

Sistem Pakar Menggunakan Metode *Natural Language Processing* Untuk Mendiagnosa Penyakit Tanaman Cabai Merah Keriting

Muhammad Yusuf^{1*}, Diajeng Ganis Samantha Murpri², Khumairah Awaliyah Ernas³
 Teknik Informatika, Universitas Muhammadiyah Sorong, Sorong, Indonesia
 *E-mail *Corresponding Author*: yusuf@um-sorong.ac.id

Abstract

Curly red chili (Capsicum annum L.) is one of the main horticultural commodities in Indonesia that often experiences a decrease in productivity due to pest and disease attacks. The main problem faced by farmers is the limited knowledge in recognizing disease symptoms and determining the right treatment, which often results in crop failure. To address this challenge, this research developed an expert system based on the Natural Language Processing (NLP) method implemented in a website with a chatbot consultation feature. The system analyzes disease symptom descriptions using NLP algorithms, then provides diagnoses and treatment recommendations quickly and accurately. The development results show that this expert system is effective in providing diagnoses of curly red chili plant diseases and facilitating farmers in taking appropriate control measures. Datasets were obtained from agricultural experts, and system development was carried out using the forward chaining method.

Keywords: *Curly red chili; Chatbot; Natural Language Processing; Website*

Abstrak

Tanaman cabai merah keriting (*Capsicum annum L.*) merupakan salah satu komoditas hortikultura utama di Indonesia yang sering mengalami penurunan produktivitas akibat serangan hama dan penyakit. Permasalahan utama yang dihadapi petani adalah keterbatasan pengetahuan dalam mengenali gejala penyakit dan menentukan penanganan yang tepat, yang seringkali mengakibatkan gagal panen. Untuk menjawab tantangan tersebut, penelitian ini mengembangkan sistem pakar berbasis metode *Natural Language Processing* (NLP) yang diterapkan dalam sebuah *website* dengan fitur konsultasi *chatbot*. Sistem menganalisis deskripsi gejala penyakit menggunakan algoritma NLP, lalu memberikan diagnosa dan rekomendasi penanganan secara cepat dan akurat. Hasil pengembangan menunjukkan bahwa sistem pakar ini efektif dalam memberikan diagnosa penyakit tanaman cabai merah keriting dan memfasilitasi petani dalam mengambil tindakan pengendalian yang tepat. *Dataset* diperoleh dari pakar pertanian, dan pengembangan sistem dilakukan dengan metode *Forward Chaining*.

Kata kunci: *Cabai Merah Keriting; Chatbot; Natural Language Processing; Website*

1. Pendahuluan

Tanaman cabai merah keriting (*Capsium annum L.*) merupakan salah satu jenis sayuran yang paling banyak ditanam oleh para petani di wilayah Indonesia. Tanaman ini menjadi pusat perhatian karena dibutuhkan sebagai bumbu masak yang memiliki citra rasa pedas berasal dari senyawa alami yang disebut capsaicin[1]. Cabai ini memiliki bentuk panjang dengan permukaan yang keriting atau berlekuk-lekuk berbeda dari cabai merah biasa yang lebih lurus. Bagi para petani membudidayakan tanaman cabai merah keriting mempunyai nilai ekonomis yang cukup tinggi dan setiap tahunnya kebutuhan cabai merah keriting terus mengalami peningkatan sehingga memberikan kesempatan bagi para petani sebagai sumber mata pencaharian[2].

Beberapa faktor yang memengaruhi produksi cabai meliputi luas lahan, kualitas bibit, penggunaan pupuk, pestisida, serta adanya hama, penyakit, dan tenaga kerja[3]. Salah satu penyebab utama turunnya produksi cabai adalah serangan hama dan penyakit yang bisa terjadi mulai dari persemaian hingga masa panen. Untuk mencegah penyebaran hama dan penyakit di lahan pertanian, penting dilakukan diagnosa sejak dini. Namun, banyak petani baru menyadari

adanya serangan setelah kerusakan terjadi, dan sering kali mereka tidak tahu jenis hama atau penyakit yang menyerang serta cara penanganan yang tepat[4].

Karena dalam hal ini, pengetahuan seorang pakar perlu diintegrasikan ke dalam bidang komputer untuk menciptakan sebuah sistem yang dapat membantu dalam mengidentifikasi penyakit pada tanaman cabai. Melalui hal ini, sistem berfokus dengan memanfaatkan kemampuan NLP dalam memahami dan menginterpretasikan bahasa alami pengguna untuk mengidentifikasi penyakit secara akurat dari data yang diberikan langsung oleh pakar sehingga mudah digunakan para petani.

Chatbot merupakan sebuah program komputer yang dirancang untuk berkomunikasi dengan manusia melalui percakapan atau pesan. *Chatbot* digunakan di berbagai industri, seperti hiburan, pendidikan, dan pariwisata, untuk layanan pelanggan dan informasi[5]. Dalam diagnosa penyakit cabai merah keriting, *chatbot* menjadi solusi efektif untuk memberikan pengetahuan, mendukung deteksi dini, dan pencegahan tanpa memerlukan konsultasi langsung dengan ahli.

Penelitian ini bertujuan untuk mengembangkan sebuah sistem pakar berbasis *Natural Language Processing* (NLP) yang mampu mendiagnosa penyakit pada tanaman cabai merah keriting dengan menganalisis gejala yang diinputkan dalam bahasa alami oleh pengguna. Sistem ini dirancang untuk memberikan rekomendasi pengobatan yang tepat secara cepat dan akurat, sehingga dapat meningkatkan efektivitas dalam penanganan penyakit tanaman. Selain itu, fitur interaktif berbasis *chatbot* diintegrasikan untuk mempermudah petani dalam melakukan deteksi dini terhadap penyakit yang menyerang tanaman mereka. Dengan demikian, penelitian ini diharapkan dapat mendukung keberlanjutan sektor pertanian melalui inovasi teknologi yang mengurangi risiko gagal panen akibat keterbatasan pengetahuan mengenai penyakit tanaman.

2. Tinjauan Pustaka

Pada penelitian yang berjudul "Implementasi Sistem Pakar Diagnosa Penyakit Cabe dengan Metode *Certainty Factor*"[6]. Pada penelitian ini terdapat kendala yaitu kurangnya data dan informasi yang lengkap dari para pakar untuk menentukan nilai kepercayaan (*certainty factor*) yang akurat bagi setiap gejala dan penyakit sehingga menjadi hambatan karena kekurangan data dari pakar dapat mempengaruhi keakuratan hasil diagnosa, dan implementasi *certainty factor* dalam sistem pakar membutuhkan perhitungan yang kompleks, terutama ketika terdapat banyak gejala dan kemungkinan penyakit. Hal ini dapat mempengaruhi kecepatan dan efektivitas sistem dalam menghasilkan diagnosa.

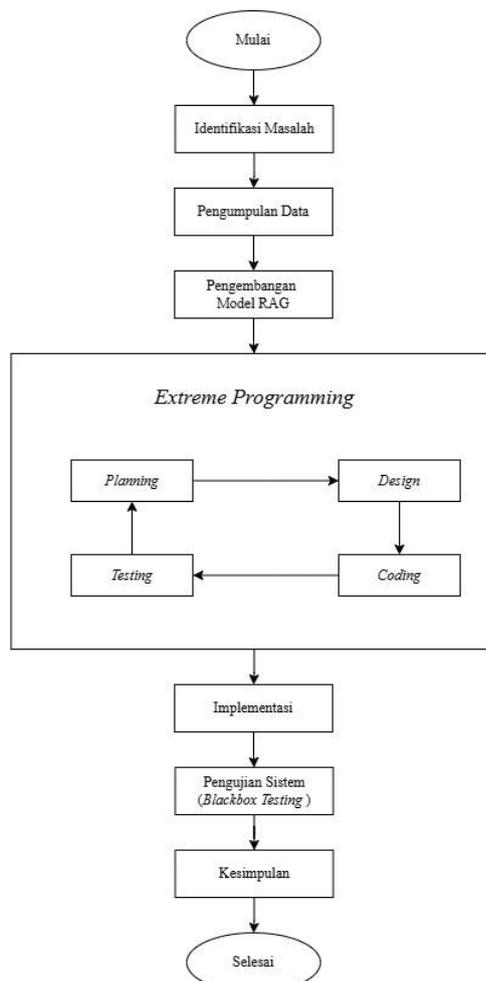
Penelitian lain yang berjudul "Sistem Pakar Identifikasi Penyakit Tanaman Cabai Rawit Menggunakan Metode *Forward Chaining* Berbasis *Web*"[7]. Sistem ini dirancang untuk membantu petani dalam mengidentifikasi penyakit pada tanaman cabai rawit secara dini menggunakan metode *forward chaining* berbasis *web*. Dengan adanya sistem pakar ini, petani dapat mengenali gejala-gejala penyakit yang muncul dan mendapatkan informasi yang diperlukan untuk penanganan yang tepat. Selain itu, aplikasi ini juga berfungsi sebagai alat bantu bagi pegawai penyuluhan dalam memberikan edukasi kepada petani, dengan harapan dapat mengurangi kerugian yang disebabkan oleh penyakit tanaman dan meningkatkan hasil pertanian secara keseluruhan.

Penelitian terakhir yang dijadikan referensi yaitu "Klasifikasi multikelas dengan presisi tinggi penyakit daun cabai melalui *EfficientNetB4* yang disesuaikan dari gambar daun cabai"[8]. Hasil penelitian menunjukkan akurasi 92% pada data uji, dengan kinerja lebih baik dibandingkan model seperti *ResNet-50*, *DenseNet121*, *MobileNet-V2*, dan *VGG-16*. Perbedaan antara penelitian ini dengan penelitian peneliti adalah penelitian peneliti mengandalkan data gejala dalam bentuk teks yang diinput oleh pengguna. Sebaliknya, penelitian *EfficientNetB4* menggunakan citra daun cabai sebagai *input* utama.

Berdasarkan tinjauan terhadap penelitian-penelitian sebelumnya, terdapat perbedaan dalam metode yang digunakan. Penelitian ini menghadirkan kebaruan dengan mengintegrasikan metode *Natural Language Processing* (NLP) dalam sistem pakar untuk mendiagnosa penyakit tanaman cabai merah keriting, yang membedakannya dari penelitian-penelitian sebelumnya. Dengan memanfaatkan NLP, sistem dapat memahami dan menginterpretasi pertanyaan serta pernyataan pengguna, memberikan *respons* yang lebih akurat dan relevan dalam diagnosis penyakit. Dengan demikian, penelitian ini tidak hanya menawarkan solusi praktis bagi petani, tetapi juga membuka peluang baru dalam pengembangan sistem pakar yang lebih canggih dan *user-friendly*.

3. Metodologi

3.1 Tahapan Penelitian



Gambar 1. Alur Penelitian

Penelitian ini dilakukan dalam beberapa tahapan yaitu identifikasi masalah yang tujuan untuk menemukan masalah yang dihadapi oleh petani cabai merah keriting dan melakukan identifikasi terhadap sumber masalah tersebut, pengumpulan data yaitu melibatkan pengumpulan informasi tentang penyakit yang mempengaruhi tanaman cabai merah keriting diinformasikan oleh ahli pakar, pengembangan model *Natural Language Processing* (NLP) menggunakan teknik *Deep Learning* untuk meningkatkan kemampuan dalam mengidentifikasi penyakit pada tanaman cabai merah keriting, setelah itu pengembang sistem dengan SDLC *Extreme Programming* (XP) untuk membangun sebuah sistem agar dapat mendiagnosa penyakit tanaman cabai merah keriting dengan menggunakan metode NLP yang digunakan oleh petani, kemudian dilanjutkan dengan implementasi hasil pengembangan sistem yaitu untuk menguji sistem yang dibangun dalam situasi nyata agar fungsinya dapat bekerja secara menyeluruh, terakhir kesimpulan dan saran yaitu untuk mengevaluasi Hasil penelitian ini memberikan rekomendasi untuk pengembangan lebih lanjut.

Gambar 1 di atas menggambarkan tahapan alur penelitian yang diterapkan dalam studi ini. Pada tahap ini, peneliti menggunakan metode *Extreme Programming* (XP) sebagai pendekatan dalam pengembangan model perangkat lunak, yang terdiri dari beberapa tahapan. Tahap pertama adalah perencanaan, yang mencakup analisis kebutuhan yang diperlukan untuk pengembangan sistem. Tahap kedua adalah desain, di mana peneliti membuat visualisasi seperti *Flowchart*, *Use Case*, *Activity Diagram* dan *User Interface* dengan tujuan agar dapat membantu menjelaskan alur penelitian yang akan dilakukan. Kemudian tahap pengkodean menggunakan model RAG untuk menghasilkan teks dalam pembuatan *chatbot*. Terakhir tahap pengujian dilakukan untuk memastikan semua fitur fungsional *software* berfungsi dengan teliti.

3.2 Pengembangan Model RAG

Tahap ini menggunakan model *Natural Language Processing* untuk menerapkan *Deep Learning* agar dapat mengidentifikasi jenis penyakit pada tanaman cabai merah keriting yang dikembangkan melalui algoritma *Natural Language Processing* dengan menggunakan teknik *embedding* untuk representasi kata atau frase dalam bentuk vektor dan perhitungan *similarity* untuk mengukur kemiripan dari teks[9].

1) Pengumpulan *Dataset*

Peneliti mengumpulkan *dataset* yang didapatkan berdasarkan hasil wawancara bersama pakar dan dituangkan ke dalam bentuk teks berupa pdf yang berjumlah 21 halaman dan memiliki 1.878 kata sebagai bahan yang digunakan untuk membangun model.

2) Pengolahan Data

Pada tahap ini proses pengolahan data dimulai dengan mengambil *file* PDF dari folder *Dataset_Cabai*, yang kemudian dimuat menggunakan modul *PyPDFLoader*. Pada setiap halaman dari dokumen berhasil diekstrak untuk dianalisis lebih lanjut. Selanjutnya, dokumen-dokumen ini dibagi menjadi bagian-bagian kecil sepanjang 1000 karakter dengan menggunakan *RecursiveCharacterTextSplitter*. Pembagian ini bertujuan agar model dapat memproses teks dengan efisien sambil tetap menjaga struktur informasi pada data. Hasil akhirnya adalah sekumpulan data yang siap digunakan untuk *embedding* dan analisis.

3) *Embedding* Dan Penyimpanan Vektor

Tahap ini menggunakan model *embedding* dari *HuggingFaceEmbeddings* dengan arsitektur *sentence-transformers/bert-base-nli-max-tokens* untuk mengubah teks menjadi representasi vektor. Setiap bagian dokumen yang telah dipecah diolah menjadi vektor dan disimpan menggunakan *Chroma*, sebuah sistem penyimpanan vektor yang efisien. Data tersebut disimpan dalam direktori '*data_laporan*', yang memudahkan pencarian dokumen relevan berdasarkan kemiripan vektor. Proses ini memastikan bahwa data dapat diakses dengan cepat untuk tahap pencarian dan analisis lebih lanjut.

4) Pencarian Data Informasi

Pada tahap ini, data yang telah disimpan dalam bentuk vektor akan digunakan untuk mencari dokumen relevan berdasarkan pertanyaan dari pengguna. Sistem akan menghitung kemiripan antara *embedding* pertanyaan dengan *embedding* dokumen menggunakan *cosine similarity*.

5) Penyusunan Jawaban Pada Model

Setelah dokumen yang relevan ditemukan, model generatif *ChatGoogleGenerativeAI* akan mengolah informasi tersebut untuk menyusun jawaban yang tepat. Model ini memastikan bahwa jawaban yang dihasilkan didasarkan pada data dan relevan dengan pertanyaan yang diajukan oleh pengguna.

6) Mengukur Tingkat Kemiripan

Di tahap ini peneliti akan memuat hasil dari proses pembagian data yang digunakan dalam pembuatan *embedding* dan penghitungan *similarity* agar model dapat belajar mengenali pola dalam data sehingga memberikan hasil yang akurat.

7) Pengujian Sistem

Pada tahap ini sistem diuji untuk memastikan kinerja proses *retrieval* dan generasi jawaban. Pengujian *retrieval* yang dilakukan pada sistem untuk menampilkan dokumen dengan tingkat kemiripan tertinggi agar memastikan hasil pada pencarian relevan. Sedangkan pengujian generasi jawaban yang dihasilkan akan dievaluasi berdasarkan keakuratan dan relevansinya terhadap pertanyaan pengguna.

3.3 *Extreme Programming*

Extreme Programming (XP) adalah metode pengembangan perangkat lunak yang bersifat ringan dan termasuk dalam kategori *agile*. Metode ini telah mendapatkan popularitas di kalangan pengembang karena pendekatannya yang sederhana, adaptif, dan fleksibel. XP

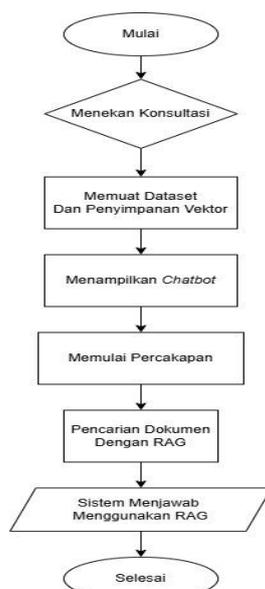
bertujuan untuk menyederhanakan proses pengembangan perangkat lunak, sehingga memudahkan tim dalam menghadapi perubahan kebutuhan dan menghasilkan solusi dengan lebih cepat serta berkualitas tinggi[10][11].

1) *Planning*

Tahap ini meliputi penentuan konteks bisnis untuk aplikasi, penetapan hasil yang diharapkan, fitur-fitur yang akan disediakan, fungsi-fungsi dari aplikasi yang sedang dikembangkan, serta estimasi waktu dan biaya. Di samping itu, juga mencakup alur kerja aplikasi.

2) *Design*

Pada tahap desain selanjutnya, dibuatlah visualisasi *flowchart* untuk menggambarkan proses kerja aplikasi yang akan dikembangkan. Visualisasi ini dapat dilihat pada gambar 2 berikut ini:



Gambar 2. *Flowchart* Sistem

Pada gambar 2 di atas menjelaskan alur kerja sistem konsultasi sekarang menggunakan *chatbot* untuk mendiagnosa penyakit berdasarkan gejala yang ditanyakan oleh *user*. Dimana *user* menekan opsi “Konsultasi Sekarang” yang akan memicu sistem untuk menampilkan *chatbot*. Setelah *chatbot* muncul, *user* dapat memulai percakapan dengan memasukkan pertanyaan terkait gejala penyakit yang dialami oleh tanaman. Sistem kemudian akan memeriksa apakah pertanyaan yang diajukan sesuai dengan gejala yang bisa dikenali oleh sistem. Jika pertanyaan tersebut sesuai, sistem akan memberikan jawaban menggunakan metode RAG (*Retrieval-Augmented Generation*). Jika pertanyaan tidak sesuai, pengguna diarahkan untuk memperbaiki pertanyaannya. Proses ini berakhir ketika sistem berhasil memberikan jawaban yang relevan untuk pertanyaan yang diajukan.

3) *Coding*

Pada tahap ini, beberapa *library* penting akan dimasukkan untuk mendukung pengembangan sistem. Di antaranya, yaitu *pandas* dan *numpy* yang digunakan untuk manipulasi data, sementara komponen dari *langchain* dimanfaatkan untuk pemrosesan bahasa alami. *Library dotenv* berperan dalam mengelola variabel lingkungan, sehingga memungkinkan konfigurasi yang lebih fleksibel dan aman. Selanjutnya, model *embedding* akan diinisialisasi menggunakan *HuggingFaceEmbeddings* yang mana model ini berfungsi untuk mengubah teks menjadi representasi vektor yang lebih sehingga dapat dipahami oleh mesin. Setelah itu, penyimpanan vektor dibuat menggunakan *Chroma*, yang menyediakan kemampuan pencarian berbasis kemiripan. Dengan memanfaatkan struktur ini, sistem dapat dengan efisien menemukan dan mengambil informasi yang relevan dari *dataset*, yang berfungsi sebagai dasar dalam menjawab pertanyaan pengguna terkait penyakit tanaman cabai merah keriting.

4) *Testing*

Tahap pengujian dilakukan dengan metode pengujian *blackbox*, yang menguji seluruh fitur fungsional perangkat lunak secara menyeluruh. Pada tahap awal, pengujian akan mengevaluasi apakah aplikasi memenuhi semua persyaratan yang diinginkan. Selanjutnya, pengujian akan memeriksa validitas *input* yang diberikan untuk memastikan proses berjalan dengan benar. Setelah pengujian selesai, pengujian akan melaporkan semua *bug* yang ditemukan kepada pengembang untuk diperbaiki.

3.4 Model Arsitektur Sistem Pakar

1) Analisa Basis Pengetahuan

Langkah awal dalam pengembangan sistem pakar adalah merancang struktur basis pengetahuan. Basis pengetahuan sendiri merupakan sekumpulan fakta yang dikumpulkan [12]. Data yang disimpan dalam basis pengetahuan diperoleh dari Bapak Ahmad Zaeni, yang merupakan pakar dalam bidang tanaman cabai.

Tabel 1 berikut memuat informasi mengenai berbagai jenis penyakit yang dapat menyerang tanaman cabai.

Tabel 1. Jenis Penyakit Pada Tanaman Cabai

Id Penyakit.	Nama Penyakit	Jenis Penyakit
P1	Penyakit Virus kuning (<i>Gemini Virus</i>)	Daun
P2	Penyakit Busuk Buah	Buah
P3	Penyakit Keriting Daun	Daun
P4	Penyakit Layu Fusarium (<i>Fusarium Oxysporum. Sp</i>)	Daun

Tabel di bawah ini mencantumkan 4 jenis penyakit yang menyerang tanaman cabai. Selain itu, terdapat juga data gejala untuk setiap penyakit yang dijelaskan dalam tabel berikut.

Tabel 2. Daftar Gejala untuk Setiap Jenis Penyakit

No.	Nama Penyakit	Gejala
1.	Penyakit Virus kuning (<i>Gemini Virus</i>)	a. Daun yang menguning merata b. Pertumbuhan yang terhambat c. Daun melengkung ke atas
2.	Penyakit Busuk Buah	a. Muncul bercak berwarna cokelat atau hitam pada permukaan buah b. Aroma tidak sedap c. Buah berair dan lembek
3.	Penyakit Keriting Daun	a. Daun berkerut b. Daun menggulung c. Tanaman menjadi kerdil
4.	Penyakit Layu Fusarium (<i>Fusarium Oxysporum. Sp</i>)	a. Tanaman mengalami kelayuan b. Daun menguning dan keriting c. Batangnya cenderung berwarna coklat

2) Model Inferensi Sistem

Sistem pakar berbasis *web* menggunakan algoritma *forward chaining* sebagai mesin inferensi untuk mendiagnosa penyakit pada tanaman cabai merah keriting.

Berikut merupakan aturan yang akan menghubungkan antara gejala yang teramati dengan penyakit yang dialami oleh tanaman cabai merah keriting. Aturan ini bersifat heuristik, yang berarti dapat membantu dalam mengidentifikasi jenis penyakit. Aturan-aturan ini dapat digunakan untuk mengembangkan model klasifikasi yang efektif dengan menggunakan metode *Natural Language Processing* (NLP). Berikut ini adalah beberapa data aturan yang telah diidentifikasi dalam menentukan gejala dan jenis penyakitnya:

- Jika daun menguning dimulai dari bagian tengah hingga menyebar, daun menjadi kecil dan muncul bercak kuning terang, pertumbuhan tanaman terhambat dan buah tidak berkembang sempurna, maka kemungkinan mengalami gejala penyakit virus kuning.

- Jika buah mengalami area lembap berwarna coklat kehitaman di bagian bawah, area busuk meluas dan buah menjadi lunak serta mengeluarkan bau tidak sedap, mengalami kelembapan tinggi yang terdeteksi dan ditemukan jamur patogen seperti *Phytophthora*, maka kemungkinan mengalami gejala penyakit busuk buah.
- Jika daun muda terlihat mengeriting dan permukaannya tidak rata, daun berubah warna menjadi hijau gelap atau kekuningan, ditemukan hama seperti kutu kebul atau *thrips* pada tanaman, maka kemungkinan mengalami gejala penyakit keriting daun.
- Jika daun tanaman mulai layu secara bertahap dimulai dari bagian bawah, daun berubah warna menjadi kuning kemudian coklat dan akhirnya mengering, akar membusuk dan jaringan batang menunjukkan warna coklat, maka kemungkinan mengalami gejala penyakit layu *fusarium*.

3) Antarmuka Pengguna (*User Interface*)

Antarmuka ini dirancang untuk memudahkan pengguna dalam memasukkan informasi, mengajukan pertanyaan, dan menerima jawaban atau rekomendasi dari sistem. Dalam konteks sistem pakar yang mengintegrasikan pemrosesan bahasa alami (NLP), antarmuka pengguna dapat berupa aplikasi *web*, aplikasi *desktop*, atau *chatbot* yang mendukung komunikasi berbasis teks atau suara.

4. Hasil dan Pembahasan

4.1 Implementasi Model RAG

1) *Dataset*

Penyakit Virus Kuning Keriting adalah penyakit yang disebabkan oleh infeksi virus yang menyerang tanaman, mempunyai gejala seperti daun keriting, daun menguning, dan mengecil yang disebabkan oleh virus gemini melalui penularan oleh kutu kebul (*bemisia tabaci*) sehingga mempunyai dampak yaitu pertumbuhan yang terhambat dan hasil menurun untuk cara pengobatannya dapat dilakukan langkah seperti pengendalian kutu kebul dengan insektisida, penggunaan mulsa plastik, dan penanaman varietas tahan virus.

Penyakit Layu *Fusarium* mempunyai gejala seperti tanaman layu, daun menguning dan kering, terutama pada bagian bawah yang disebabkan oleh jamur *fusarium oxysporum*, sehingga mempunyai dampak yaitu tanaman mati sebelum panen dan hasil berkurang pengobatannya dapat dilakukan melalui rotasi tanaman, aplikasi fungisida, dan menjaga kebersihan lahan.

Penyakit Bercak Daun *Alternaria* mempunyai gejala seperti bercak kecil bulat pada daun dan berwarna coklat dengan tepi kuning, yang disebabkan oleh jamur *alternaria solani* sehingga mempunyai dampak yaitu daun rontok dan mengurangi kemampuan fotosintesis pengobatannya dapat dilakukan melalui pengendalian dengan fungisida, rotasi tanaman, dan menjaga jarak tanam untuk sirkulasi udara yang baik.

Penyakit Layu Bakteri mempunyai gejala seperti tanaman layu mendadak, batang lunak dan berair yang disebabkan oleh bakteri *ralstonia solanacearum* sehingga mempunyai dampak yaitu tanaman mati dan hasil panen hilang pengobatannya dapat dilakukan melalui pengendalian dengan rotasi tanaman, pembersihan alat yang digunakan, penggunaan bakteri antagonis, dan menjaga sanitasi lahan.

Gambar 3. *Dataset*

Tahap ini melakukan pengumpulan data yang diperlukan sebagai bahan untuk membangun model yang diperoleh melalui wawancara langsung bersama pakar pertanian yaitu Bapak Ahmad Zaini, yang mana hasil dari wawancara tersebut kemudian disusun kedalam bentuk dokumen PDF yang terdiri dari 21 halaman dan memiliki 1.878 kata.

2) Pengolahan Data

Pada tahap ini proses pengolahan data dimulai dengan mengambil *file* PDF dari folder *Dataset_Cabai*, yang kemudian dimuat menggunakan modul *PyPDFLoader*. Pada setiap halaman dari dokumen berhasil diekstrak untuk dianalisis lebih lanjut. Selanjutnya, dokumen-dokumen ini dibagi menjadi bagian-bagian kecil sepanjang 1000 karakter dengan menggunakan *RecursiveCharacterTextSplitter*. Pembagian ini bertujuan agar model dapat memproses teks dengan efisien sambil tetap menjaga struktur informasi pada data. Hasil akhirnya adalah sekumpulan data yang siap digunakan untuk *embedding* dan analisis.

3) *Embedding* Dan Penyimpanan Vektor

Tahap ini menggunakan model *embedding* dari *HuggingFaceEmbeddings* dengan arsitektur *sentence-transformers/bert-base-nli-max-tokens* untuk mengubah teks menjadi

representasi vektor. Setiap bagian dokumen yang telah dipecah diolah menjadi vektor dan disimpan menggunakan *Chroma*, sebuah sistem penyimpanan vektor yang efisien. Data tersebut disimpan dalam direktori 'data_laporan', yang memudahkan pencarian dokumen relevan berdasarkan kemiripan vektor. Proses ini memastikan bahwa data dapat diakses dengan cepat untuk tahap pencarian dan analisis lebih lanjut.

```
Berhasil memuat 21 halaman dari DATASET CABAI.pdf
Jumlah total bagian dokumen: 72
Penyimpanan vektor berhasil dibuat dan disimpan.
D:\sispak\c.py:86: LangChainDeprecationWarning: The
nstead.
conversation_chain = LLMChain(
```

Gambar 4. Penyimpanan Vektor

4) Pencarian Data Informasi

Pada tahap ini, data yang telah disimpan dalam bentuk vektor akan digunakan untuk mencari dokumen relevan berdasarkan pertanyaan dari pengguna. Sistem akan menghitung kemiripan antara *embedding* pertanyaan dengan *embedding* dokumen menggunakan *cosine similarity*.

5) Penyusunan Jawaban Pada Model

Setelah dokumen yang relevan ditemukan, model generatif *ChatGoogleGenerativeAI* akan mengolah informasi tersebut untuk menyusun jawaban yang tepat. Model ini memastikan bahwa jawaban yang dihasilkan didasarkan pada data dan relevan dengan pertanyaan yang diajukan oleh pengguna.

6) Mengukur Tingkat Kemiripan Untuk Pengujian Model

Pada tahap ini hasil data yang dimuat melalui proses tersebut dapat digunakan untuk pembuatan *embedding* dan perhitungan *similarity*. Model akan belajar mengenali pola dalam bentuk data agar dapat memberikan hasil yang akurat. Pada gambar di atas menunjukkan tampilan hasil pada tingkat relevansi dokumen terkait dengan jawaban yang dihasilkan oleh model LMM berdasarkan pertanyaan dari pengguna. Nilai kesamaan yang berkisar antara 0,9672 hingga 0,9503 memperlihatkan seberapa besar tingkat kesamaan semantik antara konten dokumen dan respon yang dihasilkan oleh model. Dapat dilihat untuk dokumen dengan skor tertinggi terdapat pada angka 0,9672 dinilai sebagai yang paling relevan dan memuat informasi yang paling tepat untuk dijadikan sebagai acuan utama. Sedangkan dokumen yang memiliki skor lebih rendah terdapat pada angka 0,9503 hal ini tetap relevan, meskipun mungkin tidak sekomprehensif dokumen dengan skor yang lebih tinggi.

```
Hasil pencarian berdasarkan respons dengan tingkat kemiripan:
Tingkat kemiripan: 0.9672
Tingkat kemiripan: 0.9626
Tingkat kemiripan: 0.9609
Tingkat kemiripan: 0.9582
Tingkat kemiripan: 0.9578
Tingkat kemiripan: 0.9575
Tingkat kemiripan: 0.9568
Tingkat kemiripan: 0.9523
Tingkat kemiripan: 0.9521
Tingkat kemiripan: 0.9503
File Excel disimpan di embedding_manual_laporan.xlsx
(.lib) PS D:\sispak> █
```

Gambar 5. Hasil *Similarity*

Pada tahap di bawah ini akan dilakukan pengujian untuk mengevaluasi kinerja model dengan menggunakan data pengujian *input* berupa *query*, yang bertujuan untuk memastikan bahwa model dapat memberikan respon berdasarkan data yang ada dalam sistem.



Gambar 6. Pengujian Model RAG

Pada gambar 6 di atas menunjukkan hasil pengujian RAG dalam model NLP yang dirancang untuk memberikan solusi terhadap penyakit yang dialami oleh tanaman cabai merah keriting dengan tingkat kesamaan sebesar 0,9672 (97%) antara jawaban sistem dengan data pakar. Skor similaritas yang tinggi menunjukkan bahwa sistem ini efektif dan relevan untuk mendiagnosa penyakit tanaman cabai merah keriting. Hasil pengujian menunjukkan bahwa penggunaan teknologi *langchain memory* memungkinkan model untuk menyimpan informasi dari masukan pengguna, yang kemudian dapat dimanfaatkan dalam proses *forward chaining*, yang mana semakin banyak data yang dikumpulkan melalui *langchain memory*, maka semakin lengkap dan akurat fakta-fakta yang diolah untuk menghasilkan keputusan. *Prompt* pada model disesuaikan agar berperan sebagai asisten bernama *Capsicoria*, *query* yang diberikan berupa gejala penyakit yang terjadi pada tanaman cabai merah keriting seperti daun keriting, daun menguning, dan mengecil. Maka sistem akan menjelaskan nama penyakit tersebut yaitu virus kuning yang disebabkan oleh virus Gemini melalui penularan oleh kutu kebul, dan solusi dari penyakit tersebut yaitu membersihkan area sekitar tanaman dari gulma dan sisa-sisa tanaman yang terinfeksi agar kutu kebul tidak dapat berkembang biak di tempat-tempat tersebut. Sistem juga menekankan pentingnya mengamati dan merawat tanaman cabai merah keriting secara cermat untuk mencegah serangan penyakit atau hama. Jawaban yang disampaikan dirancang dengan terstruktur dan informatif, sehingga membantu petani memahami solusi yang tepat untuk tanaman mereka.

4.2 Implementasi Website

1) Halaman *Home*

Gambar 7. Halaman *Home*

Pada gambar 7 di atas menunjukkan halaman *home Capsicoria* yang merupakan sebuah sistem pakar untuk diagnosa penyakit pada cabai merah keriting. deskripsi singkat fungsi sistem pakar, serta tombol konsultasi sekarang untuk memulai konsultasi bersama sistem pakar.

2) Halaman Informasi



Gambar 8. Halaman Informasi

Pada gambar 8 di atas merupakan halaman opsi informasi yang menunjukkan informasi jenis penyakit umum tanaman cabai merah keriting dengan memberikan penjelasan mengenai nama penyakit dan cara mengatasinya.

3) Halaman Team

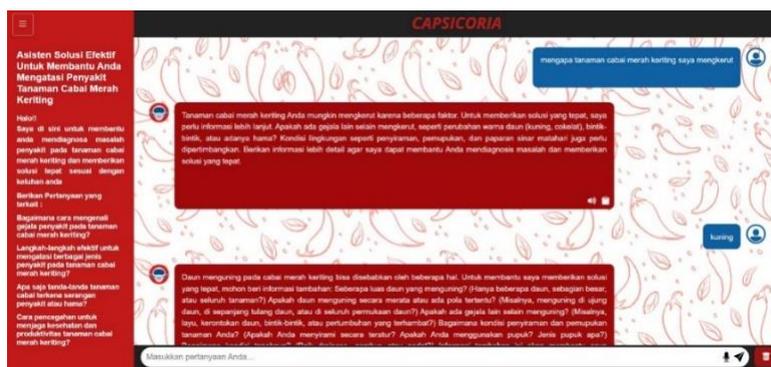


Gambar 9. Halaman Team

Tampilan *team* dari *website* sistem pakar *Capsicoria* berisi tentang informasi terkait pengembang dari sistem pakar *Capsicoria*.

4) Halaman Konsultasi

Pada gambar 10 di atas menunjukkan tampilan dari opsi Konsultasi Sekarang yang mengarahkan pengguna untuk memulai sesi konsultasi untuk bertanya mengenai penyakit cabai merah keriting yang dialami dalam bentuk teks. Dimana pengguna akan berinteraksi dengan sistem melalui format tanya jawab, dan sistem akan merespons pertanyaan pengguna mengenai gejala atau kondisi tanaman cabai yang sedang mengalami masalah



Gambar 10. Halaman Konsultasi

4.3 Hasil Uji Implementasi Sistem

Tabel 3. Hasil Pengujian Sistem

No.	Fitur	Fungsi	Ket
1.	Halaman <i>Home</i>	Melihat dan mengakses fitur yang ada	Berhasil
2.	Halaman Informasi	Melihat dan mengetahui informasi umum mengenai penyakit dari tanaman cabai merah keriting	Berhasil
3.	Halaman <i>Team</i>	Melihat dan mengetahui pembuat sistem	Berhasil
4.	Konsultasi Sekarang	Memberikan informasi secara <i>realtime</i> dalam bentuk <i>chat</i>	Berhasil
5.	Salin	Menyalin jawaban dari sistem	Berhasil
6.	<i>Speaker</i>	Untuk mendengarkan jawaban dari sistem	Berhasil
7.	<i>Mic</i>	Menyampaikan pertanyaan dalam bentuk suara	Berhasil
8.	Sampah	Membersihkan <i>room chat</i>	Berhasil
9.	Saran Pertanyaan	Sebagai pesan cepat dari <i>user</i>	Berhasil

Hasil uji implementasi pada sistem dilakukan dengan menggunakan metode pengujian *black box* yang akan menjadi fokus utama untuk menganalisis dan mengidentifikasi kinerja dari aplikasi tanpa memperlihatkan detail internalnya. Dengan cara ini, evaluasi akan dilakukan terhadap cara kerja aplikasi dalam situasi nyata, serta memberikan pemahaman yang lebih mendalam mengenai potensi perbaikan yang mungkin diperlukan.

4.4 Pembahasan

Konsep yang diusulkan dalam penelitian ini memiliki potensi besar untuk menyelesaikan masalah utama dalam diagnosa penyakit tanaman cabai merah keriting, yaitu keterbatasan akses terhadap informasi yang akurat dan cepat bagi petani. Dengan memanfaatkan metode *Natural Language Processing* (NLP), sistem dapat mengolah masukan berbasis teks secara otomatis untuk mengidentifikasi gejala dan memberikan diagnosa penyakit yang relevan. Selain itu, penggabungan teknologi chatbot memungkinkan petani berinteraksi dengan sistem secara langsung dan mudah, sehingga mereka dapat memperoleh informasi dan solusi dengan cepat tanpa memerlukan pengetahuan khusus. Pendekatan ini tidak hanya meningkatkan efisiensi deteksi penyakit, tetapi juga memperluas akses informasi ke lebih banyak petani, membantu mereka mengambil langkah penanganan yang tepat untuk menjaga produktivitas tanaman cabai merah keriting.

Penelitian sebelumnya yang berjudul "*Expert System With Certainty Factor For Early Diagnosis Of Red Chili Peppers Diseases*"[13], tentang sistem pakar untuk diagnosis penyakit pada tanaman cabai telah dilakukan menggunakan berbagai metode. Salah satu penelitian menggunakan metode *certainty factor* untuk mendiagnosa penyakit tanaman cabai, dengan melibatkan 6 jenis penyakit dan 22 gejala yang mempunyai tingkat akurasi dicapai sebesar 76%. Dengan demikian, penelitian ini memerlukan penambahan data terkait penyakit dan gejala agar hasilnya lebih lengkap.

Penelitian lain yang berjudul “Sistem Pakar Deteksi Hama Dan Penyakit Pada Tanaman Cabai Dengan Metode *Naïve Bayes*”[14], menggunakan metode *Naïve Bayes* untuk mendiagnosa penyakit pada tanaman cabai, dengan melibatkan 15 jenis hama dan penyakit serta 41 data gejala. Tingkat akurasi yang diperoleh dari penelitian tersebut mencapai 90,47%.

Sedangkan penelitian yang berjudul “Rancang Bangun Sistem Pakar Untuk Mendiagnosis Tanaman Cabai Menggunakan Metode *Bayes*”[15], untuk mendiagnosa hama dan penyakit pada tanaman cabai. Dalam penelitian tersebut terdapat 11 jenis hama dan penyakit yang dianalisis, serta 34 gejala yang digunakan sebagai data yang mempunyai hasil pengujian mencapai 0,32.

Penelitian dengan memanfaatkan teknologi kecerdasan buatan khususnya *Natural Language Processing* (NLP) memberikan peran yang penting dalam memperkuat relevansi dan keberlanjutan penelitian sebelumnya terkait diagnosa penyakit pada tanaman cabai merah keriting. Dibandingkan dengan penelitian lainnya, pendekatan ini memberikan beberapa keunggulan yang istimewa, seperti tingkat akurasi yang sangat tinggi mencapai 97%, penerapan teknologi AI yang lebih fleksibel, serta kemampuan untuk memproses informasi secara lebih spesifik dan terarah. Penelitian ini berhasil mengembangkan sistem pakar yang tidak hanya cerdas, tetapi juga lebih mudah dipahami, fleksibel, dan efisien dalam mendiagnosa penyakit tanaman. Pendekatan ini memperlihatkan potensi besar NLP dalam meningkatkan keakuratan, kecepatan, skala penerapan, serta kemudahan penggunaan sistem, sehingga memberikan solusi yang lebih efektif dan unggul bagi petani. Dengan demikian, penelitian ini bukan hanya memperluas cakupan penerapan teknologi AI di bidang pertanian, tetapi juga memberikan inovasi yang lebih relevan dan berdampak dibandingkan penelitian sebelumnya.

5. Simpulan

Kesimpulan proyek ini berisi mengenai sistem pakar berbasis *Natural Language Processing* (NLP) yang telah digabungkan dengan metode *Retrieval-Augmented Generation* (RAG) dan telah berhasil mencapai tingkat kemiripan antara jawaban sistem dengan data pakar yang berasal dari hasil *embedding* hingga 0,9672 (97%). Melalui metode teknologi *langchain memory*, model dapat mengumpulkan informasi dari masukan pengguna untuk digunakan pada proses penginputan data yang lengkap dan akurat agar dapat diolah menjadi keputusan. Selain itu sistem juga dilengkapi oleh fitur *Speech-to-Text* untuk mengenali masukan suara dan *Text-to-Speech* untuk menyampaikan respon dalam bentuk *audio*, serta kemampuan untuk menganalisis dan memberikan jawaban terhadap pertanyaan.

Daftar Referensi

- [1] Isnirobit, “Analisis Pengaruh Luas Panen, Harga Jual dan Produktivitas Terhadap Jumlah Produksi Cabai Merah (*Capsicum annum* L.) di Indonesia Tahun 1999-2019,” *AGRILAND Jurnal Ilmu Pertanian*, vol. 10, no. 3, pp. 278–290, Des 2022.
- [2] I. G. B. T. Ananta dan D. G. A. Anjasmara, “Antioxidant and Antibacterial Potency of Red Chillies Extract (*Capsicum annum* var. Longum),” *Jurnal Ilmiah Medicamento*, vol. 8, no. 1, pp. 48–55, Mar 2022, doi: 10.36733/medicamento.v8i1.3170.
- [3] I. Sari, N. D. Yanti, dan T. Hidayat, “Faktor-Faktor Yang Mempengaruhi Usahatani Cabai Rawit (*Capsicum fretescens* L.) Di Kabupaten Tabalong,” *Frontier Agribisnis*, vol. 3, no. 4, pp. 23–30, 2020.
- [4] M. R. Alamsyah dan H. Kurniawan, “Sistem Pakar Menggunakan Metode Certainty Factor untuk Mendiagnosa Hama dan Penyakit pada Tanaman Cabai,” *Jurnal Teknologi Informasi*, vol. 16, no. 2, pp. 38–45, Jul 2021.
- [5] E. Larasati Amalia dan D. Wahyu Wibowo, “Rancang Bangun Chatbot Untuk Meningkatkan Performa Bisnis,” *Jurnal Ilmiah Teknologi Informasi Asia*, vol. 13, no. 2, hlm. 137–142, 2019.
- [6] T. Christy, “Implementasi Sistem Pakar Penyakit Cabai dengan Metode Certainty Factor,” *Jurnal Pendidikan Tambusai*, vol. 6, no. 1, pp. 1546–1551, 2022, doi: <https://doi.org/10.31004/jptam.v6i1.3152>.
- [7] S. Andriani, “Sistem Pakar Identifikasi Penyakit Tanaman Cabai Rawit Menggunakan Metode Forward Chaining Berbasis Web,” *J-INTECH (Journal of Information and Technology)*, vol. 8, no. 2, pp. 85–88, Des 2020.
- [8] V. K. Pratap dan N. Suresh Kumar, “High-precision multiclass classification of chili leaf disease through customized EffecientNetB4 from chili leaf images,” *Smart Agricultural Technology*, vol. 5, pp. 1–11, Jul 2023, doi: 10.1016/j.atech.2023.100295.

-
- [9] N. A. M. Herwanza, N. S. Harahap, F. Yanto, dan F. Insani, "Penerapan Langchain Retriever dengan Model Chat Openai dalam Pengembangan Sistem Chatbot Hadis Berbasis Telegram," *JTIM: Jurnal Teknologi Informasi dan Multimedia*, vol. 6, no. 1, pp. 70–83, Mei 2024, doi: 10.35746/jtim.v6i1.514.
- [10] M. Megawaty dan N. Huda, "Pembaharuan Sistem Penentuan Untuk Klasifikasi Jenis Penyakit pada RSUD Sekayu Menggunakan Pendekatan Extreme Programming," *Jurnal Media Informatika Budidarma*, vol. 5, no. 1, pp. 66, Jan 2021, doi: 10.30865/mib.v5i1.2273.
- [11] Rudianto, "Penerapan Metode Extreme Programming Dalam Pembangunan Aplikasi Sistem Penunjang Keputusan," *Jurnal Sistem Informasi Dan Bisnis Cerdas*, vol. 16, no. 1, pp. 21–30, 2023.
- [12] N. Sulardi dan A. Witanti, "Sistem Pakar Untuk Diagnosis Penyakit Anemia Menggunakan Teorema Bayes," *Jurnal Teknik Informatika (Jutif)*, vol. 1, no. 1, pp. 19–24, Jul 2020, doi: 10.20884/1.jutif.2020.1.1.12.
- [13] F. Agus, H. E. Wulandari, dan I. F. Astuti, "Expert System With Certainty Factor For Early Diagnosis Of Red Chili Peppers Diseases," *Journal of Applied Intelligent System*, vol. 2, no. 2, pp. 52–66, Des 2017.
- [14] Y. A. Fitrianingtyas dan C. Rahmad, "Sistem Pakar Deteksi Hama Dan Penyakit Pada Tanaman Cabai Dengan Metode Naïve Bayes," *J. Polinema*, vol. 2, no. 1, pp. 68–72, 2015.
- [15] A. Mahmudi, M. M. Rokhman, dan A. E. Prasetio, "Rancang Bangun Sistem Pakar Untuk Mendiagnosis Tanaman Cabai Menggunakan Metode Bayes," *Jurnal Rekayasa Dan Manajemen Sistem Informasi*, vol. 2, no. 2, pp. 85–90, 2016.