

BAB II

LANDASAN TEORI

2.1. Tinjauan Pustaka

2.1.1. Pengertian dan Manfaat Sentimen Analisis

2.1.1.1. Pengertian Analisis Sentimen

Sistem analisis sentimen, juga dikenal sebagai opini mining, adalah studi komputasi tentang perasaan, pandangan, dan emosi yang ada dalam entitas dan atribut yang ditulis dalam teks.

Tujuannya adalah untuk mendapatkan tingkat opini positif dan perasaan negatif yang terkait dengan individu, organisasi, yayasan, barang, atau kondisi tertentu. Hasil akhir dari investigasi perasaan dapat disusun ke dalam tiga klasifikasi utama, khususnya opini yang baik, opini pesimis, dan opini nonpartisan, atau dapat juga dipisahkan untuk mengenali sumber perasaan yang baik atau perasaan pesimis dari pertemuan atau pertemuan tertentu (Nanda Fahriza & Riza, 2023)

2.1.1.2. Manfaat Analisis Sentimen

Manfaat analisis sentimen yaitu sebagai sebuah ide atau evaluasi pada berbagai bidang. Analisis sentimen ini memungkinkan pengelompokan polaritas teks dalam kalimat atau dokumen untuk memahami apakah opini pada teks tersebut bersifat positif atau negative (Khoirul Insan et al., 2023).

2.1.2. Analisis Sentimen Berbasis Aspek

2.1.2.1. Definisi dan Tujuan Analisis Sentimen Berbasis Aspek

2.1.2.1.1. Definisi Analisis sentiment Berbasis Aspek

Analisis Sentimen Berbasis Aspek (ABSA) adalah sub-bidang dari Analisis Sentimen (SA), yang merupakan tugas inti dari pemrosesan bahasa alami (NLP). SA, juga dikenal sebagai "penambangan opini", memecahkan masalah mengidentifikasi dan mengklasifikasikan pengaruh atau orientasi sentimen korpora teks yang diberikan ke dalam kategori polaritas (misalnya "positif, netral, negatif"), skor intensitas/kekuatan (misalnya dari 1 hingga 5), atau kategori lainnya. Bagian "mengidentifikasi subjek opini" dari pencarian terkait dengan granularitas SA. SA tradisional sebagian besar berfokus pada sentimen tingkat kalimat dokumenter dan dengan demikian mengasumsikan satu subjek opini. Dalam beberapa dekade terakhir, ledakan teks opini online telah menarik minat yang meningkat dalam menyaring wawasan yang lebih terarah pada entitas tertentu atau aspek-aspeknya dalam setiap kalimat melalui SA yang lebih halus. Ini adalah masalah yang ingin dipecahkan oleh ABSA(Hua et al., 2024).

Aspect Based Sentiment Analysis (ABSA) adalah jenis analisis sentimen yang lebih rinci yang mengidentifikasi aspek dan opini terkaitnya dari teks yang diberikan. ABSA melibatkan identifikasi sentimen terhadap entitas atau atribut spesifik yang disebut aspek. Aspek ini dapat disebutkan secara eksplisit dalam teks atau tersirat dari konteks. ABSA biasanya melibatkan beberapa subtugas, termasuk ekstraksi aspek,

ekstraksi opini, dan klasifikasi sentimen aspek. Teknik-teknik solusi untuk subtask ABSA ini telah berkembang dari solusi linguistik dan statistik murni ke pendekatan pembelajaran mesin yang dominan, seringkali dengan model atau modul subtask ganda yang diatur dalam sebuah pipeline(Lan et al., 2020).

2.1.2.1.2. Tujuan Analisis Sentimen Berbasis Aspek

Secara umum, jalur penelitian utama ABSA melibatkan identifikasi berbagai elemen sentimen tingkat aspek, yaitu, istilah aspek, kategori aspek, istilah opini, dan polaritas sentiment(Hua et al., 2024).

Dalam penelitian terkait *Aspect-Based Sentiment Analysis* (ABSA), tujuan utama adalah untuk memahami dan mengelompokkan sentimen pada tingkat aspek atau entitas tertentu dalam teks, terutama dalam skenario yang memerlukan pemahaman yang lebih mendalam terhadap opini seseorang. ABSA membantu mengidentifikasi elemen-elemen sentimen penting, seperti istilah suatu aspek, kategori dari aspek, istilah opini, dan polaritas sentimen

2.1.2.2. Langkah Langkah Analisis Sentimen Berbasis Aspek

Aspect term extraction (ATE) adalah tugas mendasar dari ABSA, yang bertujuan untuk mengekstrak ekspresi aspek eksplisit di mana pengguna mengekspresikan opini dalam teks yang diberikan” dan” *aspect sentiment classification task* (ASC) memprediksi polaritas sentimen untuk aspek tertentu dalam sebuah kalimat(Hua et al., 2024).

Langkah-langkah umum dalam pelaksanaan ABSA melibatkan beberapa tahapan analisis. Tahap pertama biasanya adalah ekstraksi istilah aspek, yang berfungsi mengidentifikasi objek yang dibahas dalam teks. proses dilanjutkan dengan klasifikasi sentimen aspek untuk menentukan polaritas sentimen terhadap objek tersebut, apakah positif, negatif, atau netral.

2.1.3. Model Transformer dan Arsitektur Dasar

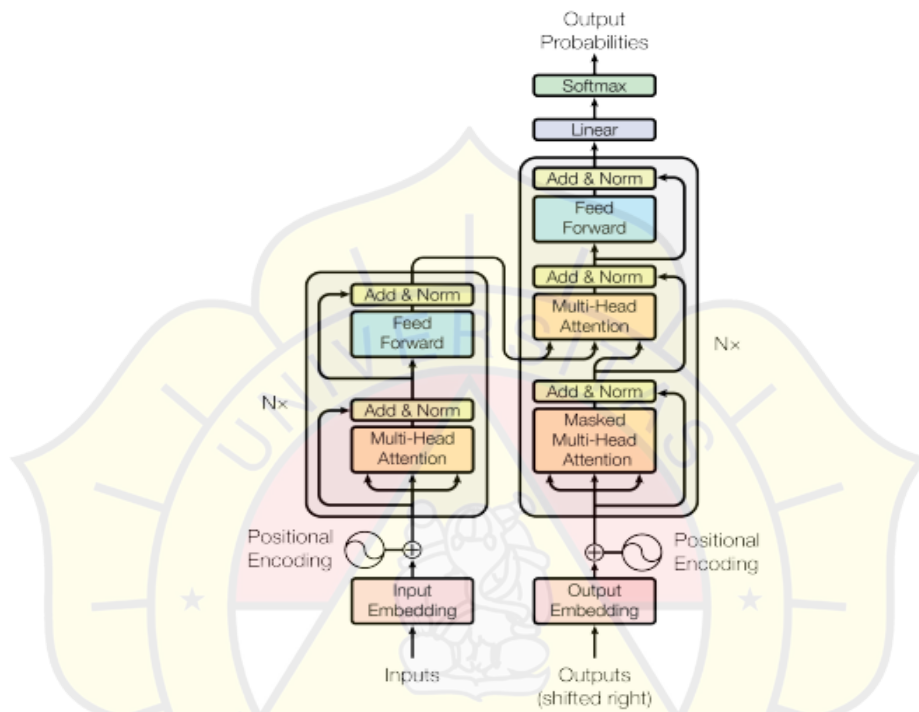
2.1.3.1. Arsitektur dasar Transformer

Model transduksi urutan dominan didasarkan pada jaringan saraf tiruan rekuren atau konvolusional yang kompleks yang mencakup pengkode dan penyandi. Model berkinerja terbaik juga menghubungkan *encoder* dan *decoder* melalui mekanisme perhatian. Makalah ini mengusulkan arsitektur jaringan sederhana baru yang disebut Transformer, yang sepenuhnya didasarkan pada mekanisme perhatian, tanpa rekurensi dan konvolusi sama sekali (Vaswani et al., 2023).

Transformer terdiri dari tumpukan lapisan *encode* dan *decode* identik. Setiap lapisan memiliki dua lapisan sub. Lapisan pertama adalah mekanisme perhatian multi-kepala, dan lapisan kedua adalah jaringan umpan maju sepenuhnya terhubung yang sederhana dan berbasis posisi. Koneksi residual digunakan di sekitar setiap lapisan sub, diikuti dengan normalisasi lapisan. (Vaswani et al., 2023)

Transformer menggunakan perhatian multi-kepala dengan tiga cara berbeda: lapisan perhatian pengkode-penyandi, lapisan *self-attention encode*, dan lapisan *self-*

attention decode. Transformer juga menggunakan penyisipan posisi untuk memasukkan informasi tentang posisi relatif atau absolut token dalam urutan (Vaswani et al., 2023).



Gambar 2. 1 Arsitektur Transformer

Dalam arsitektur ini, transformer menggunakan lapisan *self-attention* yang ditumpuk dan lapisan *point-wise* yang terhubung sepenuhnya untuk encoder dan decoder. Bagian kiri dan kanan menunjukkan bagaimana transformer mengikuti arsitektur ini.

2.1.3.2. Peran *Self-Attention* dalam Model *Transformer*

Self-attention, kadang-kadang disebut *intra-attention*, adalah mekanisme perhatian yang menghubungkan posisi yang berbeda dari satu urutan untuk menghitung representasi dari urutan tersebut (Vaswani et al., 2023).

Self-attention adalah mekanisme yang memungkinkan setiap posisi dalam sebuah urutan untuk "memperhatikan" atau mempertimbangkan informasi dari posisi lain dalam urutan yang sama. Dalam konteks model *Transformer*, *self-attention* memungkinkan model untuk menghubungkan kata-kata atau elemen teks yang relevan dalam sebuah urutan tanpa memperhitungkan jarak antara mereka. Ini sangat bermanfaat dalam memodelkan hubungan antara elemen teks yang berjauhan dalam suatu urutan.

2.1.4. Metode IndoBERT (Indonesian BERT) dan Library

2.1.4.1. Model IndoBERT

Dalam beberapa tahun terakhir, *Transformers* telah membawa kemajuan besar dalam penelitian NLP berdasarkan model yang telah dilatih sebelumnya. Terlepas dari kenyataan bahwa model berbasis perhatian memerlukan data dan GPU, pemrosesan paralel dan mekanisme perhatian lengkap *Transformer* kompatibel dengan tingkat tinggi komputasi paralel yang tersedia dalam komputasi bertenaga GPU, meningkatkan akurasi sintaksis kalimat teks. dan memiliki terbukti sangat efektif dalam semantik. Khususnya, model bahasa berbasis *Transformer* yang telah dilatih sebelumnya pada sejumlah besar teks untuk tugas-tugas sederhana seperti prediksi kata bertopeng dan

prediksi urutan kalimat telah dengan cepat menjadi populer dalam NLP. Hal ini dilakukan dengan tugas-tugas seperti: Ringkasan, analisis sintaksis, analisis wacana. Namun, dampak ini terutama terkonsentrasi pada bahasa-bahasa dengan tuntutan sumber daya yang tinggi seperti bahasa Inggris. bahasa Inggris.

Untuk melatih model yang telah dilatih sebelumnya secara efektif, penting untuk menggunakan bahasa tertentu. Karena kumpulan data dari ketiga file csv ini berisi ulasan pengguna wisata dalam bahasa Indonesia, maka dari itu diperlukan model yang telah dilatih sebelumnya untuk bahasa Indonesia. Hasilnya, model berbasis BERT telah ditingkatkan secara signifikan, yang mengarah pada pengembangan IndoBERT(Koto et al., 2020). Versi yang ditingkatkan ini dibangun berdasarkan kosakata bahasa Indonesia, yang dicapai dengan memodifikasi kerangka kerja Huggingface. IndoBERT telah dilatih dengan cermat pada kumpulan data ekstensif yang terdiri dari lebih dari 220 juta kata, yang bersumber dari berbagai platform Indonesia, termasuk Wikipedia bahasa Indonesia, artikel berita dari Kompas, Tempo, Liputan6, dan Korpus Web Indonesia. Proses pelatihan melibatkan menjalankan IndoBERT melalui 2,4 juta langkah atau 180 periode, yang membutuhkan waktu sekitar dua bulan untuk menyelesaikannya(Imaduddin et al., 2023). Atribut positif dari IndoBERT memberi motivasi menggunakan model ini digunakan untuk Analisis Sentimen Berbasis Aspek wisata kota tegal. Untuk penelitian ini, secara khusus menggunakan "IndoBERT-base-p2," yang mewakili salah satu varian model IndoBERT dari Huggingface.

2.1.4.2. Arsitektur IndoBERT

IndoBERT adalah varian BERT berbahasa Indonesia yang dikembangkan oleh huggingface. Model ini menggunakan arsitektur yang pertama kali diperkenalkan oleh (Vaswani et al., 2023). Arsitektur model yang dibangun menggunakan model transformer pada BERT secara umum yang menggunakan bahasa Inggris. Seperti pada BERT, metode ini juga menggunakan dua belas hidden layer dimana setiap hidden layer dibatasi pada dimensi 786 dan juga menggunakan 12 *attention head*. IndoBERT juga dibangun menggunakan kosakata bahasa Indonesia yang jumlahnya banyak dengan total lebih dari dua ratus dua puluh juta kata dimana sumber yang didapatkan adalah sumber bahasa yang menggunakan bahasa Indonesia yang baik dan benar seperti dari koran online dan juga dari Web Corpus Indonesia dan sumber lainnya. IndoBERT sendiri merupakan model pre-trained yang telah dikembangkan sebelumnya dengan menggunakan 2,4 juta langkah atau 180 epoch. Proses training memakan waktu hingga dua bulan. Sehingga IndoBERT memiliki performa yang baik untuk beberapa task pada kasus NLP(Juarto, 2023).

2.1.4.3. NLTK (*Natural Language Toolkit*)

NLTK dirancang dengan empat tujuan utama:

1. Kesederhanaan Untuk menyediakan kerangka kerja yang intuitif bersama dengan blok bangunan yang substansial, memberikan pengguna pengetahuan praktis tentang NLP tanpa terjebak dalam pekerjaan rumah

yang membosankan yang biasanya terkait dengan pemrosesan data bahasa beranotasi

2. Konsistensi Untuk menyediakan kerangka kerja yang seragam dengan antarmuka yang konsisten dan struktur data,
3. Ekstensibilitas Untuk menyediakan struktur di mana modul perangkat lunak baru dapat dengan mudah diakomodasi, termasuk implementasi alternatif dan pendekatan yang bersaing untuk tugas yang sama
4. Modularitas Untuk menyediakan komponen yang dapat digunakan secara independen tanpa perlu memahami toolkit lainnya (Bird, 2009).

Kontras dengan tujuan-tujuan ini adalah tiga kualitas yang tidak perlu persyaratan - kualitas yang berpotensi berguna yang sengaja kami hindari. Pertama, meskipun toolkit ini menyediakan berbagai macam fungsi, toolkit ini tidak bersifat ensiklopedis; toolkit ini adalah sebuah toolkit, bukan sebuah sistem, dan akan terus berkembang seiring dengan perkembangan NLP. Kedua, meskipun toolkit ini cukup efisien untuk mendukung tugas-tugas yang berarti, toolkit ini tidak terlalu dioptimalkan untuk kinerja runtime. pengoptimalan seperti itu sering kali melibatkan algoritme yang lebih kompleks, atau implementasi dalam bahasa pemrograman tingkat rendah seperti C atau C++. Hal ini akan membuat perangkat lunak menjadi kurang mudah dibaca dan lebih sulit untuk diinstal. Ketiga, kami telah mencoba untuk menghindari trik pemrograman yang pintar, karena kami percaya bahwa implementasi yang jelas lebih disukai daripada yang cerdas namun tidak dapat dipahami (Bird, 2009)

2.1.5. CRISP-DM dan Tahapan Merancang Sistem

Pada penelitian ini, digunakannya metodologi data mining CRISP-DM sebagai pemecah masalah yang diperlukan untuk menemukan aspek aspek dari komentar beberapa wisata di kota Tegal. Metodologi ini terdiri dari enam tahapan yaitu *Business Understanding*, *Data Understanding*, *Data Preparation*, *Modelling*, *Evaluation*, dan *Deployment*. Menurut (Hasanah et al., 2021) Proses metodologi ini terdiri dari 6 tahapan yang dapat dijelaskan sebagai berikut:

1. *Business Understanding* (Pemahaman Bisnis)

Beberapa hal yang dilakukan pada tahap ini seperti memahami kebutuhan serta tujuan dari sudut pandang bisnis selanjutnya mengartikan pengetahuan ke dalam bentuk pendefinisian masalah pada data mining dan kemudian menentukan rencana serta strategi untuk mencapai tujuan data mining.

2. *Data Understanding* (Pemahaman Data)

Tahapan ini diawali dengan mengumpulkan data, mendeskripsikan data, serta mengevaluasi kualitas data.

3. *Data Preparation* (Persiapan Data)

Dalam tahapan ini yaitu membangun dataset akhir dari berupa data mentah. Ada beberapa hal yang akan dilakukan mencakup melakukan pembersihan data (*Data Cleaning*), melakukan pemilihan data (*Data Selection*), record dan atribut-atribut, dan juga melakukan transformasi terhadap data (*Data Transformation*) untuk dijadikan masukan dalam tahap pemodelan. Penelitian ini

menggunakan *Stemmer* dari Sastrawi untuk menemukan aspek aspek dan nltk untuk preprocessing.

4. *Modelling* (Pemodelan)

Pada tahapan ini secara langsung melibatkan *Large Leanguage Model* (LLM) dan *pre train* model analisis sentimen berbahasa indonesia, penentuan teknik model, alat bantu serta algoritma yang digunakan. Penelitian ini menggunakan model bert-base-indonesian-1.5G dari github untuk menentukan sentimennya

5. *Evaluation* (Pengujian)

Tahap ini dilakukan dengan melihat tingkat performa dari pola yang dihasilkan oleh algoritma. Parameter yang digunakan untuk evaluasi komparasi algoritma adalah *Confusion Matrix* dengan aturan nilai akurasi, presisi dan recall.

6. *Deployment* (Penyebaran)

Tahapan ini dilakukan dengan pembuatan laporan dan artikel jurnal menggunakan model yang dihasilkan.





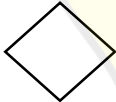
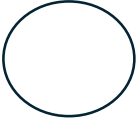
2.1.6. Pemodelan UML

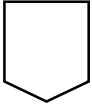
2.1.6.1. *Flowchart*

Flowchart adalah bagan yang menampilkan alir (flow) dari program atau sebuah prosedur sistem yang dibangun menggunakan simbol simbol. Flowchart berisi simbol-simbol yang menunjukkan alur instruksi sistem yang berjalan berurutan. Berikut adalah sebuah bagan yang digunakan untuk menggambarkan suatu proses

dalam program dengan simbol simbol tertentu. Simbol-simbol flowchar dapat dilihat dalam tabel 2.1 berikut:

Tabel 2. 1 Simbol Flowchart Diagram

Gambar	Simbol untuk	
	Terminator	Simbol yang digunakan untuk menunjukkan awal atau akhir program
	Garis Alir	Simbol yang digunakan untuk menunjukkan alur atau aliran program
	Proses	Simbol yang digunakan untuk proses pengolahan data
	Input Ouput Data	Simbol yang digunakan untuk memasukkan dan mengeluarkan data
	<i>Desicion</i>	Simbol yang digunakan untuk memberikan pilihan
	On Page Connector	Simbol yang digunakan untuk menghubungkan bagian-bagian flowchart dalam halaman yang sama

	Off Page Connector	Simbol yang digunakan untuk menghubungkan bagian-bagian flowchart dalam halaman yang berbeda
---	-----------------------	--

2.1.6.2. Use Case Diagram

Use Case adalah dialog antara aktor dengan sistem yang akan menggambarkan fungsi yang diberikan oleh sistem. Use Case merupakan pola atau bentuk terhadap perilaku yang menunjukkan sistem. Setiap Use Case merupakan sebuah deretan dari transaksi-transaksi yang terkait dari sebuah aktor dan sistem dalam sebuah dialog. (Risti, 2022). Simbol simbol yang ada pada Use Case dapat dilihat dalam Tabel 2.2 berikut:

Tabel 2. 2 Simbol Simbol *Use Case Diagram* (Astuti, 2009)

Simbol	Deskripsi
Use Case	fungsi yang disediakan sistem sebagai unit-unit yang saling bertukar pesan antar unit atau aktor; biasanya dinyatakan dengan menggunakan kata kerja di awal di awal frase nama <i>use case</i>
Aktor/ <i>actor</i>	orang, proses, atau sistem lain yang berinteraksi dengan sistem informasi

	<p>yang akan dibuat di luar sistem informasi yang akan dibuat itu sendiri, jadi walaupun simbol dari aktor adalah gambar orang, tapi aktor belum tentu merupakan orang; biasanya dinyatakan menggunakan kata benda di awal frase nama aktor. Aktor merupakan peran yang dimainkan oleh pemakai ketika berinteraksi dengan sistem.</p>
Asosiasi/ <i>Association</i>	<p>komunikasi antara aktor dan <i>use case</i> yang berpartisipasi pada <i>use case</i> atau <i>use case</i> memiliki interaksi dengan a</p>
Extensi/ <i>Extension</i>	<p>relasi <i>use case</i> tambahan ke sebuah <i>use case</i> dimana <i>use case</i> yang ditambahkan dapat berdiri sendiri walau tanpa <i>use case</i> tambahan itu; mirip dengan prinsip inheritance pada pemrograman berorientasi objek; biasanya <i>use case</i> tambahan memiliki nama depan yang sama dengan <i>use case</i> yang ditambahkan.</p>



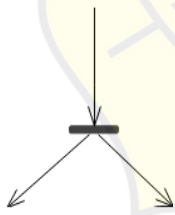
Generalisasi/ <i>Generalisation</i>	<p>Hubungan generalisasi dan spesialisasi (umum - khusus) antara dua buah <i>use case</i> dimana fungsi yang satu adalah fungsi yang lebih umum dari lainnya.</p>
Menggunakan / <i>include</i> / <i>uses</i>	<p>relasi use case tambahan ke sebuah use case dimana use case yang ditambahkan memerlukan <i>use case</i> ini untuk menjalankan fungsinya atau sebagai syarat dijalankan <i>use case</i> ini ada dua sudut pandang yang cukup besar mengenai <i>include</i> di <i>use case</i>:</p> <ul style="list-style-type: none"> • <i>include</i> atau <i>uses</i> berarti <i>use case</i> yang ditambahkan akan selalu dipanggil saat <i>use case</i> tambahan dijalankan • <i>include</i> berarti <i>use case</i> yang tambahan akan selalu melakukan pengecekan apakah <i>use case</i> yang ditambahkan telah dijalankan sebelum <i>use case</i> tambahan dijalankan.

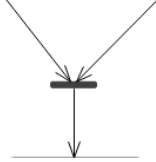

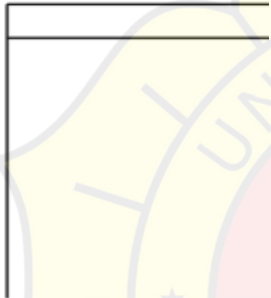
	Arah panah include mengarah pada <i>use case</i> yang dipakai
--	---

2.1.6.3. *Activity Diagram*

Activity Diagram menggambarkan workflow (aliran kerja) dari sebuah sistem atau proses bisnis atau menu yang ada pada perangkat lunak (Winda Aprianti, 2016). Simbol simbol dari *activity* diagram dapat dilihat pada tabel 2.3 berikut:

Tabel 2. 3 Tabel Activity Diagram (Hendin, 2016)

Gambar	Keterangan
	Start Point, diletakkan pada pojok kiri atas dan merupakan awal aktivitas
	End Point, akhir aktivitas
	Fork, Percabangan, digunakan untuk menunjukkan kegiatan yang dilakukan secara paralel atau untuk menggabungkan dua kegiatan paralel tersebut

	Join (Penggabungan), atau Rake, digunakan untuk menunjukkan adanya dekomposisi
	Desicion Point, menggambarkan pilihan untuk mengambil keputusan, true atau false
	Swimlane, pembagian activity diagram untuk menunjukkan siapa melakukan apa

2.1.7. Software dan Pemograman terkait

2.1.7.1. Pemrograman Python

Python adalah salah satu bahasa pemrogramam tingkat tinggi. Python terkenal oleh kalangan programmer karena penggunaannya yang lebih sederhana dari bahasa pemrograman lainnya. Selain itu, Python memiliki struktur sintak yang rapi dan mudah dipahami oleh programmer. Menurut (I Gede Harjumawan Wiratmaja Ks, 2021) Bahasa pemrograman ini menjadi umum digunakan untuk kalangan engineer seluruh

dunia dalam pembuatan perangkat lunaknya. Perangkat lunak yang dibuat bertujuan untuk mempermudah kehidupan manusia.

2.1.7.2. Google Colaboratory

Notebook Colaboratory didasarkan pada Jupyter dan berfungsi sebagai objek Google Docs: dapat dibagikan dan pengguna dapat berkolaborasi pada notebook yang sama. *Colaboratory* menyediakan runtime Python 2 dan 3 yang telah dikonfigurasi sebelumnya dengan pustaka pembelajaran mesin dan kecerdasan buatan yang penting, seperti TensorFlow, Matplotlib, dan Keras. Mesin virtual di bawah runtime (VM) dinonaktifkan setelah jangka waktu tertentu, dan semua data serta konfigurasi pengguna akan hilang. Namun demikian, notebook tetap dipertahankan, dan juga memungkinkan untuk mentransfer file dari hard disk VM ke akun Google Drive pengguna. Terakhir, layanan Google ini menyediakan runtime yang dipercepat dengan GPU, juga dikonfigurasi sepenuhnya dengan perangkat lunak yang telah diuraikan sebelumnya. Infrastruktur Google *Colaboratory* dihosting di platform Google *Cloud* (Carneiro et al., 2018)

2.1.7.3. Amazon Sagemaker AWS

Amazon SageMaker adalah layanan yang dikelola sepenuhnya yang memungkinkan ilmuwan dan pengembang data dengan cepat dan mudah membangun, melatih, dan menggunakan model pembelajaran mesin pada skala apa pun. Amazon SageMaker mencakup modul yang dapat digunakan bersama atau

secara mandiri untuk membangun, melatih, dan menerapkan model pembelajaran mesin.(Mardianto et al., 2020)

2.1.7.4. Django

Django adalah sebuah bahasa untuk pengembangan *web framework* berbasis bahasa pemrograman Python yang didesain untuk membuat aplikasi web yang dinamis, kaya fitur dan aman. Django yang dikembangkan oleh *Django Software Foundation* terus mendapatkan perbaikan sehingga membuat web framework yang satu ini menjadi pilihan utama bagi banyak pengembang aplikasi web.(Saputra & Aji, 2018). Django adalah kerangka kerja Python Web tingkat tinggi yang mendorong pengembangan cepat dan efisien. Tujuan utama Django adalah untuk memudahkan pembuatan situs web yang digerakkan oleh basis data yang kompleks. Beberapa situs terkenal yang menggunakan Django termasuk PBS, Instagram, Disqus, Washington Times, Bitbucket dan Mozilla. Kelebihan dari Django dibandingkan framework lain diantaranya(Risyda & Nuryamin, 2014):

- a) Merepresentasikan data model dari basis data yang dimuat dalam bentuk objek, sesuai dengan konsep pemrograman kekinian yang berbasis *object - oriented programming* (OOP).
- b) Django memahami situasi ketika kita harus menjaga kerahasiaan dan integritas data, Django menyediakan admin *interface* secara otomatis, tanpa perlu repot membuatnya. Halaman admin ini tidak boleh diakses oleh sembarang orang dan membutuhkan waktu yang relatif lama jika membuatnya sendiri.

- c) Pembuatan URL yang lebih mudah dan fleksibel.
- d) Sistem template yang powerful dan mudah dipelajari baik itu untuk *web developer* maupun web design serta mendukung *template hierarchy* (pewarisan).
- e) Sistem chace dapat digunakan dengan mudah.
- f) Mempermudah web developer dalam membuat web multibahasa.
- g) Tidak perlu menginstall web server selama proses development, cukup dijalankan saja.

2.1.7.5. React

React Js adalah sebuah *library* JavaScript yang di buat dan pertama kali dikembangkan oleh facebook. React JS adalah Library Javascript yang bersifat Open-Source yang mayoritas digunakan untuk membangun *User Interface* (UI) secara spesifik untuk satu halaman dalam aplikasi. ReactJS biasanya digunakan sebagai salah satu framework aplikasi untuk membuat bagian front-end dari sebuah aplikasi. React JS berfungsi untuk mengatur Layer pada tampilan untuk Desktop maupun versi mobile Aplikasi. Dengan React, user bisa membuat Reusable Component atau jenis komponen yang bisa digunakan kembali tanpa harus membangunnya dari awal lagi (Sulistyorini et al., 2022). Popularitasnya dapat diukur oleh aplikasi – aplikasi yang menggunakannya seperti Facebook, WhatsApp, Netflix, Instagram, Airbnb, American Express, Dropbox, Ebay, dan ratusan penyedia jasa pembuatan aplikasi berbasis web memanfaatkan kemampuan React JS(Sulistyorini et al., 2022).

2.1.8. Dataset dan sumber data

Dataset terdiri dari tiga tempat wisata yang berbeda yaitu Guci, Praban Lintang dan Pantai Alam Indah. Pada ketiga data tersebut masih memiliki beberapa permasalahan seperti data duplikasi komentar, data kosong dan kalimat asing yang belum diterjemahkan. Pada dataset Guci memiliki 4717 total komentar keseluruhan komentar dari wisata tersebut, lalu pada dataset Praban Lintang memiliki total 1583 total komentar keseluruhan komentar dari wisata itu, dan terakhir adalah dataset PAI yang memiliki 2484 total komentar keseluruhan dari wisata Pantai tersebut. ketiga file csv dataset tersebut didapatkan dari API google maps dan Travel komentar komentar wisata masing masing dari website google travel.

2.2. Kajian Penelitian Terdahulu

2.2.3 A SYSTEMATIC REVIEW OF ASPECT-BASED SENTIMENT ANALYSIS: DOMAINS, METHODS, AND TRENDS

Paper ini berjudul "*A Systematic Review of Aspect-Based Sentiment Analysis: Domains, Methods, and Trends*" ditulis oleh Yan Cathy Hua, Paul Denny, Katerina Taskova, dan Jörg Wicker. Paper ini dipublikasikan pada jurnal *Artificial Intelligence Review* dengan DOI [10.1007/s10462-024-10906-z](https://doi.org/10.1007/s10462-024-10906-z) dan diterbitkan pada tahun 2024.

2.2.3.1 Tujuan Penelitian

Penelitian ini bertujuan untuk memberikan tinjauan literatur sistematis (*Systematic Literature Review* - SLR) mengenai *Aspect-Based Sentiment Analysis* (ABSA), dengan fokus pada domain penelitian, formulasi permasalahan, dan pendekatan solusi. Tujuan utama adalah untuk mengidentifikasi tren dan pola tingkat

tinggi dalam penelitian ABSA, serta mengevaluasi keterkaitan antar domain aplikasi, dataset, dan metodologi solusi.

2.2.3.2 Metodologi yang Digunakan

Studi ini melibatkan analisis terhadap 727 penelitian utama yang dipilih dari 8550 hasil pencarian tanpa batasan waktu publikasi, menggunakan proses penyaringan otomatis berbasis *PDF-mining*. Penelitian ini juga melakukan analisis kuantitatif terhadap tren selama dua dekade terakhir, termasuk distribusi dataset dan domain aplikasi ABSA. Solusi yang dianalisis mencakup berbagai pendekatan, mulai dari metode linguistik tradisional hingga pembelajaran mendalam berbasis model seperti BERT dan GPT.

2.2.3.3 Temuan Utama

Hasil penelitian menunjukkan dominasi dataset berbasis ulasan produk dan layanan (70,95%) dibandingkan dengan domain lain seperti kesehatan, politik, dan pendidikan. Selain itu, pendekatan berbasis pembelajaran mendalam telah mendominasi penelitian ABSA sejak 2017, terutama melalui model *Recurrent Neural Networks* (RNN) dan mekanisme perhatian (*attention mechanism*). Namun, penelitian ini juga mengungkapkan adanya kekurangan keragaman dataset dan ketidaksesuaian antara domain penelitian dan dataset, yang dapat membatasi pengembangan solusi ABSA yang lebih luas.

2.2.3.4 Kesimpulan

Penelitian ini menyimpulkan bahwa pengembangan ABSA sangat bergantung pada domain dan dataset tertentu, terutama dataset yang telah mapan seperti SemEval.

Untuk mengatasi batasan ini, diperlukan dataset yang lebih beragam dan solusi berbasis pembelajaran lintas domain (*domain adaptation*). Studi ini juga menyoroti potensi besar dari model bahasa generatif modern (*generative large language models*), tetapi implementasinya pada ABSA masih membutuhkan eksplorasi lebih lanjut.

2.2.4 Survei Metode dan Model ABSA

Paper ini berjudul "*A Survey on Aspect-Based Sentiment Analysis: Tasks, Methods, and Challenges*" ditulis oleh Wenxuan Zhang, Xin Li, Yang Deng, Lidong Bing, dan Wai Lam. Paper ini diterbitkan sebagai survei komprehensif pada jurnal terkait di tahun 2022.

2.2.4.1 Tujuan Penelitian

Penelitian ini bertujuan untuk mengulas berbagai tugas, metode, dan tantangan dalam *Aspect-Based Sentiment Analysis* (ABSA), meliputi tugas-tugas seperti ekstraksi aspek, klasifikasi sentimen, serta hubungan antar elemen sentimen.

2.2.4.2 Metodologi yang Digunakan

Dalam survei ini, penulis menggunakan pendekatan literatur komprehensif dengan merujuk pada berbagai dataset dan model yang telah diterapkan, termasuk model berbasis BERT dan transformers. Data yang dikaji berasal dari berbagai dataset seperti SemEval-2014 untuk ulasan produk dan layanan, sehingga dapat menunjukkan potensi solusi pada ABSA.

2.2.4.3 Temuan Utama

Hasil survei menunjukkan bahwa model ABSA modern, termasuk yang berbasis BERT, mencapai skor F1 sekitar 70%-80% pada dataset benchmark.

Tantangan utama dalam domain transfer dan kendala bahasa dibahas secara mendalam, dengan saran solusi pada penggunaan model pra-latih.

2.2.4.4 Kesimpulan

Survei ini mengidentifikasi bahwa model ABSA modern telah mencapai performa yang cukup tinggi, namun masih terdapat tantangan besar dalam hal transfer lintas domain dan bahasa yang memerlukan pendekatan khusus.

2.2.5 BERT Post-Training untuk Pemahaman Ulasan dan ABSA

Paper ini berjudul "*Bert Post-Training for Review Reading Comprehension and Aspect-Based Sentiment Analysis*" oleh Hu Xu, Lei Shu, Philip S. Yu, dan Bing Liu, diterbitkan pada tahun 2019 di konferensi terkait.

2.2.5.1 Tujuan Penelitian

Tujuan penelitian ini adalah untuk memahami representasi internal BERT dan mekanisme perhatian yang diterapkan dalam tugas ABSA dan pemahaman ulasan.

2.2.5.2 Metodologi yang Digunakan

Data latih dalam penelitian ini adalah ulasan gabungan dari Amazon dan Yelp dengan ukuran lebih dari 20 GB, yang kemudian diolah dengan fine-tuning BERT BASE selama 4 epoch. Pelatihan dilakukan menggunakan satu GPU TITAN RTX Analisis dilakukan pada 150 sampel dari domain laptop dan restoran dari dataset SemEval 2014.

2.2.5.3 Temuan Utama

Penelitian ini menemukan bahwa hanya 2 hingga 4 dari 144 self-attention head BERT yang secara konsisten mampu mengodekan kata-kata konteks atau opini terkait

aspek. Skor F1 pada klasifikasi sentimen aspek mencapai 83%, dengan temuan bahwa aspek representasi didominasi oleh semantik domain dan aspek, bukan sentimen.

2.2.5.4 Kesimpulan

Penelitian ini menunjukkan bahwa post-training BERT dapat meningkatkan kemampuan ABSA, namun menemukan bahwa representasi aspek lebih berfokus pada makna semantik daripada sentimen, memberikan wawasan untuk peningkatan di masa depan.

