \backslash frac\{\sigma_{\{st\}(v)}\}\{\sigma_{\{st\}}\}$$

2. 24 Rumus Betweenness Centrality

Keterangan:

(C_B(v)) = nilai *betweenness centrality* pada *node* (v).

(\sigma_{st}) = jumlah jalur terpendek antara *node* (s) dan *node* (t).

(\sigma_{st}(v)) = jumlah jalur terpendek antara *node* (s) dan *node* (t) yang melalui *node* (v).

Selain itu, *closeness centrality* digunakan untuk melihat kedekatan suatu *node* dengan *node* lain dalam jaringan. Ukuran ini dapat dinyatakan sebagai berikut.

$$C_{C(v)} = \backslash frac\{n-1\}\left\{\sum_{\{u \neq v\}d(v,u)}\right\}$$

2. 25 Rumus Closeness Centrality

Keterangan:

(C_C(v)) = nilai *closeness centrality* pada *node* (v).

(n) = jumlah seluruh *node* dalam jaringan.

(d(v,u)) = jarak terpendek antara *node* (v) dan *node* (u).

Dalam penelitian ini, SNA digunakan sebagai analisis pendukung terhadap hasil *Aspect-Based Sentiment Analysis*. ABSA digunakan untuk menentukan sentimen pada setiap aspek layanan, sedangkan SNA digunakan untuk memetakan kata atau

isu yang membentuk pola keluhan pada ulasan negatif. Dengan demikian, hubungan antara ABSA dan SNA terletak pada penggunaan hasil sentimen negatif dari ABSA sebagai dasar untuk membangun jaringan kata. Pendekatan ini membantu menjelaskan tidak hanya aspek yang memiliki sentimen negatif, tetapi juga kata atau isu apa saja yang saling berkaitan dalam keluhan pengguna.

2.2.15 Metrik Evaluasi Model

Metrik evaluasi model digunakan untuk mengukur performa model klasifikasi dalam memprediksi kelas sentimen. Pada penelitian ini, evaluasi dilakukan terhadap model IndoBERT dan mDeBERTa-v3. Metrik yang digunakan meliputi *accuracy*, *precision*, *recall*, *F1-score*, *macro average*, *weighted average*, *classification report*, dan *confusion matrix* [30].

Accuracy digunakan untuk mengukur proporsi prediksi yang benar dari seluruh data uji. Metrik ini menunjukkan seberapa banyak data yang berhasil diklasifikasikan dengan benar oleh model. Rumus *accuracy* dapat dinyatakan sebagai berikut.

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN}$$

2. 26 Rumus Accuracy

Keterangan:

(TP) = *True Positive*, yaitu data positif yang diprediksi benar sebagai positif.

(TN) = *True Negative*, yaitu data negatif yang diprediksi benar sebagai negatif.

(FP) = *False Positive*, yaitu data negatif yang salah diprediksi sebagai positif.

(FN) = *False Negative*, yaitu data positif yang salah diprediksi sebagai negatif.

Precision digunakan untuk mengukur ketepatan model dalam memprediksi suatu kelas. Metrik ini menunjukkan berapa banyak prediksi pada suatu kelas yang benar dibandingkan seluruh data yang diprediksi sebagai kelas tersebut. Rumus *precision* dapat dinyatakan sebagai berikut.

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

2. 27 Rumus Precision

Recall digunakan untuk mengukur kemampuan model dalam menemukan seluruh data yang benar-benar termasuk ke dalam suatu kelas. Metrik ini menunjukkan berapa banyak data aktual pada suatu kelas yang berhasil dikenali oleh model. Rumus *recall* dapat dinyatakan sebagai berikut.

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

2. 28 Rumus Recall

F1-score merupakan rata-rata harmonik antara *precision* dan *recall*. Metrik ini digunakan untuk melihat keseimbangan antara ketepatan prediksi dan kemampuan model dalam menemukan data yang relevan. Rumus *F1-score* dapat dinyatakan sebagai berikut.

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

2. 29 Rumus F1-Score

Pada klasifikasi multikelas, nilai *precision*, *recall*, dan *F1-score* dapat dihitung untuk setiap kelas sentimen. Penelitian ini menggunakan tiga kelas sentimen, yaitu positif, negatif, dan netral. Oleh karena itu, digunakan *macro average* dan *weighted average* untuk merangkum performa model pada seluruh kelas.

Macro average menghitung rata-rata metrik dari seluruh kelas dengan bobot yang sama. Rumus *macro average* dapat dinyatakan sebagai berikut.

$$Macro\ Average = \frac{1}{K} \sum_{k=1}^{K} M$$

2. 30 Rumus Macro Average

Keterangan:

(K) = jumlah kelas sentimen.

(M_k) = nilai metrik pada kelas ke-(k), misalnya *precision*, *recall*, atau *F1-score*.

Weighted average menghitung rata-rata metrik dari seluruh kelas dengan mempertimbangkan jumlah data pada setiap kelas. Rumus *weighted average* dapat dinyatakan sebagai berikut.

$$Weighted\ Average\ = \sum_{\{k=1\}}^{\{K\}} \ \backslash frac\{n_k\}\{N\}M_k$$

2. 31 Rumus Weighted Average

Keterangan:

(n_k) = jumlah data pada kelas ke-(k).

(N) = jumlah seluruh data.

(M_k) = nilai metrik pada kelas ke-(k).

Dalam penelitian ini, *F1-score macro* digunakan sebagai metrik utama dalam membandingkan performa IndoBERT dan mDeBERTa-v3. Metrik ini dipilih karena dapat menunjukkan performa rata-rata setiap kelas dengan bobot yang sama. Hal ini penting karena jumlah data pada setiap kelas sentimen tidak selalu seimbang.

Selain metrik numerik, penelitian ini juga menggunakan *classification report* dan *confusion matrix*. *Classification report* digunakan untuk menampilkan nilai *precision*, *recall*, dan *F1-score* pada setiap kelas. Sementara itu, *confusion matrix* digunakan untuk melihat pola kesalahan prediksi antar kelas. Dengan menggunakan beberapa metrik tersebut, evaluasi model tidak hanya bergantung pada *accuracy*, tetapi juga mempertimbangkan performa model pada masing-masing kelas sentimen.

2.2.16 Kompleksitas Komputasi Model

Kompleksitas komputasi model merupakan aspek yang digunakan untuk melihat kebutuhan sumber daya dalam proses pelatihan dan evaluasi model. Pada model berbasis *transformer*, kompleksitas komputasi menjadi penting karena model jenis ini umumnya memiliki jumlah parameter yang besar dan

membutuhkan sumber daya komputasi yang lebih tinggi dibandingkan metode klasifikasi sederhana. Kompleksitas tersebut dapat terlihat dari jumlah parameter model, ukuran data, panjang token, jumlah *epoch*, ukuran *batch*, penggunaan memori, kebutuhan GPU, serta waktu pelatihan.

Dalam penelitian ini, kompleksitas komputasi dibahas karena model yang digunakan, yaitu IndoBERT dan mDeBERTa-v3, merupakan model berbasis *transformer*. Kedua model tersebut memiliki karakteristik yang berbeda. IndoBERT merupakan model monolingual yang dikembangkan untuk bahasa Indonesia, sedangkan mDeBERTa-v3 merupakan model multilingual yang memiliki cakupan lintas bahasa. Perbedaan karakteristik tersebut dapat memengaruhi kebutuhan komputasi pada proses *fine-tuning*.

Salah satu indikator kompleksitas komputasi adalah jumlah parameter model. Jumlah parameter menunjukkan banyaknya bobot yang dipelajari atau digunakan oleh model dalam membangun representasi teks. Secara umum, jumlah parameter model dapat dinyatakan sebagai berikut.

$$P = \sum_{\{l=1\}_l^{\{L\}p}}$$

2. 32 Rumus Jumlah Parameter

Keterangan:

(P) = jumlah seluruh parameter model.

(L) = jumlah lapisan atau komponen model.

(p_l) = jumlah parameter pada lapisan atau komponen ke-(l).

Selain jumlah parameter, waktu pelatihan juga menjadi indikator penting dalam melihat kompleksitas komputasi. Waktu pelatihan dipengaruhi oleh jumlah data, jumlah *epoch*, ukuran *batch*, panjang token, arsitektur model, serta perangkat komputasi yang digunakan. Secara sederhana, waktu pelatihan total dapat dinyatakan sebagai berikut.

$$T_{\{total\}} = \sum_{\{e=1\}_e^{\{E\}T}}$$

2. 33 Rumus Waktu Pelatihan Total

Keterangan:

(T_{total}) = total waktu pelatihan model.

(E) = jumlah *epoch*.

(T_e) = waktu pelatihan pada *epoch* ke-(e).

Kompleksitas pelatihan juga dapat dilihat dari jumlah langkah pembaruan parameter atau *training steps*. Jumlah langkah pelatihan dipengaruhi oleh jumlah data latih, ukuran *batch*, dan jumlah *epoch*. Secara umum, jumlah *training steps* dapat dinyatakan sebagai berikut.

$$S = \lceil \backslash frac\{N\}\{B\}\rceil \times E$$

2. 34 Rumus Jumlah Training Steps

Keterangan:

(S) = jumlah *training steps*.

(N) = jumlah data latih.

(B) = ukuran *batch*.

(E) = jumlah *epoch*.

Pada proses *fine-tuning*, ukuran *batch* dan penggunaan GPU berpengaruh terhadap efisiensi pelatihan. Ukuran *batch* yang lebih besar dapat mempercepat proses pelatihan, tetapi membutuhkan memori GPU yang lebih besar. Sebaliknya, ukuran *batch* yang lebih kecil dapat mengurangi penggunaan memori, tetapi dapat meningkatkan waktu pelatihan. Oleh karena itu, konfigurasi seperti *batch size*, *gradient accumulation*, panjang token maksimum, dan jumlah *epoch* perlu disesuaikan dengan kemampuan perangkat yang digunakan.

Dalam penelitian ini, pembahasan kompleksitas komputasi digunakan untuk memberikan gambaran bahwa evaluasi model tidak hanya dilihat dari performa klasifikasi, tetapi juga dari kebutuhan sumber daya yang digunakan. Dengan demikian, perbandingan IndoBERT dan mDeBERTa-v3 tidak hanya mencakup

nilai *accuracy*, *precision*, *recall*, dan *F1-score*, tetapi juga mempertimbangkan efisiensi komputasi seperti jumlah parameter, konfigurasi pelatihan, penggunaan GPU, dan waktu pelatihan.

2.3 Framework/Algoritma yang digunakan

Penelitian ini menggunakan CRISP-DM sebagai *framework* utama. *Framework* ini digunakan karena penelitian dilakukan melalui tahapan *data mining* yang sistematis dan berurutan. Tahapan tersebut meliputi *business understanding*, *data understanding*, *data preparation*, *modeling*, *evaluation*, dan *deployment*. Dalam penelitian ini, CRISP-DM disesuaikan dengan kebutuhan analisis sentimen berbasis aspek pada ulasan pengguna aplikasi Gojek.

Selain CRISP-DM, penelitian ini menggunakan beberapa pendekatan utama, yaitu *LLM-assisted labeling*, *fine-tuning* IndoBERT, *fine-tuning* mDeBERTa-v3, serta *Semantic Network Analysis* berbasis *keyword co-occurrence*. *LLM-assisted labeling* digunakan untuk membantu pembentukan dataset berlabel aspek dan sentimen. IndoBERT dan mDeBERTa-v3 digunakan sebagai model klasifikasi sentimen berbasis aspek. Sementara itu, *Semantic Network Analysis* digunakan untuk memetakan keterkaitan kata dan isu dominan pada ulasan negatif pengguna aplikasi Gojek.

2.3.1 CRISP-DM

CRISP-DM atau *Cross Industry Standard Process for Data Mining* merupakan *framework* yang digunakan untuk mengatur proses *data mining* secara sistematis. *Framework* ini terdiri atas enam tahap utama, yaitu *business understanding*, *data understanding*, *data preparation*, *modeling*, *evaluation*, dan *deployment* [31], [32].

Dalam penelitian ini, CRISP-DM digunakan sebagai kerangka kerja utama karena tahapan penelitian melibatkan proses pengumpulan data, pengolahan data teks, pelabelan berbantuan LLM, pemodelan klasifikasi, evaluasi model, interpretasi hasil, dan penyajian hasil penelitian. Setiap tahap CRISP-DM

diterapkan sesuai dengan kebutuhan penelitian analisis sentimen berbasis aspek pada ulasan aplikasi Gojek.

1. Business Understanding

Tahap *business understanding* digunakan untuk memahami masalah dan tujuan penelitian. Permasalahan utama dalam penelitian ini adalah kebutuhan untuk memahami persepsi pengguna terhadap layanan Gojek berdasarkan ulasan aplikasi. Ulasan pengguna tidak hanya memuat sentimen umum, tetapi juga berisi opini terhadap beberapa aspek layanan.

Gojek memiliki beberapa layanan dalam satu aplikasi. Layanan tersebut mencakup Aplikasi, Layanan, Driver, Tarif & Harga, GoPay & Pembayaran, serta Promo & Iklan. Oleh karena itu, analisis sentimen umum belum cukup untuk mengetahui sumber kepuasan atau keluhan pengguna. Penelitian ini menggunakan *Aspect-Based Sentiment Analysis* agar opini pengguna dapat dikaitkan dengan aspek layanan tertentu [6], [15].

Output dari tahap ini adalah rumusan masalah, tujuan penelitian, batasan masalah, dan kebutuhan analisis.

2. Data Understanding

Tahap *data understanding* digunakan untuk memahami sumber dan karakteristik data. Data yang digunakan dalam penelitian ini berupa ulasan pengguna aplikasi Gojek dari Google Play Store. Data ulasan tersebut memuat teks ulasan, *rating*, tanggal ulasan, versi aplikasi, dan atribut pendukung lainnya.

Pada tahap ini dilakukan pemeriksaan awal terhadap dataset. Pemeriksaan meliputi jumlah data, struktur kolom, data kosong, data duplikat, distribusi *rating*, dan karakteristik teks ulasan. Tahap ini dilakukan agar kondisi awal data diketahui sebelum masuk ke tahap *cleaning*, *preprocessing*, dan pelabelan [2], [4].

Output dari tahap ini adalah gambaran awal dataset, struktur data, jumlah data mentah, distribusi *rating*, dan informasi kualitas awal data.

3. Data Preparation

Tahap *data preparation* digunakan untuk menyiapkan data agar layak digunakan dalam proses pelabelan dan pemodelan. Tahap ini mencakup *cleaning*, penghapusan duplikasi, penghapusan teks kosong, normalisasi teks, dan *preprocessing*. Data yang telah diproses kemudian digunakan pada tahap pelabelan sentimen berbasis aspek.

Pada tahap ini juga dilakukan *LLM-assisted labeling*. LLM digunakan sebagai alat bantu untuk memberikan label aspek dan sentimen pada ulasan pengguna Gojek. Aspek yang digunakan dalam penelitian ini dibatasi pada enam kategori, yaitu Aplikasi, Layanan, Driver, Tarif & Harga, GoPay & Pembayaran, serta Promo & Iklan. Kelas sentimen yang digunakan terdiri atas positif, negatif, dan netral [7], [26].

Hasil pelabelan LLM tidak langsung digunakan sebagai dataset final tanpa pemeriksaan. Dataset hasil pelabelan divalidasi untuk memastikan tidak terdapat label tidak valid, data duplikat, teks kosong, atau format label yang tidak sesuai. Validasi juga dilakukan untuk mengurangi risiko kesalahan label karena dataset latih dan data uji dalam penelitian ini berasal dari proses pelabelan berbantuan LLM.

Setelah proses pemeriksaan, dataset dikonversi ke format *aspect-level*. Format ini digunakan karena satu ulasan dapat menghasilkan lebih dari satu pasangan aspek dan sentimen. *Output* dari tahap ini adalah dataset berlabel aspek dan sentimen yang siap digunakan untuk proses *fine-tuning* IndoBERT dan mDeBERTa-v3.

4. Modeling

Tahap *modeling* digunakan untuk membangun model klasifikasi sentimen berbasis aspek. Model yang digunakan adalah IndoBERT dan mDeBERTa-v3.

Kedua model dipilih karena sama-sama berbasis *transformer*, tetapi memiliki karakteristik berbeda.

IndoBERT mewakili model monolingual yang dikembangkan untuk bahasa Indonesia. Model ini digunakan karena data penelitian berupa ulasan pengguna berbahasa Indonesia. Sementara itu, mDeBERTa-v3 digunakan sebagai model multilingual. Perbandingan kedua model dilakukan untuk mengetahui model yang lebih sesuai dalam klasifikasi sentimen berbasis aspek pada ulasan aplikasi Gojek [9], [11].

Pada tahap ini, dataset dibagi menjadi data latih dan data uji. Input model disusun berdasarkan teks ulasan dan aspek layanan, sedangkan target klasifikasi adalah label sentimen. Kedua model di-*fine-tune* menggunakan dataset yang sama agar hasil evaluasi dapat dibandingkan secara adil.

Output dari tahap ini adalah model IndoBERT dan mDeBERTa-v3 yang telah dilatih pada dataset sentimen berbasis aspek.

5. Evaluation

Tahap *evaluation* digunakan untuk mengukur performa model. Evaluasi dilakukan terhadap IndoBERT dan mDeBERTa-v3 menggunakan *accuracy*, *precision*, *recall*, *F1-score weighted*, *F1-score macro*, *classification report*, dan *confusion matrix* [30].

Evaluasi dilakukan dalam dua bentuk. Pertama, evaluasi umum dilakukan untuk membandingkan performa IndoBERT dan mDeBERTa-v3 pada data uji. Kedua, evaluasi berdasarkan aspek dominan dilakukan untuk mengetahui performa model terbaik pada kelompok aspek layanan tertentu.

Pada tahap ini juga dilakukan analisis kesalahan prediksi untuk melihat pola kesalahan model pada kelas negatif, netral, dan positif. Selain itu, hasil klasifikasi sentimen negatif digunakan sebagai dasar untuk analisis lanjutan menggunakan *Semantic Network Analysis*.

Output dari tahap ini adalah hasil perbandingan model, model terbaik, nilai metrik evaluasi, *classification report*, *confusion matrix*, dan analisis kesalahan prediksi.

6. Deployment

Tahap *deployment* dalam penelitian ini tidak dimaknai sebagai integrasi langsung ke sistem Gojek. Tahap ini dilakukan dalam bentuk penyajian hasil penelitian, penyimpanan model, penyusunan *pipeline* analisis, dan pembuatan demonstrasi sederhana menggunakan Streamlit.

Deployment digunakan untuk menunjukkan bagaimana model dan hasil analisis dapat disajikan dalam bentuk aplikasi demonstrasi. Aplikasi tersebut dapat menampilkan proses prediksi sentimen berbasis aspek dan ringkasan hasil analisis. Dengan demikian, tahap *deployment* berperan sebagai media penyajian hasil penelitian, bukan sebagai implementasi operasional pada sistem produksi.

Output dari tahap ini adalah model tersimpan, *pipeline* analisis, visualisasi hasil evaluasi, dan aplikasi demonstrasi.

2.3.2 LLM-Assisted Labeling

LLM-assisted labeling merupakan pendekatan pelabelan data yang memanfaatkan *Large Language Model* sebagai alat bantu anotasi. Pendekatan ini digunakan karena dataset ulasan pengguna berjumlah besar dan memerlukan label aspek serta sentimen pada setiap ulasan. Pelabelan manual terhadap seluruh data membutuhkan waktu panjang, sehingga LLM digunakan untuk membantu pembentukan dataset berlabel [7], [26].

Dalam penelitian ini, LLM diarahkan menggunakan *prompt* terstruktur. *Prompt* tersebut berisi instruksi untuk membaca ulasan pengguna, menentukan aspek layanan yang dibahas, memberikan label sentimen, dan menghasilkan keluaran dalam format yang telah ditentukan. Aspek yang digunakan adalah Aplikasi, Layanan, Driver, Tarif & Harga, GoPay & Pembayaran, serta Promo & Iklan. Kelas sentimen yang digunakan terdiri atas positif, negatif, dan netral.

Model LLM yang digunakan adalah Qwen2.5-0.5B-Instruct. Model tersebut digunakan untuk menghasilkan label aspek dan sentimen pada data ulasan Gojek. Output pelabelan dibuat dalam format JSON agar hasil pelabelan lebih mudah diperiksa, divalidasi, dan diproses kembali [8].

Hasil pelabelan LLM diperiksa sebelum digunakan pada tahap pemodelan. Pemeriksaan dilakukan untuk memastikan aspek dan sentimen berada pada kategori yang sesuai. Data dengan label tidak valid, duplikasi, teks kosong, atau format tidak sesuai diperbaiki atau dikeluarkan dari dataset. Selain itu, hasil pelabelan LLM dipahami sebagai label berbantuan, bukan sebagai label yang sepenuhnya bebas dari kesalahan. Oleh karena itu, penelitian ini melakukan validasi manual terhadap sampel hasil pelabelan dan audit distribusi label untuk mengurangi risiko bias dari proses pelabelan LLM.

Dengan demikian, *LLM-assisted labeling* berperan sebagai bagian dari tahap *data preparation* dalam CRISP-DM. Hasil dari tahap ini adalah dataset berlabel aspek dan sentimen yang kemudian dikonversi ke format *aspect-level* untuk proses *fine-tuning* model klasifikasi.

2.3.3 Fine-Tuning IndoBERT

IndoBERT digunakan sebagai salah satu model klasifikasi dalam penelitian ini. Model ini dipilih karena dikembangkan untuk bahasa Indonesia. Karakter tersebut sesuai dengan data penelitian yang berupa ulasan pengguna aplikasi Gojek berbahasa Indonesia [9], [10].

IndoBERT digunakan dalam skema *sequence classification*. Pada skema ini, teks ulasan dan aspek layanan menjadi input model, sedangkan label sentimen menjadi output yang diprediksi oleh model. Kelas sentimen yang digunakan terdiri atas negatif, netral, dan positif.

Pada penelitian ini, model yang digunakan adalah indobenchmark/indobert-base-p1. Model tersebut di-*fine-tune* menggunakan dataset hasil *LLM-assisted labeling* yang telah dikonversi ke format *aspect-level*. Proses *fine-tuning*

dilakukan agar model dapat menyesuaikan representasi bahasa dengan pola sentimen pada ulasan aplikasi Gojek [15].

Sebelum proses pelatihan, teks ulasan ditokenisasi menggunakan *tokenizer* IndoBERT. Dataset kemudian dibagi menjadi data latih dan data uji. Model dilatih menggunakan data latih dan dievaluasi menggunakan data uji. Hasil evaluasi IndoBERT kemudian dibandingkan dengan mDeBERTa-v3 menggunakan metrik evaluasi yang sama [28], [30].

2.3.4 Fine-Tuning mDeBERTa-v3

mDeBERTa-v3 digunakan sebagai model pembanding terhadap IndoBERT. Model ini merupakan model *transformer* multilingual dari keluarga DeBERTa-v3. Model tersebut dipilih karena memiliki cakupan lintas bahasa dan dapat digunakan pada teks berbahasa Indonesia [11].

Dalam penelitian ini, mDeBERTa-v3 digunakan dalam skema *sequence classification*. Input yang diberikan berupa teks ulasan dan aspek layanan. Target yang diprediksi adalah label sentimen, yaitu negatif, netral, dan positif.

Model yang digunakan adalah microsoft/mdeberta-v3-base. Model tersebut di-*fine-tune* menggunakan dataset yang sama dengan IndoBERT. Penggunaan dataset yang sama dilakukan agar hasil evaluasi kedua model dapat dibandingkan secara adil.

IndoBERT mewakili model khusus bahasa Indonesia, sedangkan mDeBERTa-v3 mewakili model multilingual. Perbandingan ini digunakan untuk mengetahui apakah model monolingual bahasa Indonesia atau model multilingual lebih sesuai dalam tugas klasifikasi sentimen berbasis aspek pada ulasan aplikasi Gojek [11], [16].

Evaluasi mDeBERTa-v3 dilakukan menggunakan metrik yang sama dengan IndoBERT. Metrik tersebut meliputi *accuracy*, *precision*, *recall*, *F1-score macro*, *F1-score weighted*, *classification report*, dan *confusion matrix* [30].

2.3.5 Semantic Network Analysis Berbasis Keyword Co-occurrence

Semantic Network Analysis digunakan sebagai analisis pendukung dalam penelitian ini. Pendekatan yang digunakan adalah *keyword co-occurrence*, yaitu pemetaan hubungan kata berdasarkan kemunculan bersama dalam teks ulasan. Dalam jaringan tersebut, kata direpresentasikan sebagai *node*, sedangkan hubungan antarkata direpresentasikan sebagai *edge* [12], [20].

SNA digunakan setelah tahap klasifikasi sentimen dilakukan. Data yang digunakan pada tahap ini difokuskan pada ulasan dengan sentimen negatif. Ulasan negatif dipilih karena lebih relevan untuk mengidentifikasi keluhan, hambatan, dan permasalahan layanan yang dialami pengguna aplikasi Gojek.

Pada penelitian ini, SNA digunakan untuk melihat isu dominan dan keterkaitan kata yang muncul dalam ulasan negatif pengguna. Kata-kata dominan dianalisis berdasarkan frekuensi, hubungan antarkata, dan nilai sentralitas dalam jaringan. Dengan pendekatan ini, hasil penelitian tidak hanya menampilkan performa model, tetapi juga menunjukkan kata atau isu yang sering muncul bersama dalam pengalaman negatif pengguna [20].

SNA dalam penelitian ini digunakan sebagai analisis pendukung terhadap hasil ABSA. ABSA menunjukkan sentimen pada setiap aspek layanan, sedangkan SNA menunjukkan kata atau isu yang membentuk sentimen negatif tersebut. Dengan demikian, hubungan antara ABSA dan SNA terletak pada penggunaan data sentimen positif dan negatif hasil ABSA sebagai dasar pembentukan jaringan kata.

SNA dalam penelitian ini tidak digunakan untuk menganalisis hubungan antar pengguna atau akun. Analisis ini hanya digunakan untuk memetakan hubungan semantik antarkata dalam teks ulasan. Oleh karena itu, istilah SNA dalam penelitian ini merujuk pada *Semantic Network Analysis*, bukan *Social Network Analysis*.

2.4 Tools/software yang digunakan

Penelitian ini menggunakan beberapa *tools*, *library*, dan perangkat lunak untuk mendukung proses pengumpulan data, pengolahan teks, pelabelan, pemodelan, evaluasi, visualisasi, *Semantic Network Analysis*, dan *deployment* sederhana. Penggunaan *tools* tersebut disesuaikan dengan tahapan penelitian yang dilakukan, mulai dari proses *scraping* ulasan aplikasi Gojek hingga penyajian hasil analisis. *Tools* dan perangkat lunak yang digunakan dijelaskan sebagai berikut.

2.4.1 Python

Python merupakan bahasa pemrograman tingkat tinggi yang banyak digunakan untuk pengolahan data, *machine learning*, *deep learning*, pemrosesan teks, dan visualisasi data [32]. Dalam penelitian ini, Python digunakan sebagai bahasa pemrograman utama untuk menjalankan seluruh *pipeline* penelitian, mulai dari *scraping* data, *cleaning*, *preprocessing*, *LLM-assisted labeling*, pembentukan dataset *aspect-level*, pemodelan, evaluasi, *Semantic Network Analysis*, hingga penyajian hasil.

2.4.2 Jupyter Notebook

Jupyter Notebook merupakan lingkungan kerja interaktif yang digunakan untuk menulis dan menjalankan kode Python dalam bentuk *cell* [33]. Dalam penelitian ini, Jupyter Notebook digunakan untuk menjalankan eksperimen secara bertahap, mendokumentasikan proses penelitian, menampilkan *output* setiap tahap, serta memeriksa hasil pengolahan data, pelabelan, pemodelan, dan evaluasi model.

2.4.3 google-play-scraper

google-play-scraper merupakan *library* Python yang digunakan untuk mengambil data dari Google Play Store, termasuk ulasan pengguna dan metadata aplikasi [34]. Dalam penelitian ini, *google-play-scraper* digunakan untuk mengumpulkan data ulasan pengguna aplikasi Gojek dari Google Play Store. Data hasil *scraping* kemudian digunakan sebagai data awal dalam proses analisis sentimen berbasis aspek.

2.4.4 Pandas

Pandas merupakan *library* Python yang digunakan untuk mengolah data tabular, seperti membaca, membersihkan, memfilter, menggabungkan, menganalisis, dan menyimpan data [35]. Dalam penelitian ini, Pandas digunakan untuk mengelola dataset ulasan, membersihkan data, memeriksa data kosong dan duplikasi, menghitung distribusi label, menyusun dataset *aspect-level*, serta menyimpan hasil analisis dalam format file yang dibutuhkan.

2.4.5 NumPy

NumPy merupakan *library* Python yang menyediakan struktur *array* dan fungsi komputasi numerik [36]. Dalam penelitian ini, NumPy digunakan untuk mendukung proses perhitungan numerik, manipulasi *array*, pengolahan hasil prediksi model, serta mendukung proses komputasi pada tahap evaluasi dan analisis data.

2.4.6 Hugging Face Transformers

Hugging Face Transformers merupakan *library* yang menyediakan berbagai model dan *tokenizer* berbasis *transformer* [37]. Dalam penelitian ini, *library* ini digunakan untuk memuat *tokenizer* dan model IndoBERT serta mDeBERTa-v3. *Hugging Face Transformers* digunakan pada tahap tokenisasi, *fine-tuning*, prediksi, dan evaluasi model klasifikasi sentimen berbasis aspek.

2.4.7 PyTorch

PyTorch merupakan *framework deep learning* yang digunakan untuk membangun, melatih, dan mengevaluasi model *neural network* [38]. Dalam penelitian ini, PyTorch digunakan sebagai *framework* utama dalam proses *fine-tuning* model IndoBERT dan mDeBERTa-v3. PyTorch juga digunakan untuk menjalankan proses pelatihan pada GPU, mengatur tensor, serta mendukung proses komputasi model berbasis *transformer*.

2.4.8 scikit-learn

scikit-learn merupakan *library* Python yang digunakan untuk *machine learning*, pembagian data, dan evaluasi model [29]. Dalam penelitian ini, *scikit-*

learn digunakan untuk membagi dataset menjadi data latih dan data uji, menghitung metrik evaluasi, menghasilkan *classification report*, serta membuat *confusion matrix*. Metrik yang dihitung meliputi *accuracy*, *precision*, *recall*, *F1-score macro*, dan *F1-score weighted*.

2.4.9 NetworkX

NetworkX merupakan *library* Python yang digunakan untuk membangun, memproses, dan menganalisis struktur jaringan atau graf [39]. Dalam penelitian ini, NetworkX digunakan untuk membangun jaringan kata berbasis *keyword co-occurrence* pada tahap *Semantic Network Analysis*. Kata dalam ulasan negatif direpresentasikan sebagai *node*, sedangkan hubungan kemunculan bersama antarkata direpresentasikan sebagai *edge*.

2.4.10 Matplotlib

Matplotlib merupakan *library* Python yang digunakan untuk membuat visualisasi data dalam bentuk grafik statis [40]. Dalam penelitian ini, Matplotlib digunakan untuk membuat grafik distribusi sentimen, distribusi aspek, visualisasi hasil evaluasi model, grafik perbandingan model, serta visualisasi hasil *Semantic Network Analysis*. Visualisasi tersebut digunakan untuk membantu interpretasi hasil penelitian.

2.4.11 Streamlit

Streamlit merupakan *framework* Python yang digunakan untuk membuat aplikasi web interaktif secara sederhana [41]. Dalam penelitian ini, Streamlit digunakan untuk membuat demo sederhana yang menampilkan hasil prediksi sentimen berbasis aspek menggunakan model terbaik. Penggunaan Streamlit tidak dimaknai sebagai integrasi langsung ke sistem Gojek, tetapi sebagai bentuk *deployment* sederhana untuk menunjukkan bagaimana model dan hasil analisis dapat disajikan dalam bentuk aplikasi interaktif.