BAB 1

PENDAHULUAN

1.1 Latar Belakang

Perkembangan teknologi kecerdasan buatan (*Artificial Intelligence*) dalam beberapa tahun terakhir mengalami peningkatan yang sangat signifikan, khususnya pada bidang *Natural Language Processing* (*NLP*). Teknologi ini memungkinkan komputer untuk memahami, memproses, dan merespon bahasa manusia secara lebih alami dan kontekstual. Salah satu implementasi dari *NLP* yang berkembang pesat adalah *chatbot*, yang kini banyak digunakan dalam berbagai sektor seperti layanan pelanggan, pendidikan, kesehatan, hingga pertanian[1].

Natural Language Processing (*NLP*) adalah cabang ilmu kecerdasan buatan yang memfokuskan pemrosesan bahasa alami yang umumnya digunakan oleh manusia untuk berkomunikasi. Komputer membutuhkan waktu untuk memahami bahasa yang diberikan agar dapat berada sejalan dengan maksud pengguna. Berbagai implementasi dari *NLP* mencakup *chatbot* (aplikasi percakapan di mana pengguna memungkinkan interaksi dengan komputer), *stamming* atau *lemmatization* (proses pemangkasan kata dalam suatu bahasa ke bentuk dasar guna mengidentifikasi fungsi setiap kata dan kalimat), *summarization* (rangkuman dari teks bacaan), *translation tools* (alat penerjemah bahasa), dan berbagai aplikasi yang memberi kemungkinan komputer untuk memahami perintah-perintah dalam bahasa yang di *input* oleh pengguna[2].

Menurut penelitian terdahulu oleh Nguyen *chatbot* berbasis *NLP* mampu meningkatkan efisiensi layanan informasi hingga 45% dibandingkan sistem

konvensional. *BERT* (*Bidirectional Encoder Representations from Transformers*) adalah model bahasa yang revolusioner dalam pemrosesan bahasa alami, mengandalkan *encoder* untuk menghasilkan representasi kontekstual dari teks *input*. Melalui pendekatan tokenisasi, *embedding*, dan *attention mechanisms* pada setiap layer transformer, *BERT* memungkinkan pemahaman hubungan antar kata secara mendalam dan bidireksional. Keunikan *BERT* terletak pada kemampuannya untuk memproses konteks dari kedua arah, menciptakan representasi vektor yang kaya makna[3]. Penelitian oleh Kumar & Singh juga menunjukkan bahwa penggunaan model berbasis transformer seperti *BERT* mampu meningkatkan akurasi pemahaman bahasa alami hingga lebih dari 90%.

HA Simon berpendapat sebuah kecerdasan buatan (*AI*) adalah bidang studi, aplikasi, dan pelatihan pemrograman komputer untuk melakukan hal-hal yang dianggap cerdas oleh manusia. Saat ini banyak pemanfaatan *AI* yang digunakan dalam berbagai bidang teknologi salah satunya yaitu *chatbot ChatGPT*[4]. Selain itu, Peneliti oleh Zhang menyatakan bahwa *chatbot* berbasis *AI* dapat mengurangi beban kerja manusia dalam penyampaian informasi secara signifikan.

Dalam konteks perkebunan kelapa sawit, kebutuhan akan sistem informasi yang cepat, akurat, dan mudah diakses menjadi sangat penting. Industri kelapa sawit merupakan salah satu sektor perkebunan strategis nasional dengan kontribusi besar terhadap perekonomian Indonesia[5]. Industri kelapa sawit tidak hanya menjadi penyumbang devisa negara, tetapi juga berkontribusi terhadap penyerapan tenaga kerja dan peningkatan kesejahteraan masyarakat, khususnya di wilayah pedesaan[6]. Penelitian oleh Rahman menunjukkan bahwa petani sering mengalami

kesulitan dalam mendapatkan informasi teknis terkait pengelolaan tanaman, terutama dalam hal diagnosis penyakit dan pemupukan. Hal ini berdampak pada penurunan produktivitas.

Berdasarkan hasil pengamatan lapangan yang dilakukan peneliti, khususnya pada petani swadaya di Desa Batang Kumu, ditemukan bahwa dalam pengelolaan perkebunan masih bersifat *konvensional* dan belum berbasis data. Pertama, dalam aspek pemupukan, banyak petani yang memberikan dosis pupuk tidak sesuai dengan umur dan kondisi tanaman karena tidak memiliki data acuan yang jelas. Hal ini menyebabkan biaya produksi membengkak namun produktivitas *TBS* tidak optimal. Kedua, pada penanganan hama dan penyakit, petani cenderung terlambat melakukan tindakan pengendalian karena sulit membedakan gejala awal serangan Ganoderma, ulat api, atau busuk pucuk. Keterlambatan ini mengakibatkan kehilangan hasil panen mencapai 15-30% per tahun. Minimnya akses terhadap penyuluh pertanian yang jumlahnya terbatas, membuat petani hanya mengandalkan pengalaman pribadi atau informasi dari sesama petani yang belum tentu akurat. Ketiga, dalam penentuan waktu panen dan rotasi panen, keputusan masih bersifat tradisional. Petani memanen berdasarkan perkiraan warna buah tanpa mempertimbangkan kriteria matang panen *CPO* >22%. Akibatnya, rendemen *CPO* yang dihasilkan pabrik rendah dan harga jual *TBS* petani menjadi turun.

Melihat kompleksitas permasalahan tersebut, diperlukan suatu solusi berbasis teknologi yang mampu mengintegrasikan berbagai variabel penting dalam pengelolaan kelapa sawit, seperti umur tanaman, gejala visual daun, kondisi cuaca, harga pupuk, serta riwayat serangan hama dan penyakit. Teknologi tersebut

diharapkan dapat menghasilkan rekomendasi yang objektif, cepat, dan mudah diakses oleh petani.

Seiring dengan perkembangan teknologi informasi, pendekatan berbasis *Natural Language Processing* (*NLP*) menjadi salah satu alternatif yang potensial untuk membantu proses pengambilan keputusan. Salah satu model yang banyak digunakan adalah *Bidirectional Encoder Representations from Transformers* (*BERT*), yang memiliki kemampuan memahami konteks bahasa secara lebih mendalam. Oleh karena itu, penelitian ini bertujuan untuk merancang dan membangun *chatbot* berbasis *NLP* dengan pemodelan *BERT* dalam pengelolaan perkebunan kelapa sawit berbasis *web*. Sistem ini diharapkan dapat menjadi asisten digital bagi petani swadaya, khususnya di Desa Batang Kumu, dalam memperoleh rekomendasi yang akurat dan berbasis data untuk meningkatkan produktivitas dan efisiensi usaha tani kelapa sawit.

Penelitian oleh Li mengungkapkan bahwa penerapan sistem berbasis *AI* dalam pertanian dapat meningkatkan produktivitas hingga 30%. *Chatbot* merupakan pengembangan aplikasi komputer yang dirancang untuk dapat berinteraksi dengan manusia melalui pesan teks, maupun suara. *Chatbot* telah dibekali dengan kecerdasan buatan dan pemrosesan bahasa alami (*NLP*) yang membuatnya menjadi aplikasi komputer yang cerdas dan dapat menjawab pertanyaan yang diberikan oleh manusia. *Chatbot* dibangun untuk membantu manusia dalam hal pelayanan informasi (*customer service*), dengan topik yang sudah di tetapkan. Banyak *chatbot* yang sudah ada dibangun sesuai dengan topik dan permasalahan yang ingin dipecahkan oleh seseorang untuk keperluan pribadi

ataupun keperluan bisnis lainnya. Di dalam *chatbot* tersebut telah ditanamkan model pengetahuan untuk menjawab pertanyaan-pertanyaan yang sesuai dengan konteks yang telah disusun[7]. Selanjutnya, penelitian oleh Ahmad menunjukkan bahwa *chatbot* dapat digunakan sebagai media edukasi digital yang efektif bagi petani dalam memahami kondisi tanaman.

Berdasarkan kajian penelitian terdahulu, sebagian besar penelitian *chatbot* berbasis *NLP* masih berfokus pada penyediaan layanan informasi atau tanya jawab (*FAQ*) secara umum. Penelitian tersebut belum banyak mengintegrasikan kemampuan analisis kondisi pengguna untuk menghasilkan rekomendasi sebagai bentuk dukungan pengambilan keputusan, khususnya pada sektor pertanian kelapa sawit. Dalam konteks penelitian ini, *NLP* memiliki peran penting karena petani sering menggunakan variasi bahasa yang beragam, termasuk dialek lokal. Selain itu, pertanyaan yang diajukan oleh petani tidak hanya bersifat *informatif*, tetapi juga berkaitan dengan pengambilan keputusan. Contohnya seperti “daun sawit saya menguning, harus diapakan apa?”, “pupuk apa yang cocok untuk tanaman umur 3 tahun?”. Pertanyaan-pertanyaan tersebut memerlukan analisis kondisi serta rekomendasi tindakan yang tepat. Penelitian oleh Chen juga menegaskan bahwa integrasi *NLP* dan *decision support system* mampu membantu pengguna dalam mengambil keputusan yang lebih tepat berbasis data dan analisis.

Oleh karena itu, penelitian ini bertujuan untukmengimplementasikan *chatbot* berbasis *NLP* dengan pemodelan *BERT* yang tidak hanya berfungsi sebagai sistem informasi, tetapi juga bisa membantu petani swadaya dalam pengelolaan perkebunan kelapa sawit.

1.2 Rumusan Masalah

Berdasarkan latar belakang di atas rumusan masalah dari penelitian ini adalah, “bagaimana merancang dan mengimplementasikan *chatbot* menggunakan model *BERT* (*Bidirectional Encoder Representations from Transformers*) dalam pengelolaan perkebunan kelapa sawit?”.

1.3 Tujuan Penelitian

Berdasarkan rumusan masalah tersebut, maka tujuan dari penelitian ini adalah: “merancang dan mengimplementasikan *chatbot* menggunakan model *BERT* (*Bidirectional Encoder Representations from Transformers*) dalam pengelolaan perkebunan kelapa sawit”.

1.4 Batasan Masalah

Berdasarkan tujuan penelitian di atas maka penelitian ini penting untuk dibatasi pada hal-hal berikut.

1. Fokus utama *chatbot*, memberikan informasi pengelolaan kelapa sawit.
2. Penelitian ini dibatasi pada *implementasi chatbot* dengan menggunakan *Bidirectional Encoder Representations from Transformers* (*BERT*) untuk informasi pengelolaan Perkebunan kelapa sawit.
3. *Input* dan *output* dari program *chatbot* ini hanya berupa teks.
4. Bahasa *pemrograman* yang digunakan dalam pengembangan chatbot ini adalah *Python*.
5. *Chatbot* ini menggunakan *dataset* yang telah dikumpulkan dan diproses, berupa daftar pertanyaan dan juga jawaban yang relevan.

6. Chatbot hanya pengelolaan pemupukan, pengendalian hama, panen, dan umum.
7. Hanya menggunakan Bahasa Indonesia.

1.5 Manfaat Penelitian

Manfaat yang diperoleh dari pembuatan penelitian ini adalah sebagai berikut:

1. Manfaat *Teoritis*

   Penelitian ini diharapkan dapat memberikan wawasan, informasi, dan pemahaman yang mendalam mengenai penerapan *chatbot* dalam layanan pengelolaan kelapa sawit, sehingga mempermudah proses penyampaian informasi kepada masyarakat.

2. Manfaat *Praktisi*

   1) Bagi Peneliti Sebagai sarana untuk mengembangkan ilmu pengetahuan, keterampilan, dan kompetensi selama menjalani pendidikan di perguruan tinggi, yang akan diterapkan secara ilmiah dan terstruktur.
   2) Bagi petani kelapa sawit untuk memudahkan dan mempercepat dalam memperoleh informasi pengelolaan kelapa sawit dan menciptakan pengalaman layanan konsultasi yang efisien.

1.6 Sistematika Penelitian

Sistematika penelitian penelitian ini terdiri dari pokok-pokok permasalahan yang dibahas dan diuraikan menjadi beberapa bagian:

Laporan penelitian ini disusun dengan sistematika sebagai berikut:

BAB 1: PENDAHULUAN

Bab ini berisi uraian mengenai latar belakang penelitian, rumusan masalah, tujuan penelitian, batasan masalah, manfaat penelitian, metodologi penelitian, serta sistematika penulisan skripsi.

BAB 2: LANDASAN TEORI

Bab ini membahas teori-teori yang menjadi dasar dalam penelitian, Selain itu, bab ini juga memuat tinjauan penelitian terdahulu yang relevan sebagai bahan perbandingan dan pendukung penelitian.

BAB 3: METODOLOGI PENELITIAN

Bab ini menjelaskan jenis dan pendekatan penelitian, sumber dan teknik pengumpulan data, tahapan penelitian, serta teknik analisis data yang digunakan untuk mengkaji *chatbot* Berbasis *NLP* dengan pemodelan *BERT* untuk pengelolaan Perkebunan kelapa sawit.

BAB 4: ANALISIS DAN PERANCANGAN

Bab ini berisi analisis dan perancangan *chatbot* berbasis *NLP* dengan pemodelan *BERT*

BAB 5: IMPLEMENTASI DAN PENGUJIAN

Bab ini memaparkan proses pengolahan data secara lebih rinci, *interpretasi* hasil analisis, serta evaluasi terhadap temuan penelitian yang diperoleh dari data yang telah diklasifikasikan.

BAB 6: KESIMPULAN DAN SARAN

Bab ini berisi kesimpulan dari seluruh rangkaian penelitian yang telah dilakukan serta saran yang dapat diberikan untuk pengembangan kebijakan maupun penelitian selanjutnya.

BAB 2

LANDASAN TEORI

2.1 *Chatbot*

Chatbot merupakan program komputer yang dibuat untuk menirukan percakapan layaknya interaksi antarmanusia. Teknologi ini dilengkapi dengan kecerdasan buatan serta pemrosesan bahasa alami, sehingga mampu berfungsi sebagai sistem cerdas yang dapat memahami dan memberikan jawaban atas pertanyaan pengguna[8].

Dalam penelitian modern, *Chatbot* modern mampu memproses percakapan *multi-turn*, menafsirkan konteks kompleks, dan memberikan pengalaman interaktif yang menyerupai percakapan manusia nyata. Teknologi *AI* terbaru memungkinkan *chatbot* belajar secara berkelanjutan (*continuous learning*) dari interaksi sebelumnya, meningkatkan kemampuan memahami bahasa, konteks, dan preferensi pengguna. Selain itu, integrasi *chatbot* dengan *analitik* data, *system CRM,* dan manajemen layanan memungkinkan perusahaan memperoleh wawasan strategis, memprediksi kebutuhan pelanggan, menyesuaikan layanan secara *proaktif*, serta meningkatkan pengalaman pelanggan secara menyeluruh[9].

Chatbot banyak digunakan dalam berbagai sektor, terutama layanan pelanggan, pendidikan, dan sistem informasi. Keunggulan utama *chatbot* adalah kemampuannya dalam memberikan layanan secara *real-time*, konsisten, serta dapat beroperasi selama 24 jam tanpa henti. Hal ini menjadikan *chatbot* sebagai solusi efektif dalam meningkatkan efisiensi layanan dan mengurangi beban kerja manusia.

Selain itu, perkembangan *chatbot* berbasis *AI* juga menunjukkan bahwa integrasi dengan model *deep learning* dapat meningkatkan kualitas interaksi antara manusia dan sistem. Penelitian ini penting untuk mengidentifikasi secara *empiris* bagaimana *chatbot* berdampak pada kepuasan dan loyalitas pelanggan, serta faktor-faktor kunci yang menentukan keberhasilan implementasinya dalam berbagai konteks *industry*[10]. Dalam konteks penelitian ini, *chatbot* digunakan sebagai antarmuka utama untuk membantu pengguna dalam memperoleh informasi terkait pengelolaan perkebunan kelapa sawit.

2.2 *Natural Language Processing* (*NLP*)

Menurut Sulianta *Natural Language Processing* merupakan salah satu teknologi inti dalam kecerdasan buatan yang memungkinkan mesin mengolah bahasa alami melalui analisis *linguistik* dan pembelajaran *statistik*. Teknologi ini terdiri dari beberapa tahap, seperti *tokenization*, *part-of-speech tagging*, *parsing*, dan *semantic analysis* yang berfungsi memahami makna kalimat secara kontekstual[11].

1) *Tokenization*

   Dalam pemrosesan bahasa alami (*Natural Language Processing*) dan salah satu komponen utama dari *information extraction* yang bertujuan untuk mengidentifikasi dan mengklasifikasikan entitas yang tercantum pada suatu teks. Pada akhirnya, entitas yang telah dikenali akan ditandai dengan label atau tag khusus yang menunjukkan jenis entitas tersebut.

2) *part-of-speech tagging* (*POS*)

Part-of-speech (*POS*) *tagging* atau secara singkat dapat ditulis sebagai *tagging* merupakan proses pemberian penanda *POS* atau kelas sintaktik pada tiap kata di dalam *corpus*. Dikarenakan tag secara umum juga diaplikasikan pada tanda baca, maka dalam proses *tagging*, tanda baca seperti tanda titik, tanda koma, dll perlu dipisahkan dari kata-kata.

3) *Parsing*

parsing adalah suatu teknik dalam *natural language processing* (*NLP*) yang digunakan untuk mengidentifikasi hubungan *gramatikal* antara satu kata dengan kata lain dalam suatu kalimat. Teknik ini memperlihatkan hubungan antara satu kata dengan kata lain secara *visual*, sehingga memudahkan kita untuk memahami struktur kalimat dan makna yang terkandung di dalamnya.

4) *semantic analysis*

Menurut Liu *sentiment analysis* (analisis sentimen) atau sering disebut juga dengan *opinion mining* (penambangan opini) adalah studi komputasi untuk mengenali dan mengekspresikan opini, sentimen, evaluasi, sikap, emosi, subjektifitas, penilaian atau pandangan yang terdapat dalam suatu teks. Dave menjelaskan bahwa sebuah alat bantu penambangan opini merupakan pemrosesan sekumpulan hasil pencarian dari suatu item yang diberikan, menghasilkan satu daftar atribut produk (misal kualitas, fitur, dan lain-lain) dan menghitung agregasi dari opini dari masing-masing atribut tersebut (rendah, sedang, tinggi)[12].

Dalam perkembangan *Transformer* telah mendapatkan popularitas yang signifikan di bidang pemrosesan bahasa alami (*NLP*) karena penggunaannya yang

luas dalam banyak tugas, termasuk namun tidak terbatas pada penerjemahan bahasa, produksi teks, dan menjawab pertanyaan[13]. Hal ini menjadikan *NLP* lebih efektif dalam menangani data bahasa alami yang kompleks dan tidak terstruktur.

Dalam penerapan *NLP* untuk menyelesaikan suatu permasalahan, digunakan berbagai teknik, di antaranya:

1. *Sentence segmentation* adalah salah satu teknik dalam *NLP* yang berfungsi untuk mengidentifikasi unit proses berupa kumpulan kata. Tujuan dari teknik ini adalah untuk mendeteksi atau menentukan batas akhir dari sebuah kalimat.
2. *Tokenization Teknik* ini merupakan salah satu dasar dalam *NLP*, yang berfungsi dengan cara memecah kalimat menjadi token-token, di mana setiap token merepresentasikan sebuah kata.
3. *Stopword removal* adalah salah satu teknik dalam *NLP* yang bertujuan untuk menghilangkan kata-kata yang tidak membawa informasi penting, sehingga dapat meningkatkan akurasi hasil pemrosesan[14].

Dalam penelitian ini, *NLP* digunakan sebagai tahap awal untuk memproses *input* pengguna sebelum dilakukan klasifikasi menggunakan model berbasis *deep learning* yaitu *BERT*.

2.3 *Text Preprocessing*

Text Preprocessing memiliki peran yang sangat penting dalam sistem *Natural Language Processing* (*NLP*) karena tahap ini bertanggung jawab untuk mengidentifikasi karakter, kata, dan kalimat yang menjadi unit dasar yang akan

diolah pada seluruh tahap pemrosesan selanjutnya, mulai dari analisis *morfologi* hingga penandaan ucapan[15].

Tahapan *preprocessing* meliputi *case folding*, data *cleaning*, *tokenization*, *stopword removal*, serta *stemming* atau *lemmatization*. Setiap tahapan memiliki peran penting dalam meningkatkan kualitas data input sehingga model dapat bekerja lebih optimal[16].

1. *Case Folding* adalah teknik *praproses* teks yang sederhana namun sangat efektif, meskipun sering terlupakan. Teknik ini digunakan untuk mengatasi masalah ketidakonsistenan dalam penggunaan huruf besar dan kecil dalam data teks, yang sering kali tidak memiliki struktur yang terorganisir.
2. Tokenisasi atau *Tokenizing* adalah proses mengubah teks atau dokumen menjadi bagianbagian yang lebih kecil yang disebut token. Token tersebut dapat berupa kata-kata, frasa, tanda baca, atau entitas lainnya. Tujuan utama dari tokenisasi adalah untuk memungkinkan komputer memproses dan memahami teks dengan lebih efisien.
3. Dalam proses pembersihan data atau *Cleaning*, beberapa langkah penting dilakukan untuk memastikan kebersihan dan konsistensi data. Langkah-langkah tersebut meliputi mengisi nilai-nilai yang kosong agar tidak ada data yang hilang atau tidak terwakili, meratakan data yang tidak konsisten untuk memastikan format dan standar yang seragam, serta menangani gangguan data seperti nilai-nilai yang anomali atau tidak relevan sehingga kualitas data tetap terjaga dan analisis selanjutnya dapat dilakukan dengan lebih akurat.

4. *Stopword* removal adalah proses dalam pemrosesan teks yang bertujuan untuk menghapus kata-kata yang umum dan sering muncul dalam teks, namun memiliki sedikit atau bahkan tidak ada nilai *informatif* dalam analisis teks (Putra Negara, 2023). Kata-kata ini disebut *stopwords*. Proses *stopword removal* biasanya dilakukan sebagai salah satu tahapan dalam *preprocessing* teks sebelum analisis lebih lanjut dilakukan.
5. *Stemming* merupakan langkah dalam pemrosesan teks yang bertujuan untuk menghasilkan bentuk dasar atau kata dasar dari kata-kata yang terdapat dalam teks. Fokus utama dari proses *stemming* adalah untuk menyederhanakan variasi kata dalam teks dengan menghilangkan akhiran kata sehingga dapat diperoleh bentuk dasarnya. Dengan demikian, *stemming* membantu dalam mengonsolidasikan variasi *morfologis* dari katakata yang memiliki akar yang sama. Proses ini berkontribusi dalam meningkatkan konsistensi dan mempermudah analisis teks dalam berbagai penerapan Pemrosesan Bahasa Alami (*Natural Language Processing/NLP*).
6. *Lematisasi atau Lemmatization* adalah proses pada pemrosesan teks yang bertujuan untuk menghasilkan bentuk kata dasar atau kata lema dari kata-kata yang ada dalam suatu teks[17].

Tahap ini berfungsi untuk membersihkan data teks dari elemen-elemen yang tidak relevan, seperti simbol atau kata-kata yang kurang penting, sehingga menghasilkan teks yang lebih terstruktur[18].

2.4 *Intent Classification*

Intent classification merupakan proses dalam *NLP* yang bertujuan untuk mengidentifikasi maksud atau tujuan dari input pengguna. Dalam sistem *chatbot*, *intent classification* digunakan untuk menentukan jenis respons yang harus diberikan oleh sistem. *Chatbot* ini menggabungkan klasifikasi *intent* statis dengan kalkulasi data dinamis secara *hybrid*. Pada tahapan *intent classification* digunakan model statis (*Logistic Regression + TF-IDF*) yang sudah dilatih sebelumnya, tetapi tahapan pemrosesan data dilakukan secara *realtime* pada *dataset CSV*[19].

Model berbasis *deep learning* seperti *BERT* telah terbukti memiliki akurasi yang lebih tinggi dalam melakukan *intent classification* dibandingkan metode tradisional seperti *Naïve Bayes* atau *SVM*[20].

Selain itu, Metode *Naive Bayes* dan *SVM* bergantung pada jenis *dataset* yang digunakan serta parameter tertentu, seperti teknik ekstraksi fitur dan representasi kata. Selain itu, telah terbukti bahwa penggunaan *FastText* sebagai metode penerapan dapat meningkatkan efisiensi klasifikasi *sentiment*[21].

2.5 *Entity Extraction atau Named Entity Recognition* (*NER*)

Named Entity Recognition adalah sub-bidang dalam pemrosesan bahasa alami (*Natural Language Processing*) dan salah satu komponen utama dari *information extraction* yang bertujuan untuk mengidentifikasi dan mengklasifikasikan entitas yang tercantum pada suatu teks.

Dalam pengembangan sistem *chatbot*, *entity extraction* digunakan untuk memahami konteks spesifik dari pertanyaan pengguna sehingga sistem dapat memberikan jawaban yang lebih relevan dan tepat sasaran.

Penelitian terbaru menunjukkan bahwa model berbasis *transformer* seperti *BERT* dan *RoBERTa* mampu meningkatkan akurasi *entity recognition* secara signifikan dibandingkan pendekatan *rule-based*[22]. Hal ini disebabkan oleh kemampuan model dalam memahami konteks kalimat secara menyeluruh.

2.6 *Knowledge Base*

Knowledge Based System adalah suatu sistem yang menggunakan set pengetahuan (*knowledge*) yang dikodekan ke bahasa mesin untuk dapat membantu manusia dalam menyelesaikan masalah yang dihadapi[23].

Dalam penelitian produk, *Knowledge Based* merupakan metode sistem rekomendasi pada penelitian ini dengan kelebihan dapat memberikan skala prioritas pelanggan sesuai keinginan pelanggan terhadap produk[24].

Selanjutnya, *chatbot AI* memiliki basis pengetahuan (*knowledge base*) yang berisi informasi, data, atau aturan yang digunakan untuk menghasilkan jawaban. Basis pengetahuan ini dapat berupa kumpulan pertanyaan dan jawaban (*FAQ*), dokumen pembelajaran, atau data akademik lainnya. Selain itu, terdapat mesin *inferensi* atau modul pemrosesan yang bertugas untuk menentukan respon yang paling tepat berdasarkan hasil pemrosesan bahasa alami dan basis pengetahuan yang tersedia[25].

2.7 *Bidirectional Encoder Representations from Transformers* (*BERT*)

BERT adalah model representasi bahasa yang dikembangkan oleh *Google* pada tahun 2018 dan menjadi salah satu terobosan besar dalam bidang pemrosesan bahasa alami. Model ini memperkenalkan pendekatan baru yang mampu memahami konteks kalimat secara lebih akurat dibandingkan model-model sebelumnya[26].

Penelitian *BERT* memiliki performa yang lebih unggul dibandingkan metode machine learning tradisional dalam berbagai tugas *NLP* seperti:

1. *Text Classification*

   Klasifikasi teks (*text classification*) telah menjadi bidang yang banyak diteliti di bidang *Natural Language Processing* (*NLP*). Penerapan klasifikasi teks telah banyak dilihat dalam kehidupan sehari-hari. Salah satu penerapan klasifikasi teks yang dapat ditemui adalah proses pengklasifikasian teks berita berdasarkan kategori tertentu. Saat ini, terdapat dua metode yang paling umum digunakan dalam melakukan klasifikasi teks, yaitu menggunakan metode tradisional (*parametric algorithm*) dan *deep learning*[27].

2. *Question Answering*,

   Question answering adalah fondasi untuk menemukan jawaban yang tepat karena ia menentukan "apa" yang sebenarnya dicari oleh pengguna, sehingga sistem dapat memfokuskan pencariannya pada tipe entitas atau informasi yang relevan. Dalam alur kerja *QA* yang bertanggung

jawab untuk memahami maksud dan jenis informasi yang dicari oleh pengguna dari pertanyaan yang diajukan dalam bahasa alami[28].

3. *Natural Language Inference*

   Natural Language Inference (*NLI*), atau yang sering disebut juga sebagai *Recognizing Textual Entailment* (*RTE*), merupakan salah satu tugas penting dalam bidang *Natural Language Processing* (*NLP*). *NLI* bertujuan untuk menentukan hubungan logis antara dua kalimat, yaitu *premis* (kalimat utama) dan *hipotesis* (kalimat yang diuji terhadap premis)[29].

Hal ini disebabkan oleh kemampuan *BERT* dalam memahami konteks global dari sebuah kalimat.

Confusion matrix dapat digunakan untuk menghitung berbagai *performance metrics* yang berfungsi untuk mengukur kinerja model klasifikasi yang telah dibuat yaitu *accuracy, precision, recall, dan f1-score*.

1. *Accuracy*

   Accuracy adalah metrik yang digunakan untuk mengukur seberapa sering model prediksi memberikan hasil yang benar, baik itu untuk kelas positif maupun kelas negatif. *Accuracy* dihitung dengan membandingkan jumlah prediksi yang benar yaitu *True Positive* (*TP*) maupun *True Negative* (*TN*) dengan keseluruhan jumlah prediksi yang dibuat oleh model. Nilai *accuracy* 18 dalam konteks pembelajaran mesin dapat diperoleh dengan Persamaan berikut ini:

$$accuracy = \left(\frac{TP+TN}{TP+TN+FP+FN}\right) x\ 100 \qquad \ldots(1)$$

   Ket: (TP) *True Positive*

   (TN) *True Negative*

   (FP) *False Positive*

   (FN) *False Negative*

   Atau rumus umumnya dapat diperoleh dengan Persamaan berikut ini:

$$accuracy = \left(\frac{\text{Total Data Benar}}{\text{Total Data Salah}}\right) x\ 100 \qquad \ldots(2)$$

2. *Precision*

   Precision adalah metrik evaluasi yang berfungsi untuk mengukur seberapa baik model dalam membuat prediksi yang benar untuk kelas positif

dari total prediksi positif yang dilakukan. Lengkapnya, *precision* merupakan rasio prediksi benar positif yang dibandingkan dengan keseluruhan hasil yang diprediksi positif oleh model sehingga dapat diketahui seberapa akurat prediksi *positif* dari model. *Precision* sering digunakan ketika kesalahan *False Positive* (*FP*) perlu diminimalkan. Nilai *precision* dapat diperoleh dengan Persamaan berikut ini:

$$precision = \left(\frac{TP}{TP+FP}\right) x\ 100 \qquad \ldots(3)$$

Ket: (TP) *True Positive*

(FP) *False Positive*

3. *Recall*

Recall adalah *metrik* evaluasi yang berfungsi untuk mengukur seberapa baik model mampu menemukan semua *instance positif* yang sebenarnya. *Recall* mengukur *sensitivitas* model, semakin kecil jumlah *False Negative* (*FN*) berarti model mampu menemukan hampir semua data *positif* yang ada. *Recall* sangat berguna dalam situasi di mana *False Negative* (*FN*) perlu diminimalkan. Nilai *recall* dapat diperoleh dengan Persamaan berikut ini:

$$recall = \left(\frac{TP}{TP+FN}\right) x\ 100 \qquad \ldots(4)$$

Ket: (TP) *True Positive*

(FN) *False Negative*

4. *F1-score*

F1-score adalah *metrik* yang digunakan untuk menyeimbangkan precision dan recall dalam satu nilai, sehingga memberikan gambaran seberapa baik model dalam memprediksi data *positif* secara akurat dan juga menangkap semua data *positif* yang sebenarnya. Nilai *f1-score* dapat diperoleh dengan Persamaan berikut ini[30]:

$$f1{-}score = 2\ (\frac{precision\ x\ recall}{precision\ x\ recall})\ x\ 100 \qquad \ldots(5)$$

Dalam pengembangan *chatbot*, *BERT* banyak digunakan untuk meningkatkan akurasi *intent classification* dan pemahaman bahasa alami. kemampuan sistem komputer untuk memahami, menafsirkan, dan menghasilkan bahasa manusia secara alami. *NLP* berperan penting dalam menciptakan interaksi yang empatik dan kontekstual[31].

Selain itu, *BERT Embedding*, yaitu representasi teks berbasis *transformer* yang mampu menangkap konteks semantik dalam suatu kalimat, setiap teks diolah melalui model *BERT* untuk menghasilkan vektor *embedding* berdimensi tetap yang mewakili keseluruhan kalimat[32]. Hal ini membuat *BERT* menjadi salah satu model yang paling banyak digunakan dalam pengembangan sistem *NLP* modern.

2.8 Perbandingan Model *Transformer* dan Alasan Pemilihan *BERT*

Dalam perkembangan *Natural Language Processing* (*NLP*), model berbasis *transformer* telah menjadi pendekatan utama dalam berbagai tugas pemrosesan bahasa, termasuk klasifikasi teks.

Perbandingan karakteristik model dapat dilihat pada Tabel berikut.

Tabel 2. 1 Perbandingan Model *Transformer BERT*

Model	**Karakteristik**	**Kelebihan**	**Kekurangan**
BERT	*Transformer encoder* (*bidirectional*)	Memahami konteks dua arah, performa stabil	Membutuhkan *resource* besar
IndoBERT	*BERT* khusus bahasa Indonesia	Lebih akurat untuk teks Indonesia	Bergantung pada data pretraining
RoBERTa	Optimasi *BERT* (*training* lebih besar)	Performa lebih tinggi dari *BERT*	Membutuhkan data dan komputasi besar
ALBERT	Versi ringan *BERT*	Lebih efisien dan cepat	Potensi penurunan akurasi

Alasan Peneliti memilih Model *BERT*. Berdasarkan perbandingan tersebut, model *BERT* dipilih sebagai metode utama dalam penelitian ini karena beberapa pertimbangan.

Pertama, *BERT* memiliki kemampuan dalam memahami konteks kata secara dua arah, sehingga mampu menangkap makna kalimat dengan lebih baik dibandingkan model lainnya[33]. Kedua, dibandingkan dengan *RoBERTa* yang membutuhkan data pelatihan yang sangat besar, *BERT* lebih stabil dan efektif digunakan pada dataset dengan ukuran terbatas seperti dalam penelitian ini[34]. Ketiga, meskipun *IndoBERT* lebih spesifik untuk bahasa Indonesia, model *BERT* multilingual dinilai lebih fleksibel dalam menangani variasi bahasa dan istilah yang terdapat dalam dataset penelitian[35]. Keempat, dibandingkan dengan *ALBERT* yang berfokus pada efisiensi, *BERT* memberikan keseimbangan yang lebih baik antara performa dan kompleksitas model[36].

Selain itu, berdasarkan hasil pengujian yang telah dilakukan, model *BERT* mampu mencapai akurasi sebesar 93,33%, yang menunjukkan bahwa model ini memiliki performa yang sangat baik dalam tugas klasifikasi teks. Dengan demikian, *BERT* dipilih sebagai model yang paling sesuai karena mampu memberikan keseimbangan antara kemampuan memahami konteks dan performa klasifikasi yang optimal.

2.9 Kelapa Sawit

Kelapa sawit merupakan salah satu komoditas potensial sebagai sumber penghasil bioenergi. *CPO* merupakan bahan baku utama untuk menghasilkan bahan bakar nabati berupa biodiesel. Sedangkan limbah perkebunan kelapa sawit dalam bentuk tandan kosong, serat, cangkang serta limbah cair (*POME*), berpotensi untuk dimanfaatkan sebagai sumber energi listrik[37]. Pengelolaan perkebunan kelapa sawit mencakup berbagai aspek penting seperti pemeliharaan tanaman, pemupukan, pengendalian hama dan penyakit, serta proses panen dan distribusi. Kompleksitas dalam pengelolaan ini menuntut adanya sistem informasi yang dapat membantu dalam pengambilan keputusan secara cepat dan akurat.

Penelitian terbaru menunjukkan bahwa digitalisasi dalam sektor pertanian, termasuk perkebunan kelapa sawit, dapat meningkatkan efisiensi operasional dan produktivitas secara signifikan. Teknologi berbasis kecerdasan buatan juga mulai diterapkan untuk membantu proses monitoring kondisi tanaman dan prediksi hasil panen. Transformasi struktural dari sektor pertanian ke sektor industri juga telah meningkatkan pendapatan per kapita dan mengantarkan masyarakat Indonesia dari

agraris menuju ekonomi yang mengandalkan proses peningkatan nilai tambah berbasis industri yang diakselerasi oleh perkembangan teknologi digital[38].

Selain itu, penggunaan sistem berbasis data dalam perkebunan kelapa sawit terbukti mampu membantu pengelola dalam menentukan strategi terbaik untuk meningkatkan hasil produksi serta mengurangi risiko kerugian[39]. Transformasi digital ini menjadi salah satu langkah penting dalam *modernisasi* sektor Perkebunan.

2.10 Pengelolaan Kelapa Sawit

Pengelolaan kelapa sawit merupakan proses pengaturan dan pengendalian kegiatan perkebunan yang mencakup aspek kebijakan, kelembagaan, serta keberlanjutan lingkungan dalam mendukung transformasi ekonomi pertanian[40].

Pengelolaan dalam perkebunan kelapa sawit berkaitan dengan pengaturan kegiatan panen sebagai bagian dari sistem produksi yang menghubungkan kebun dan pabrik[41]. Pengelolaan dilakukan untuk meningkatkan efektivitas pengendalian dan produktivitas tanaman kelapa sawit. Pengelolaan kelapa sawit berkelanjutan menekankan keseimbangan antara aspek ekonomi, sosial, dan lingkungan dalam industri Perkebunan[42].

2.11 *Google Colab*

Google colab merupakan layanan *cloud* gratis yang disediakan oleh *Google* dalam bentuk *executable document* yang dapat digunakan untuk menulis, mengedit, menyimpan serta membagikan program yang telah ditulis melalui *Google Drive*.

Proses training data set dilakukan dengan bantuan *Google Colab*, karena *google colab* menyediakan *GPU* dengan kapasitas 12 GB dengan *support N-Vidia*[43].

Dengan menggunakan *Google Colab,* proses *preprocessing* data, pelatihan model, dan evaluasi performa dapat dilakukan secara efisien tanpa memerlukan perangkat keras khusus, sekaligus mendukung *reproduktifitas* penelitian karena kode dan hasil analisis dapat dengan mudah dibagikan kepada pihak lain untuk diverifikasi atau dikembangkan lebih lanjut[44].

Google Colab adalah sebuah *IDE* (*Integrated Development Environment*) untuk pemrograman *Python* dimana pemrosesan akan dilakukan oleh server *Google* yang memiliki perangkat keras dengan performa yang tinggi. Dari sisi perangkat lunak, *Google Colab* telah menyediakan hampir sebagian besar pustaka (*library*) yang dibutuhkan[45].

2.12 *Unified Modelling Language* (*UML*)

UML merupakan suatu teknik untuk memodelkan sistem. Pengertian lainnya, *UML* adalah seperangkat aturan dan notasi untuk spesifikasi sistem *software*. Notasi ini menyediakan satu set elemen grafis untuk pemodelan sistem. Perancangan dan pembangunan aplikasi atau *software* berbasis objek atau *Object Oriented Analysis and Design* (*OOAD*) menganggap segala sesuatunya adalah objek serta sistem dipandang sebagai interaksi dari banyak objek yang dimodelkan menggunakan *UML*[46].

Dalam melakukan perancangan untuk membuat aplikasi harus mempertimbangkan tujuan dan manfaat. Perancangan ini dibuat berdasarkan

konsep *UML* menjadi beberapa bagian yaitu, *use case diagram, activity diagram, class diagram, dan sequence diagram*[47].

1. *Use Case Diagram*

   Use case diagram merupakan *tools* untuk mengembangkan sistem dengan menjelaskan hubungan antara *actor* yang terlibat dalam sistem. Diagram ini biasanya digunakan untuk mengkomunikasikan fungsi *high-level* dari sistem dan ruang lingkup sistem yang menggambarkan beberapa *ekspektasi* (harapan) dari sistem oleh praktisi di luar sistem. *Diagram* ini berguna untuk orang di luar sistem yang biasanya digunakan untuk menentukan perilaku sistem dari sudut pandang pengguna.

   Sebuah *use case* menjelaskan sebuah fungsi yang disediakan oleh sistem yang menghasilkan *output* yang terlihat untuk seorang *actor*. Sedangkan seorang *actor* menjelaskan setiap *entitas* yang berinteraksi dengan sistem. Identifikasi *actor* dan *use case* menghasilkan definisi batas sistem (*system boundary*), yaitu membedakan tugas/kegiatan yang diselesaikan oleh sistem dan tugas/kegiatan yang diselesaikan oleh lingkungannya. *Actor* berada di luar *system boundary*, sedangkan *use case* berada di dalam *system boundary.*

2. *Activity Diagram*

   Activity diagram merupakan salah satu yang paling banyak digunakan di antara semua diagram *UML*. Alur *activity diagram* dapat digunakan untuk mengontrol urutan tindakan yang dieksekusi atau untuk merepresentasikan komunikasi data. Biasanya diagram ini digunakan untuk

desain tingkat tinggi seperti memodelkan proses bisnis, dan desain tingkat rendah seperti untuk mendeskripsikan algoritma yang akan diimplementasikan.

Activity diagram menunjukkan adanya alur kerja atau aktivitas yang berjalan dari suatu sistem. *Diagram* ini berfokus pada aliran kontrol (*control flow*) dari satu aktivitas ke aktivitas lainnya, berupa sebuah proses yang didorong oleh *internal processing*. *Diagram* aktivitas bersifat *hierarkis*, maksudnya adalah suatu aktivitas yang dibuat dari tindakan atau grafik sub-aktivitas dan aliran objek yang terkait.

3. *Sequence Diagram*

Sequence diagram adalah *diagram UML* paling umum kedua yang merepresentasikan bagaimana objek berinteraksi dan bertukar pesan dari waktu ke waktu (*over time*). *Sequence diagram* menggambarkan bagaimana peristiwa (*aktivitas*) dalam *use case* yang dipetakan ke dalam operasi kelas objek dalam *class diagram*. Penerimaan umum *sequence diagram* dapat dikaitkan dengan sifatnya yang relatif *intuitif* dan kemampuan untuk menggambarkan perilaku *parsial*.

Pokok dari *sequence diagram* yaitu menunjukkan interaksi antar objek dan indikasi hubungan antar objek. *Sequence diagram* mewakili interaksi antara bagian-bagian dari sistem. Tujuan utama dari *sequence diagram* adalah untuk mengungkapkan bagaimana objek berinteraksi satu sama lain untuk mengimplementasikan suatu fungsi yang dapat mewakili

object collaboration model yang ada dalam pikiran perancang sistem untuk program masa depan saat *runtime*.

4. *Class Diagram*

Class diagram adalah *class* yang menggambarkan struktur dan penjelasan dari *class*, *package*, dan *object* serta hubungan satu sama lain seperti *containment*, *inheritance*, *associations*, dan lain-lain. *Class* adalah *abstraksi* yang merepresentasikan struktur dan perilaku umum dari sekumpulan *object*. *Object* adalah turunan dari *class* yang dibuat (*created*), dimodifikasi (*modified*), dan dihancurkan (*destroyed*) selama eksekusi sistem. Sebuah *object* memiliki status yang menyertakan nilai *attribute*-nya dan hubungannya dengan *object* lain. *Class diagram* berfungsi sebagai cara untuk merencanakan dan berkomunikasi antar *developers*.

Class diagram merepresentasikan struktur statis dari sistem yang mewakili *object class* dalam sistem dan *relationship* pada setiap *class*. Selain itu, *class diagram* digunakan untuk memodelkan tampilan struktur sistem yang mencakup kemampuan untuk membawa data dan menjalankan tindakan (*execute actions*), juga sebagai dasar untuk pengembangan sistem perangkat lunak lebih lanjut, misalnya *design phase*[48].

2.13 *Visual Studio Code*

Visual Studio Code merupakan editor kode sumber gratis dan *opensource* yang dikembangkan oleh *Microsoft*. Ini tersedia untuk *Windows*, *macOS*, *Linux*, dan bahkan dapat dijalankan di *web browser*[49]. *Visual Studio Code* berfungsi sebagai editor pemrograman[50].

Beberapa fitur utama dari *Visual Studio Code* antara lain:

1. *IntelliSense*: Fitur penyelesaian kode yang cerdas, memberikan saran sintaks, parameter, dan referensi berdasarkan konteks kode yang sedang ditulis.
2. *Debugging*: Memungkinkan pengembang untuk menjalankan dan menganalisis kode secara langsung tanpa perlu menggunakan alat *debugging eksternal*.
3. *Git Integration*: Mendukung sistem kendali versi *Git* secara langsung di dalam editor, memungkinkan pengguna untuk melakukan *commit*, *push*, *pull*, dan merge dengan mudah.
4. *Extensions Marketplace*: Memungkinkan pengembang menambahkan berbagai ekstensi seperti *Flutter*, *Dart*, dan lainnya untuk meningkatkan fungsionalitas editor.
5. *Cross-Platform*: Dapat dijalankan di berbagai sistem operasi seperti *Windows*, *macOS*, dan *Linux*[51].

2.14 *Python*

Python adalah salah satu bahasa pemrograman yang paling populer karena sintaksnya yang sederhana namun kuat dan dukungannya yang kuat terhadap paradigma Pemrograman Berorientasi Objek. Selain itu, *Python* memiliki banyak *library* dan *framework* yang dapat digunakan untuk membuat aplikasi desktop secara cepat dan efisien, seperti *Tkinter*, *PyQt*, dan *Kivy*[52].

Chatbot menggunakan *Python* untuk mendukung manajemen hubungan pelanggan. *Chatbot* ini dirancang untuk meningkatkan interaksi dan pelayanan dengan memberikan respons cepat dan akurat terhadap pertanyaan pelanggan[53].

Sistem dibangun melalui tahapan pengumpulan data *kuesioner*, pembentukan fungsi keanggotaan, penentuan aturan *fuzzy*, hingga proses *defuzzifikasi*. penelitian ini bertujuan membangun suatu sistem yang mampu menganalisis kompleksitas data dan memberikan rekomendasi keputusan secara efisien terkait tingkat stres mahasiswa[54].

2.15 *Database*

Database atau basis data merupakan database elektronik, kumpulan data, dan atau informasi yang disimpan didalam komputer secara sistematik dan telah diatur secara khusus untuk pencarian dan pengambilan cepat oleh komputer. *Database* ini dapat memfasilitasi penyimpanan, pengambilan, modifikasi, dan penghapusan data dalam hubungannya dengan berbagai operasi pemrosesan data.

Basis data atau *database* mempunyai manfaat yang sangat menguntungkan khususnya untuk teknologi digital, yang kini telah mengubah hampir setiap aspek kehidupan modern[55].

Basis data merupakan kumpulan dari data yang memiliki hubungan antara satu dengan yang lainnya, tersimpan pada perangkat keras komputer dan dapat digunakan perangkat lunak untuk memanipulasinya. Dari definisi ini, terdapat tiga hal yang berhubungan dengan basis data, yaitu sebagai berikut:

1. Data yang terdapat dalam komputer itu sendiri yang diorganisasikan dalam bentuk basis data;
2. Simpanan permanen *(storage)* digunakan untuk menyimpan basis data tersebut. Simpanan ini merupakan salah satu bagian dari teknologi perangkat keras yang digunakan pada sistem informasi. Simpanan permanen pada umumnya berupa sebuah *hard disk*;
3. Perangkat lunak untuk memanipulasi data. Perangkat lunak ini dapat dibuat Sendiri dengan menggunakan bahasa pemrograman komputer atau dibeli dalam bentuk suatu paket. Banyak paket perangkat lunak yang disediakan untuk memanipulasi basis data. Paket perangkat lunak ini disebut dengan *database* manajemen sistem[56].

2.16 *XAMPP*

XAMPP digunakan sebagai *standalone* s*erver* (berdiri sendiri) atau biasa disebut dengan *localhost*. Hal tersebut memudahkan dalam proses pengeditan, desain, dan pengembangan aplikasi[57].

Xampp singkatan dari *X Apache MySQL PHP Perl*,X adalah sistem operasi (*Windows, Linux, Unix*), merupakan paket *software* yang terdiri dari *server web* (*Apache*), *database* (*MySQL – MariaDB*), dan pengembangan aplikasi (*PHP* dan *Perl*); disebut juga sebagai *software stack*, *XAMPP* dikembangkan oleh grub pengguna server *web Apache – ApacheFriends.org*[58].

2.17 *Hypertext Markup Language* (*HTML*)

HTML merupakan bahasa standar yang digunakan dokumen yang ada dalam *website*, Bahasa pemograman *HTML* menggunakan *tag* (akhiran) yang menandakan cara suatu *keyword*, kebanyakan *browser* mengenali akhiran *HTML*, biasanya *tag* berpasangan dan setiap *tag* ditandai dengan symbol[59].

Hypertext Markup Language (*HTML*) adalah bahasa pemrograman yang mengatur cara menampilkan dan mengelola informasi di internet. *HTML* pertama kali dibuat oleh Tim *Berners-Lee* saat bekerja di *CERN* dan mulai dikenal luas melalui *browser Mosaic*. Pada awal tahun 1990-an, *HTML* mengalami kemajuan pesat, dan setiap versi baru dari *HTML* selalu membawa peningkatan kemampuan dan fitur dibandingkan versi sebelumnya[60].

2.18 *Website*

Website merupakan kumpulan halaman pada suatu domain di internet yang dibuat dengan tujuan tertentu dan saling berhubungan serta dapat diakses secara luas melalui halaman depan menggunakan sebuah *URL website*[61].

Web adalah sistem berkaitan dengan file yang digunakan sebagai media untuk menampilkan, *text*, *image*, *multimedia* dan lainnya di jaringan *internet*, baik yang bersifat statis atau dinamis yang membentuk *building chain* yang saling terkait masing-masing terhubung dengan jaringan halaman (*hyperlink*)[62].

2.19 pengujian *black-box*

Pengujian *blackbox* adalah salah satu aktivitas dalam proses pengembangan sistem yang dirancang secara terencana dan sistematis untuk memastikan atau

memancarkan keakuratan hasil yang diinginkan dari perangkat lunak yang dikembangkan. Aktivitas ini mencakup pembuatan serta penerapan sejumlah kasus uji *(test case)* tertentu guna menjamin kualitas perangkat lunak sesuai dengan kepuasan pengguna. Proses pengujian perangkat lunak memiliki peran penting dalam memancarkan kualitas perangkat lunak[63].

Pengujian *Blackbox* adalah metode pengujian yang focus kepada kebutuhan fungsional dari aplikasi, seorang penguji dapat mendefiniskan *Test Case* dan melakukan evaluasi pada kebutuhan fungsional aplikasi[64]. Pengujian *BlackBox* bertujuan untuk memverifikasi bahwa setiap proses beroperasi sesuai dengan kebutuhan yang diinginkan. Penguji dapat menginterpretasikan kondisi masukan sebagai suatu himpunan dan melaksanakan pengujian khusus terhadap fungsi sistem. Oleh karena itu, pengujian menjadi metode pelaksanaan program yang bertujuan untuk mengidentifikasi kesalahan atau *error*, dan kemudian melakukan perbaikan sehingga sistem dapat dianggap sebagai layak digunakan[65].

2.20 Penelitian Terkait

Berikut ini merupakan penelitian terkait dengan Skripsi tersebut:

Tabel 2. 2 Penelitian Terkait

No	Nama	Judul	Metode	Hasil Peneliti
1	Pratama et al. (2023)	Pengembangan *Chatbot* menggunakan *BERT*untuk pengelolaan Komunikasi Beasiswa	*Fine-tuning indoBERT* (*MLM, NSP, intent classification*)	Mencapai *F1-score* 86,74% menunjukkan pemahaman konteks yang baik

2	Sari at al. (2023)	*Implementasi Chatbot NLP* Menggunakan Model *BERT* untuk Pelayanan Klinik Gigi	*Agile development+training* model *BERT*	Akurasi 0,86; meningkatkan efisiensi layanan konsultasi
3	Wijaya et al. (20224)	*Implementasi Chatbot Watsapp* Berbasis *NLP* Dengan *Hybride BERT-GPT*	*Hybrid* Model (*BERT-GPT*)	Respon *adaptif* terhadap pertanyaan kompleks dan tidak terstuktur
4	Putri et al, (2022)	Optimasi *Chatbot* dengan pemanfaatan *Natural Language processing*	*NLP* (*sentiment analysis, intent detection*)	Peningkatan Kualitas respons *chatbot* secara signifikan
5	Rahman et al, (2023)	*Chatbot* Model *Development Using BERT for* Halal *Tourism Information*	*Fine-tuning BERT*	Informasi lebih relevan dan berbasis konteks pengguna

BAB 3

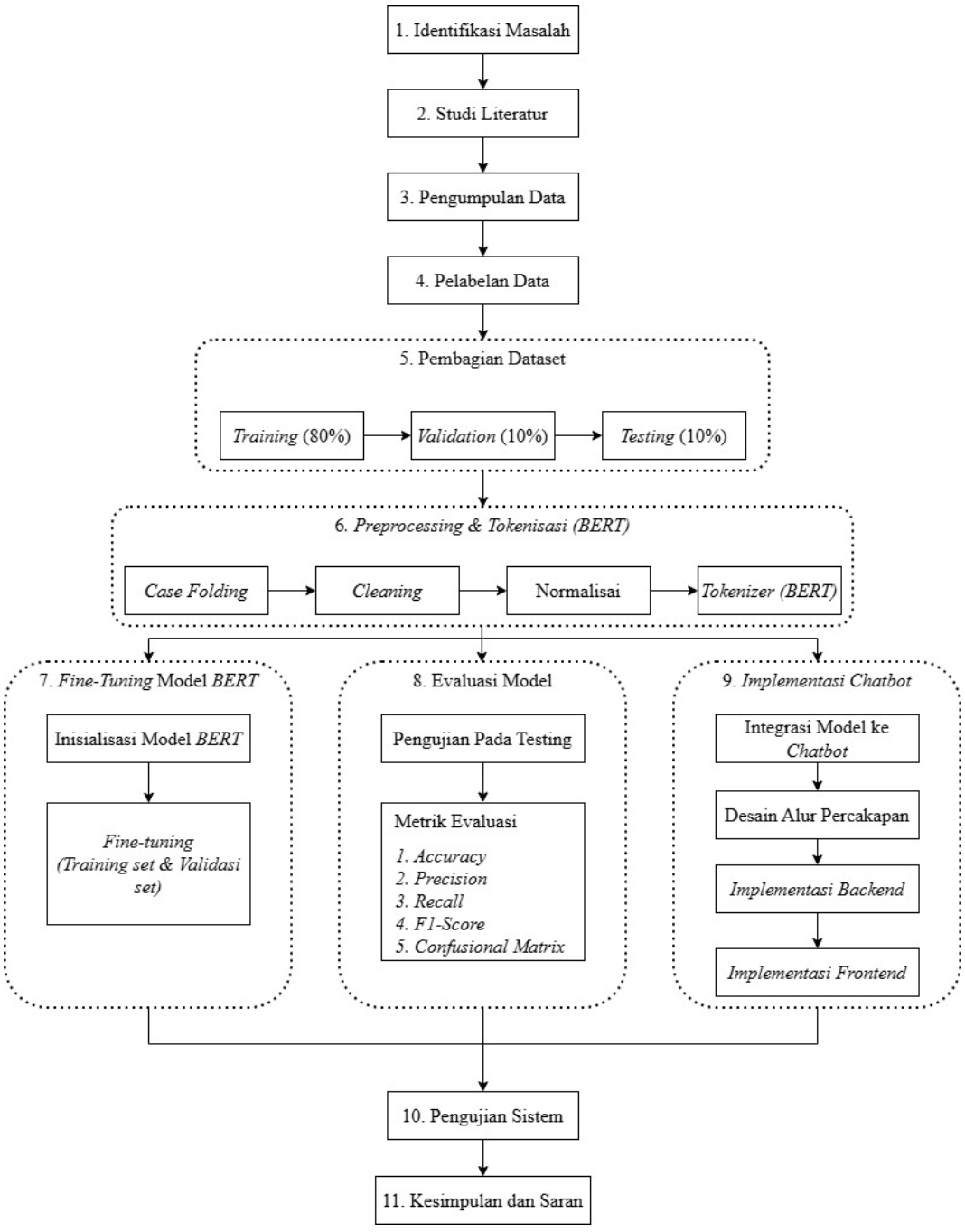
METODOLOGI PENELITIAN

3.1 Tahapan Penelitian

Tahapan penelitian merupakan rangkaian proses yang disusun secara sistematis untuk mencapai tujuan penelitian yang telah ditetapkan. Dalam penelitian ini, tahapan penelitian dirancang untuk mengembangkan *chatbot* berbasis *Natural Language Processing* (*NLP*) dengan pemodelan *BERT* dalam pengelolaan perkebunan kelapa sawit.

Alur penelitian dimulai dari identifikasi permasalahan yang terjadi di lapangan, dilanjutkan dengan pengumpulan dan pengolahan data, pengembangan model, hingga implementasi sistem *chatbot*. Setiap tahapan dilakukan secara berurutan dan saling berkaitan, sehingga hasil yang diperoleh dapat menggambarkan kondisi sebenarnya serta menghasilkan sistem yang optimal.

Tahapan penelitian ini tidak hanya berfokus pada pengembangan model, tetapi juga mencakup proses evaluasi dan pengujian untuk memastikan bahwa sistem yang dihasilkan memiliki tingkat akurasi dan kinerja yang baik. Dengan adanya tahapan yang terstruktur, penelitian ini diharapkan mampu menghasilkan solusi yang efektif dalam membantu pengelolaan informasi pada perkebunan kelapa sawit. Adapun Langkah-langkah dari penelitian ini dapat dilihat pada Gambar 3.1.



Gambar 3. 1 Alur Metodologi Penelitian

3.1.1 *Identifikasi* Masalah

Identifikasi masalah merupakan tahap awal yang bertujuan untuk memahami permasalahan dalam pengelolaan informasi pada perkebunan kelapa sawit. Pada kondisi saat ini, penyampaian informasi terkait perawatan tanaman, pemupukan, pengendalian hama, serta jadwal panen masih dilakukan secara manual dan belum terintegrasi dalam suatu sistem berbasis teknologi.

Kondisi tersebut menimbulkan beberapa kendala, antara lain keterlambatan penyampaian informasi, potensi kesalahan komunikasi, serta rendahnya efisiensi dalam proses pengambilan keputusan. Selain itu, keterbatasan tenaga ahli di lapangan menjadi hambatan dalam menyediakan informasi yang cepat dan akurat bagi pekerja maupun pengelola perkebunan.

Permasalahan lainnya adalah kesulitan dalam memahami data teks yang tidak terstruktur, seperti catatan lapangan dan pertanyaan pengguna yang bervariasi. Oleh karena itu, diperlukan suatu sistem yang mampu mengolah dan memahami bahasa alami secara otomatis.

Berdasarkan permasalahan tersebut, penelitian ini mengusulkan pengembangan *chatbot* berbasis *Natural Language Processing* (*NLP*) dengan memanfaatkan model *BERT* untuk meningkatkan efektivitas penyampaian informasi dalam pengelolaan perkebunan kelapa sawit.

3.1.2 Studi *Literatur*

Studi *literatur* dilakukan untuk memperoleh landasan *teoritis* yang relevan serta memahami perkembangan teknologi yang berkaitan dengan penelitian ini.

Sumber *literatur* meliputi jurnal ilmiah, buku, serta publikasi terkait *Natural Language Processing* (*NLP*), *chatbot*, dan model *BERT* (*Bidirectional Encoder Representations from Transformers*). Melalui studi ini, dipahami bahwa *NLP* merupakan bidang yang memungkinkan komputer untuk memahami, menginterpretasikan, dan menghasilkan bahasa manusia.

Model *BERT* dipelajari karena kemampuannya dalam memahami konteks kata secara dua arah (*bidirectional*), sehingga mampu meningkatkan performa pada tugas klasifikasi teks.

Selain itu, penelitian terdahulu yang berkaitan dengan *implementasi chatbot* dan analisis teks juga dianalisis untuk mengidentifikasi kelebihan dan keterbatasan metode yang telah digunakan. Hasil studi *literatur* ini menjadi dasar dalam menentukan pendekatan yang digunakan dalam penelitian.

3.1.3 Pengumpulan Data

Pengumpulan data merupakan tahap penting dalam penelitian karena data yang diperoleh akan menjadi dasar dalam pengembangan model. Data yang digunakan dalam penelitian ini diperoleh dari wawancara dan berbagai sumber, antara lain literatur pertanian, dokumen terkait perkebunan, serta referensi lain yang relevan. Dataset yang digunakan berjumlah 1200 data.

Tabel 3. 1 Sumber Data

NO	Sumber Data	Banyaknya Data
1	BPTP Riau	11
2	Augmentasi	395
3	Cybext Kementan	192
4	Ditjenbun Kementan	112
5	PPKS Medan	107
6	Sinar Tani	100

7	BPDPKS	70
8	Jurnal Penelitian Kelapa Sawit	68
9	Agro Indonesia	50
10	Majalah Kelapa Sawit Indonesia	40
11	Trobos Agri	40
12	Buku Fauzi et al.	15
	Total	1.200

Dataset terdiri dari beberapa *atribut*, yaitu teks sebagai *input* berupa pertanyaan pengguna, *intent* sebagai *label klasifikasi*, sumber sebagai asal data, serta *kb*_topik sebagai topik *spesifik*.

Data yang telah dikumpulkan kemudian dikelompokkan ke dalam empat kategori utama, yaitu pemupukan, pengendalian hama, panen, dan informasi umum. Distribusi jumlah data pada masing-masing kategori dapat dilihat pada Tabel berikut.

Tabel 3. 2 *Distribusi intent*

No	Kategori	Jumlah Data
1	Pemupukan	326
2	Pengendalian Hama	326
3	Panen	274
4	Informasi Umum	274
	Total	1.200

Berdasarkan tabel tersebut, dapat diketahui bahwa *distribusi* data pada setiap kategori *relatif* seimbang sehingga dapat mendukung proses pelatihan model secara optimal.

3.1.4 Pelabelan Data

Dataset keseluruhan berjumlah 1.200 data. Setelah pembagian secara stratified, data pengujian berjumlah 120 data yang terdiri atas 33 data pemupukan, 33 data pengendalian hama, 27 data panen, dan 27 data informasi umum. Pelabelan

data dilakukan untuk mengelompokkan data ke dalam kategori tertentu yang akan digunakan dalam proses pelatihan model.

Dalam penelitian ini, data diklasifikasikan ke dalam beberapa kategori, seperti:

1. Pemupukan, 33 data.
2. Pengendalian hama, 33 data.
3. Panen, 27 data.
4. Informasi umum, 27 data.

Ketepatan dalam pelabelan sangat penting karena akan mempengaruhi kemampuan model dalam memahami pola data. Oleh karena itu, setiap data diperiksa secara teliti agar label yang diberikan sesuai dengan makna yang terkandung dalam teks.

Hasil dari pelabelan ini akan digunakan sebagai dasar dalam proses pembelajaran model *BERT* sehingga model dapat mengenali berbagai jenis pertanyaan dan memberikan respon yang sesuai.

3.1.5 *Pre-Processing* Data

Pre-processing data merupakan tahap pengolahan data awal untuk membersihkan dan menyiapkan data sebelum digunakan dalam pelatihan model.

Data teks yang diperoleh biasanya masih mengandung berbagai elemen yang tidak diperlukan, seperti simbol, tanda baca, atau kata yang tidak memiliki makna penting. Oleh karena itu, dilakukan beberapa tahapan *preprocessing*, antara lain:

Tahapan yang dilakukan meliputi:

1. *Case folding* (mengubah huruf menjadi *lowercase*)
2. *Cleaning* (menghapus simbol dan karakter tidak relevan)
3. *Tokenizing* (memecah teks menjadi token/kata)
4. Normalisasi kata (mengubah kata tidak baku menjadi baku)
5. *Encoding* (mengubah teks menjadi representasi numerik)

Namun, dalam penggunaan model *BERT*, proses *preprocessing* dilakukan secara lebih minimal karena *BERT* telah memiliki *tokenizer* bawaan (*WordPiece tokenizer*) yang mampu menangani struktur teks secara efektif. Pada penelitian ini, proses *tokenisasi* dan *encoding* dilakukan menggunakan *tokenizer* bawaan *BERT* sehingga *preprocessing* dilakukan secara minimal.

3.1.6 *Splitting* Data

Setelah data melalui tahap *preprocessing*, langkah selanjutnya adalah membagi *dataset* menjadi beberapa bagian, yaitu data:

1. Data *training*: 960 Sampel (80%).
2. Data *validation*: 120 Sampel (10%).
3. Data *testing*: 120 Sampel (10%).

Data *training* digunakan untuk melatih model, data *validation* digunakan untuk mengevaluasi performa selama proses pelatihan, sedangkan data *testing* digunakan untuk mengukur performa akhir model terhadap data yang belum pernah dilihat sebelumnya.

3.1.7 Model *Training*

Pada tahap ini, model *BERT* dilatih menggunakan data *training* untuk memahami pola bahasa dan konteks dari teks yang digunakan. Proses pelatihan dilakukan dengan teknik *fine-tuning*, di mana model yang telah dilatih sebelumnya disesuaikan dengan *dataset* penelitian. Model akan mempelajari hubungan antar kata serta makna dari setiap kalimat sehingga mampu memahami maksud dari pertanyaan pengguna.

Adapun konfigurasi parameter yang digunakan adalah sebagai berikut:

1. Model: *BERT Base Multilingual*
2. *Maximum sequence length*: 128
3. *Batch size*: 16
4. *Learning rate*: 2e-5
5. *Epoch*: 5

Proses pelatihan dilakukan menggunakan *Google Colab* dengan dukungan *GPU* untuk mempercepat proses *training*. Pemilihan parameter ini didasarkan pada penelitian sebelumnya yang menunjukkan bahwa *konfigurasi* tersebut mampu memberikan performa optimal dalam tugas klasifikasi teks.

Selama proses pelatihan, parameter model akan diperbarui secara bertahap untuk meningkatkan akurasi prediksi. Hasil dari tahap ini adalah model yang mampu mengenali pola bahasa dan memberikan respon yang sesuai.

3.1.8 Evaluasi Model

Evaluasi model dilakukan untuk mengukur kinerja model dalam mengklasifikasikan data berdasarkan intent pengguna. Tahap ini bertujuan untuk mengetahui sejauh mana model mampu memahami konteks pertanyaan dan memberikan prediksi yang sesuai. Evaluasi dilakukan menggunakan beberapa metrik, yaitu *accuracy, precision, recall, dan F1-score.* Selain itu, digunakan juga *confusion matrix* untuk melihat distribusi hasil prediksi model terhadap masing-masing kelas.

Confusion matrix digunakan untuk mengevaluasi performa model dengan membandingkan antara nilai aktual dan hasil prediksi. *Matriks* ini terdiri dari:

1. *True Positive* (TP)
2. *True Negative* (TN)
3. *False Positive* (FP)
4. *False Negative* (FN)

Melalui *confusion matrix*, dapat diketahui jenis kesalahan yang dilakukan oleh model dalam proses klasifikasi. Nilai dari masing-masing metrik digunakan untuk menilai performa model secara menyeluruh. Semakin tinggi nilai metrik, maka semakin baik kemampuan model dalam melakukan klasifikasi. Apabila hasil evaluasi belum memenuhi kriteria yang diharapkan, maka dilakukan perbaikan melalui proses pelatihan ulang (*retraining*) atau penyesuaian parameter model. Sebaliknya, jika hasil evaluasi telah memenuhi kriteria, maka model dapat digunakan pada tahap *implementasi chatbot*.

3.1.9 *Implementasi Chatbot*

Setelah model memiliki performa yang baik, model di *implementasikan* ke dalam sistem *chatbot*. *Chatbot* dirancang agar dapat menerima *input* dari pengguna, memproses teks menggunakan model *BERT*, dan memberikan respon yang sesuai. Sistem ini dapat diintegrasikan ke dalam aplikasi berbasis *web* atau *mobile*. *Implementasi* ini bertujuan untuk memanfaatkan model yang telah dikembangkan dalam bentuk sistem yang dapat digunakan secara langsung.

3.1.10 Pengujian Sistem

Pengujian sistem dilakukan untuk memastikan *chatbot* dapat berjalan dengan baik dan memberikan respon yang akurat. Pengujian yang di gunakan *Blackbox*. Hasil pengujian digunakan untuk mengetahui apakah sistem telah memenuhi kebutuhan pengguna.

3.1.11 Kesimpulan dan Saran

Tahap terakhir adalah penarikan kesimpulan berdasarkan hasil penelitian yang telah dilakukan. Kesimpulan berisi ringkasan hasil pengembangan *chatbot* serta tingkat keberhasilan model dalam memahami bahasa pengguna. Sementara itu, saran diberikan sebagai rekomendasi untuk pengembangan sistem selanjutnya, seperti penambahan dataset atau peningkatan fitur *chatbot*.